

9.3

IBM MQ Technický přehled

IBM

Poznámka

Než začnete používat tyto informace a produkt, který podporují, přečtěte si informace, které uvádí [“Poznámky” na stránce 297](#).

Toto vydání se vztahuje na verzi 9 vydání 3 produktu IBM® MQ a na všechna následná vydání a úpravy, není-li v nových vydáních uvedeno jinak.

Když odešlete informace na adresu IBM, udělujete IBM nevýhradní právo používat nebo distribuovat informace libovolným způsobem, který považuje za odpovídající, aniž by vám tím vznikl jakýkoliv závazek.

© **Copyright International Business Machines Corporation 2007, 2024.**

Obsah

Technický přehled.....	5
Úvod do řazení zpráv do fronty.....	5
Hlavní funkce a výhody řazení zpráv do fronty.....	7
Terminologie řazení zpráv do fronty.....	9
Zprávy a fronty.....	12
IBM MQObjekty.....	14
Typy objektů.....	15
Pojmenování objektů IBM MQ.....	35
Distribuované řazení do front a klastry.....	41
Komponenty distribuovaného řazení do front.....	45
Komponenty klastru.....	53
Systém zpráv publikování/odběru.....	59
Komponenty publikování/odběru.....	59
Příklad konfigurace publikování/odběru jednoho správce front.....	84
Distribuované sítě publikování/odběru.....	85
IBM MQ Výběrové vysílání.....	102
Počáteční koncepce výběrového vysílání.....	102
Přehled produktu MQ Telemetry.....	103
Úvod do produktu MQ Telemetry.....	105
Případy použití telemetrie.....	106
Připojení telemetrických zařízení ke správci front.....	112
Protokoly připojení telemetrie.....	113
Služba (MQXR) telemetrie.....	113
Kanály telemetrie.....	113
IBM MQ Telemetry Transport protokol.....	114
MQTT klienti.....	114
Odeslání zprávy klientovi MQTT.....	115
Odeslání zprávy aplikaci IBM MQ z klienta MQTT.....	123
MQTT publikování/odběr aplikací.....	124
Aplikace telemetrie.....	125
Integrace produktu MQ Telemetry se správcem front.....	125
MQTT bezstavové a stavové relace.....	128
Když není klient MQTT připojen.....	128
Volné spojení mezi klienty MQTT a aplikacemi IBM MQ.....	129
Zabezpečení MQ Telemetry.....	129
MQ Telemetry globalizace.....	130
Výkon a rozšiřitelnost produktu MQ Telemetry.....	130
Zařízení podporovaná produktem MQ Telemetry.....	132
Zabezpečení v adresáři IBM MQ.....	133
Podpora zabezpečení TLS spravovaného klienta IBM MQ.NET.....	134
IBM MQ MQI clients.....	135
Proč používat klienty IBM MQ ?.....	137
Co je to rozšířený transakční klient?.....	138
Jak se klient připojuje k serveru.....	139
Správa a podpora transakcí.....	141
Rozšíření prostředků správce front.....	142
Jazyková rozhraní produktu IBM MQ Java.....	143
IBM MQ classes for JMS/Jakarta Messaging.....	144
IBM MQ Poskytovatel systému zpráv.....	154
IBM MQ for z/OS koncepce.....	154
Správce front v systému z/OS.....	156
Inicializátor kanálu v systému z/OS.....	157

Podmínky a úlohy pro správu produktu IBM MQ for z/OS.....	159
Sdílené fronty a skupiny sdílení front.....	161
Použití front v rámci skupiny.....	205
Správa úložiště na systému z/OS.....	218
přihlášení IBM MQ for z/OS.....	222
Definice systému v systému z/OS.....	233
Obnova a restart na systému z/OS.....	244
Koncepce zabezpečení v produktu IBM MQ for z/OS.....	260
Dostupnost na z/OS.....	267
Monitorování a statistiky na IBM MQ for z/OS.....	270
Dispozice jednotky zotavení na systému z/OS.....	271
IBM MQ a další produkty z/OS.....	273
IBM MQ a CICS.....	274
IBM MQ a IMS.....	275
IBM MQ a adaptéry z/OS Batch, TSO a RRS.....	279
IBM MQ for z/OS a WebSphere Application Server.....	280
Managed File Transfer.....	281
Jak MFT pracuje s produktem IBM MQ?.....	283
MFT přehled topologie.....	284
Přehled produktu MFTREST API.....	285
IBM MQ Internet Pass-Thru.....	285
Použití MQIPT.....	286
Jak MQIPT funguje.....	288
Možné konfigurace produktu MQIPT.....	289
Kompatibilní konfigurace.....	291
Podporované konfigurace kanálů.....	293
Podmínky ukončení a selhání kanálu.....	294
Bezpečnost zpráv.....	294
Správci front s více instancemi a vysoká dostupnost.....	294
IBM MQ Console a REST API.....	295
Poznámky.....	297
Informace o programovacím rozhraní.....	298
Ochranné známky.....	298

IBM MQ Technický přehled

Produkt IBM MQ použijte k připojení aplikací a správě distribuce informací v rámci vaší organizace.

Produkt IBM MQ umožňuje programům vzájemnou komunikaci v síti různých komponent (procesorů, operačních systémů, subsystémů a komunikačních protokolů) pomocí konzistentního rozhraní API. Aplikace navržené a napsané pomocí tohoto rozhraní jsou známé jako aplikace pro řazení zpráv do front.

Následující dílčí témata slouží k vyhledání informací o řazení zpráv do fronty a dalších funkcích poskytovaných produktem IBM MQ.

Související pojmy

[Úvod do produktu IBM MQ](#)

[Kde najít požadavky na produkt a informace o podpoře](#)

Související úlohy

[Plánování architektury IBM MQ](#)

Související odkazy

[“Hlavní funkce a výhody řazení zpráv do fronty” na stránce 7](#)

Tyto informace zdůrazňují některé funkce a výhody řazení zpráv do fronty. Popisuje funkce, jako je zabezpečení a integrita dat front zpráv.

Úvod do řazení zpráv do fronty

Produkty IBM MQ umožňují programům vzájemnou komunikaci v síti různých komponent (procesorů, operačních systémů, subsystémů a komunikačních protokolů) pomocí konzistentního rozhraní API.

Aplikace navržené a napsané pomocí tohoto rozhraní jsou označovány jako aplikace *řazení zpráv do front*, protože používají styl *systému zpráv a řazení do front*:

- Systém zpráv znamená, že programy komunikují tak, že si vzájemně posílají data ve zprávách, místo aby si vzájemně volaly přímo.
- Řazení do fronty znamená, že zprávy jsou ukládány do front v paměti, což umožňuje, aby programy byly spouštěny nezávisle na sobě, v různých rychlostech a časech, na různých místech a bez logického spojení mezi nimi.

Řazení zpráv do fronty se používá při zpracování dat po mnoho let. Dnes se nejčastěji používá v elektronické poště. Bez řazení do fronty vyžaduje odeslání elektronické zprávy na dlouhé vzdálenosti, aby byl každý uzel na trase k dispozici pro předávání zpráv, a adresáti musí být přihlášení a vědomi si skutečnosti, že se jim pokoušíte odeslat zprávu. V systému front jsou zprávy uloženy na mezilehlých uzlech, dokud není systém připraven je předat. V místě jejich konečného určení jsou uloženy v elektronické schránce, dokud není adresát připraven je číst.

I tak je dnes mnoho složitých obchodních transakcí zpracováváno bez nutnosti řazení do fronty. Ve velké síti může systém udržovat tisíce připojení v připraveném stavu. Pokud jedna část systému trpí problémem, mnoho částí systému se stane nepoužitelnými.

Fronty zpráv lze považovat za elektronickou poštu pro programy. V prostředí fronty zpráv provádí každý program, který je součástí sady aplikací, dobře definovanou, samostatnou funkci v reakci na specifický požadavek. Chcete-li komunikovat s jiným programem, musí program vložit zprávu do předdefinované fronty. Druhý program načte zprávu z fronty a zpracuje požadavky a informace obsažené ve zprávě. Takže řazení zpráv do fronty je styl komunikace mezi programy.

Řazení do fronty je mechanismus, pomocí kterého jsou zprávy zadržovány, dokud není aplikace připravena je zpracovat. Řazení do fronty vám umožňuje:

- Komunikujte mezi programy (které mohou být spuštěny v různých prostředích), aniž byste museli psát komunikační kód.
- Vyberte pořadí, ve kterém program zpracovává zprávy.

- Vyvažování zátěže na systému uspořádáním více než jednoho programu pro obsluhu fronty, když počet zpráv překročí prahovou hodnotu.
- Zvyšte dostupnost aplikací uspořádáním alternativního systému pro obsluhu front v případě, že primární systém není k dispozici.

Co je to fronta zpráv?

Fronta zpráv, známá jednoduše jako fronta, je pojmenované místo určení, kam mohou být odesílány zprávy. Zprávy se hromadí ve frontách, dokud je nenačtou programy, které tyto fronty obsluhují.

Fronty jsou umístěny ve správci front a jsou tímto správcem spravovány (viz [“Terminologie řízení zpráv do fronty”](#) na stránce 9). Fyzická povaha fronty závisí na operačním systému, na kterém je spuštěn správce front. Fronta může být buď dočasnou oblastí vyrovnávací paměti v paměti počítače, nebo datovou sadou na zařízení trvalého úložiště (například na disku). Za fyzickou správu front odpovídá správce front a zúčastněným aplikačním programům to není zřejmé.

Programy přistupují k frontám pouze prostřednictvím externích služeb správce front. Mohou otevřít frontu, vložit do ní zprávy, získat z ní zprávy a zavřít frontu. Mohou také nastavit a dotazovat se na atributy front.

Různé styly řízení zpráv do fronty

dvoubodový

Jedna zpráva je umístěna do fronty a jedna aplikace tuto zprávu přijme.

V systému zpráv typu point-to-point musí odesílající aplikace znát informace o přijímající aplikaci, aby mohla odeslat zprávu této aplikaci. Odesílající aplikace může například potřebovat znát název fronty, do které mají být informace odeslány, a také zadat název správce front.

Publikování/odběr

Kopie každé zprávy publikované publikační aplikací je doručena každé zainteresované aplikaci. Může existovat mnoho, jedna nebo žádná aplikace, která by měla zájem. V publikování/odběru je zainteresovaná aplikace známa jako odběratel a zprávy jsou zařazeny do fronty identifikované odběrem.

Systém zpráv typu publikování/odběr vám umožňuje oddělit poskytovatele informací od spotřebitelů těchto informací. Odesílající a přijímající aplikace nemusí vědět tolik o sobě pro informace, které mají být odeslány a přijaty. Další informace viz [“Systém zpráv publikování/odběru”](#) na stránce 59.

Výhody řízení zpráv do fronty pro návrháře aplikací a vývojáře

Produkt IBM MQ umožňuje aplikačním programům používat *řízení zpráv do fronty* k účasti na zpracování řízeném zprávami. Aplikační programy mohou komunikovat napříč různými platformami pomocí příslušných softwarových produktů pro řízení zpráv do fronty. Aplikace z/OS mohou například komunikovat prostřednictvím IBM MQ for z/OS. Aplikace jsou chráněny před mechanikou základní komunikace. Mezi další výhody řízení zpráv do fronty patří:

- Můžete navrhnout aplikace pomocí malých programů, které můžete sdílet mezi mnoha aplikacemi.
- Pomocí těchto stavebních bloků můžete rychle vytvářet nové aplikace.
- Aplikace zapsané pro použití technik řízení zpráv do front nejsou ovlivněny změnami ve způsobu práce správců front.
- Nemusíte používat žádné komunikační protokoly. Správce front se zabývá všemi aspekty komunikace za vás.
- Programy, které přijímají zprávy, nemusí být spuštěny v době, kdy jsou jim odesílány. Zprávy jsou uchovány ve frontách.

Návrháři mohou snížit náklady na své aplikace, protože vývoj je rychlejší, je potřeba méně vývojářů a požadavky na znalosti programování jsou nižší než u aplikací, které nepoužívají fronty zpráv.

Produkt IBM MQ implementuje společné rozhraní API známé jako *rozhraní fronty zpráv* (nebo MQI) při každém spuštění aplikací. To vám usnadní portovat aplikační programy z jedné platformy na druhou.

Podrobnosti o rozhraní MQI viz [Přehled rozhraní fronty zpráv](#).

Hlavní funkce a výhody řazení zpráv do fronty

Tyto informace zdůrazňují některé funkce a výhody řazení zpráv do fronty. Popisuje funkce, jako je zabezpečení a integrita dat front zpráv.

Hlavní funkce aplikací, které používají techniky řazení zpráv do fronty, jsou:

- [“Žádná přímá spojení mezi programy”](#) na stránce 7
- [“Časově nezávislá komunikace”](#) na stránce 8
- [“Malé programy”](#) na stránce 8
- [“Zpracování řízené zprávami”](#) na stránce 8
- [“Zpracování řízené událostí”](#) na stránce 8
- [“Priorita zprávy”](#) na stránce 9
- [“Zabezpečení”](#) na stránce 9
- [“Integrita dat”](#) na stránce 9
- [“Podpora obnovy”](#) na stránce 9

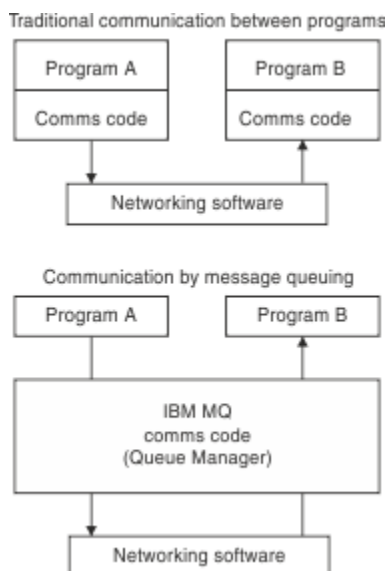
Poznámka: Při zvažování klientů a serverů IBM MQ nemusíte měnit serverovou aplikaci tak, aby podporovala další IBM MQ MQI clients na nových platformách. Podobně může IBM MQ MQI client beze změny fungovat s dalšími typy serverů.

Žádná přímá spojení mezi programy

Řazení zpráv do fronty je technika pro nepřímou komunikaci mezi programy. Lze jej použít v rámci libovolné aplikace, kde spolu programy komunikují. Komunikace probíhá prostřednictvím jednoho programu, který vkládá zprávy do fronty (ve vlastnictví správce front), a jiného programu, který tyto zprávy získává z fronty.

Programy mohou získat zprávy, které byly vloženy do fronty jinými programy. Ostatní programy mohou být připojeny ke stejnému správci front jako přijímající program nebo k jinému správci front. Tento jiný správce front může být v jiném systému, v jiném počítačovém systému nebo dokonce v jiném podniku či podniku.

Mezi programy, které komunikují pomocí front zpráv, nejsou žádná fyzická připojení. Program odesílá zprávy do fronty vlastněné správcem front a jiný program načítá zprávy z fronty (viz [Obrázek 1 na stránce 8](#)).



Obrázek 1. Řazení zpráv do fronty v porovnání s tradiční komunikací

Stejně jako v případě elektronické pošty se jednotlivé zprávy, které jsou součástí transakce, pohybují v síti obchodu-a-vpřed.základ. Pokud dojde k selhání propojení mezi uzly, zpráva se uchová, dokud se odkaz neobnoví, nebo operátor nebo program zprávu přesměruje.

Mechanismus, kterým se zpráva přesouvá z fronty do fronty, je před programy skrytý. Proto jsou programy jednodušší.

Časově nezávislá komunikace

Programy požadující práci ostatních nemusí čekat na odpověď na požadavek. Mohou provádět jinou práci a zpracovat odpověď, buď když dorazí, nebo později. Při psaní aplikace systému zpráv nemusíte vědět (nebo se vás to týká), když program odešle zprávu, nebo když je cíl schopen přijmout zprávu. Zpráva není ztracena; je uchovávána správcem front, dokud není cíl připraven ji zpracovat. Zpráva zůstane ve frontě, dokud ji program neodebere. To znamená, že programy odesílající a přijímající aplikace jsou odpojeny; odesílatel může pokračovat ve zpracování, aniž by čekal, až příjemce potvrdí přijetí zprávy. Cílová aplikace nemusí být ani spuštěna, když je zpráva odeslána. Může načíst zprávu poté, co byla spuštěna.

Malé programy

Řazení zpráv do fronty vám umožňuje využívat výhod používání malých, samostatných programů. Namísto jednoho velkého programu provádějícího všechny části úlohy postupně můžete úlohu rozdělit do několika menších nezávislých programů. Požadující program odesílá zprávy každému ze samostatných programů a žádá je o provedení jejich funkce; když je každý program dokončen, výsledky se odešlou zpět jako jedna nebo více zpráv.

Zpracování řízené zprávami

Když zprávy dorazí do fronty, mohou automaticky spustit aplikaci pomocí *spouštěče*. V případě potřeby lze aplikace zastavit po zpracování zprávy (nebo zpráv).

Zpracování řízené událostí

Programy lze řídit podle stavu front. Můžete například zajistit, aby se program spustil, jakmile zpráva dorazí do fronty, nebo můžete určit, že se program nespustí, dokud ve frontě nebude například 10 zpráv nad určitou prioritou, nebo 10 zpráv s libovolnou prioritou ve frontě.

Priorita zprávy

Program může přiřadit prioritu zprávě, když ji vloží do fronty. Určuje pozici ve frontě, na kterou je nová zpráva přidána.

Programy mohou získat zprávy z fronty buď v pořadí, ve kterém jsou zprávy ve frontě, nebo získáním specifické zprávy. (Program může chtít získat specifickou zprávu, pokud hledá odpověď na požadavek, který odeslal dříve.)

Zabezpečení

K dispozici jsou prostředky zabezpečení, včetně ověřování aplikací při použití správce front, kontroly autorizace při použití prostředků, jako je fronta ve správci front, a šifrování dat zpráv při jejich přenosu po síti a při jejich umístění ve frontách. Další informace o zabezpečení viz [Přehled zabezpečení](#).

Integrita dat

Integritu dat poskytují jednotky práce. Synchronizace začátku a konce pracovních jednotek je plně podporována jako volba pro každý příkaz MQGET nebo MQPUT, což umožňuje potvrdit nebo odvolat výsledky transakce. Podpora synchronizačních bodů pracuje buď interně, nebo externě s produktem IBM MQ v závislosti na formě koordinace synchronizačních bodů vybrané pro aplikaci.

Podpora obnovy

Aby byla obnova možná, jsou protokolovány všechny trvalé aktualizace produktu IBM MQ . Je-li zotavení nezbytné, jsou obnoveny všechny trvalé zprávy, všechny probíhající transakce jsou odvolány a všechna potvrzení a vrácení synchronizačního bodu jsou zpracována běžným způsobem správce SPM. Další informace o trvalých zprávách naleznete v tématu [Trvání zpráv](#).

Terminologie řazení zpráv do fronty

Tyto informace poskytují přehled o některých termínech používaných při řazení zpráv do fronty.

Mezi ně patří:

- [Kanály](#)
- [Klastr](#)
- [IBM MQ MQI client](#)
-  [Použití front v rámci skupiny](#)
- [Zpráva](#)
- [Agent oznamovacího kanálu](#)
- [Message Descriptor, Deskriptor zprávy](#)
- [dvoubodový](#)
- [Publikování/odběr](#)
- [Fronta](#)
- [Správce front](#)
-  [Skupina sdílení front](#)
-  [Sdílená fronta](#)
- [Odběr](#)
- [Téma](#)

Kanály

Kanály se používají k přesouvání zpráv z jednoho správce front do jiného a chrání aplikace před základními komunikačními protokoly. Správci front mohou existovat ve stejném systému, v různých systémech na stejné platformě nebo na různých platformách. Odeslané zprávy mohou pocházet z mnoha míst:

- Uživatelem napsané aplikační programy, které přenášejí data z jednoho uzlu do jiného.
- Aplikace administrace napsané uživatelem, které používají příkazy PCF nebo MQAI.
- Soubor IBM MQ Explorer.
- Správci front, kteří odesílají zprávy událostí instrumentace jinému správci front.
- Správci front, kteří odesílají příkazy vzdálené administrace jinému správci front. Například pomocí příkazů MQSC nebo administrative REST API.

Další informace o kanálech viz [“Definice kanálů”](#) na stránce 30.

Klastr

Klastr je síť správců front, kteří jsou nějakým způsobem logicky přidruženi.

V síti systému IBM MQ, která používá distribuované řazení do front bez klastrování, je každý správce front nezávislý. Pokud jeden správce front potřebuje odeslat zprávy jinému správci front, musí mít definovanou přenosovou frontu a kanál pro vzdáleného správce front.

Existují dva různé důvody pro použití klastrů: snížit administraci systému a zlepšit dostupnost a vyrovnávání pracovní zátěže.





Jakmile vytvoříte i ten nejmenší klastr, budete mít prospěch ze zjednodušené správy systému. Správci front, kteří jsou součástí klastru, potřebují méně definic, a proto je riziko výskytu chyby ve vašich definicích nižší.

Další informace o klastrování viz [Klastry](#).

IBM MQ MQI client

IBM MQ Klienti *rozhraní MQI* jsou nezávisle instalovatelné komponenty produktu IBM MQ. Klient MQI umožňuje spouštět aplikace IBM MQ s komunikačním protokolem, interaktivně spolupracovat s jedním či více servery rozhraní MQI (Message Queue Interface) na jiných platformách a připojovat se ke správcům front.

Úplné podrobnosti o instalaci a použití komponent produktu IBM MQ MQI client naleznete v následujících tématech:

-  [Instalace klienta IBM MQ na AIX](#)
-  [Instalace klienta IBM MQ na Linux®](#)
-  [Instalace klienta IBM MQ na Windows](#)
-  [Instalace klienta IBM MQ na IBM i](#)

a [Konfigurace připojení mezi serverem a klientem](#).

Použití front v rámci skupiny



Správci front ve skupině sdílení front mohou komunikovat pomocí normálních kanálů nebo můžete použít techniku nazvanou *řazení do front v rámci skupiny* (IGQ), která vám umožňuje provádět rychlý přenos zpráv bez definování kanálů. Toto platí pouze pro IBM MQ pro z/OS.

Další informace o řazení do front v rámci skupiny naleznete v tématu [“Použití front v rámci skupiny”](#) na stránce 205.

Zpráva

Ve frontě zpráv je zpráva kolekcí dat odeslaných jedním programem a určených pro jiný program. Viz [IBM MQ zprávy](#).

Informace o typech zpráv naleznete v tématu [Typy zpráv](#).

Agent oznamovacího kanálu

Agent kanálu zpráv je jedním koncem kanálu. Dvojice agentů kanálů zpráv, jeden odesílající a jeden přijímající, tvoří kanál a přesouvá zprávy z jednoho správce front do jiného.

Chcete-li získat informace o tom, jak se používají agenti kanálu zpráv, prohlédněte si téma [Úvod do distribuované správy front](#).

deskriptor zprávy

Zpráva IBM MQ se skládá z řídicích informací a dat aplikace.

Řídicí informace jsou definovány ve struktuře deskriptoru zpráv (MQMD) a obsahují následující položky:

- Typ zprávy
- Identifikátor pro zprávu
- Priorita pro doručení zprávy

Strukturu a obsah dat aplikace určují zúčastněné programy, nikoli IBM MQ.

Další informace viz [MQMD](#).

Dvoubodový systém zpráv

V systému zpráv typu point-to-point se každá zpráva pohybuje od jedné produkující aplikace k jedné spotřebovávající aplikaci. Zprávy jsou přenášeny prostřednictvím produkující aplikace vkládající zprávy do fronty a přijímající aplikace je získává z této fronty.

Systém zpráv publikování/odběru

V systému zpráv publikování/odběru je kopie každé zprávy publikované publikační aplikací doručena každé zainteresované aplikaci. Může existovat mnoho, jedna nebo žádná zainteresovaných aplikací. V publikování/odběru je zainteresovaná aplikace známa jako odběratel a zprávy jsou zařazeny do fronty identifikované odběrem.

Další informace viz téma [“Systém zpráv publikování/odběru”](#) na stránce 59.

Fronta

Pojmenované místo určení, do kterého lze odesílat zprávy. Zprávy se hromadí ve frontách, dokud je nenačtou programy, které tyto fronty obsluhují.

Další informace viz téma [“Fronty”](#) na stránce 19.

Správce front

Správce front je systémový program, který aplikacím poskytuje služby řazení do front.

Poskytuje rozhraní API (Application Programming Interface), které umožňuje programům vkládat zprávy do front a získávat je z nich. Správce front poskytuje další funkce, které umožňují administrátorům vytvářet nové fronty, měnit vlastnosti existujících front a řídit činnost správce front.

Má-li být služba řazení zpráv do fronty produktu IBM MQ v systému k dispozici, musí být spuštěn správce front. V jednom systému může být spuštěn více než jeden správce front (například pro oddělení

testovacího systému od *aktivního* systému). Pro aplikaci je každý správce front identifikován *manipulátor připojení (Hconn)*.

Služby správce front může využívat mnoho různých aplikací současně a tyto aplikace mohou být zcela nesouvisející. Aby mohl program používat služby správce front, musí vytvořit připojení k tomuto správci front.

Aby mohly aplikace odesílat zprávy aplikacím, které jsou připojeny k jiným správcům front, musí být tito správci front schopni komunikovat mezi sebou. Produkt IBM MQ implementuje protokol *store-and-forward*, který zajišťuje bezpečné doručování zpráv mezi těmito aplikacemi.

Další informace viz téma [“Správci front” na stránce 27.](#)

Skupina sdílení front



Správci front, kteří mají přístup ke stejné sadě sdílených front, tvoří skupinu s názvem *skupina sdílení front* (QSG). Vzájemně komunikují s prostředkem CF (coupling facility), který ukládá sdílené fronty. Toto platí pouze pro IBM MQ for z/OS.

Další informace viz téma [“Sdílené fronty a skupiny sdílení front” na stránce 161.](#)

Sdílená fronta



Sdílená fronta je typ lokální fronty se zprávami, ke kterým může přistupovat jeden nebo více správců front v prostředí sysplex. Nejedná se o stejnou frontu jako frontu sdílenou více než jednou aplikací s použitím stejného správce front. Toto platí pouze pro IBM MQ for z/OS.

Předplatné

Aplikace publikování/odběru může registrovat zájem o zprávy týkající se specifických témat. Když to aplikace provádí, je známa jako odběratel a termín odběru definuje, jak jsou odpovídající zprávy zařazeny do fronty pro zpracování.

Odběr obsahuje informace o identitě odběratele a identitě cílové fronty, do které mají být umístěna publikování. Obsahuje také informace o tom, jak má být publikování umístěno do cílové fronty.

Další informace viz téma [“Odběratelé a odběry” na stránce 62.](#)

Téma

Téma je znakový řetězec, který popisuje obsah informací publikovaných ve zprávách v rámci publikování/odběru.

Témata jsou klíčem k úspěšnému doručování zpráv v systému publikování/odběru. Namísto zahrnutí specifické cílové adresy do každé zprávy přiřadí vydavatel zprávě téma. Správce front ztotožní téma se seznamem odběratelů, kteří jsou přihlášení k jeho odběru, a doručí zprávu všem těmto odběratelům.

Další informace viz téma [“Témata” na stránce 64.](#)

Zprávy a fronty

Zprávy a fronty jsou základními komponentami systému front zpráv.

Co je to zpráva?

Zpráva je řetězec bajtů, který má význam pro aplikace, které jej používají. Zprávy se používají k přenosu informací z jednoho aplikačního programu do jiného (nebo mezi různými částmi stejné aplikace). Aplikace mohou být spuštěny na stejné platformě nebo na různých platformách.

Zpráva IBM MQ se skládá z:

- *Data aplikace.* Obsah a struktura dat aplikace jsou definovány aplikačními programy, které je používají.
- *Deskriptor zprávy.* Deskriptor zprávy identifikuje zprávu a obsahuje další řídicí informace, například typ zprávy a prioritu, kterou odesílající aplikaci přiřadila zprávě.

Formát deskriptoru zprávy je definován pomocí IBM MQ. Úplný popis deskriptoru zprávy viz [MQMD-deskriptor zprávy](#).

- *Vlastnosti zprávy.* Metadata o zprávě. Obsah vlastností zprávy jsou definovány aplikačními programy, které je používají. Další informace viz [Vlastnosti zprávy](#).

Délky zpráv

Výchozí maximální délka zprávy je 4 MB, i když ji můžete zvýšit na maximální délku 100 MB (kde 1 MB se rovná 1 048 576 bajtů). V praxi může být délka zprávy omezena:

- Maximální délka zprávy definovaná pro přijímací frontu
- Maximální délka zprávy definovaná pro správce front
- Maximální délka zprávy definovaná frontou
- Maximální délka zprávy definovaná odesílající nebo přijímající aplikací
- Množství úložiště, které je k dispozici pro zprávu

Odeslání všech informací, které aplikace vyžaduje, může trvat několik zpráv.

Jak aplikace odesílají a přijímají zprávy?

Aplikační programy odesílají a přijímají zprávy pomocí **volání MQI**.

Chcete-li například vložit zprávu do fronty, aplikace:

1. Otevře požadovanou frontu zadáním volání MQI MQOPEN .
2. Vydá volání MQI MQPUT pro vložení zprávy do fronty.

Jiná aplikace může načíst zprávu ze stejné fronty zadáním volání MQI MQGET .

Další informace o voláních MQI naleznete v tématu [Volání MQI](#).

Co je to fronta?

Fronta je datová struktura používaná k ukládání zpráv.

Každá fronta je vlastněna *správce front*. Správce front je zodpovědný za údržbu front, které vlastní, a za uložení všech přijatých zpráv do odpovídajících front. Zprávy mohou být vloženy do fronty aplikačními programy nebo správcem front jako součást jeho normálního provozu.

Předdefinované fronty a dynamické fronty

Fronty lze charakterizovat způsobem, jakým jsou vytvářeny:

- **Předdefinované fronty** vytváří administrátor pomocí příslušných příkazů MQSC nebo PCF. Předdefinované fronty jsou trvalé; existují nezávisle na aplikacích, které je používají, a přežijí IBM MQ restarty.
- **Dynamické fronty** jsou vytvořeny, když aplikace odešle požadavek MQOPEN s uvedením názvu *modelové fronty*. Vytvořená fronta je založena na *definici fronty šablony*, která se nazývá modelová fronta. Modelovou frontu můžete vytvořit pomocí příkazu MQSC DEFINE QMODEL. Atributy modelové fronty (například maximální počet zpráv, které v ní mohou být uloženy) jsou zděděny libovolnou dynamickou frontou, která je z ní vytvořena.

Modelové fronty mají atribut, který určuje, zda má být dynamická fronta trvalá nebo dočasná. Trvalé fronty přežívají po restartování aplikace a správce front; dočasné fronty jsou při restartu ztraceny.

Načítání zpráv z front

Vhodně autorizované aplikace mohou načítat zprávy z fronty podle následujících algoritmů načtení:

- První dovnitř-první ven (FIFO).
- Priorita zprávy, jak je definována v deskriptoru zprávy. Zprávy se stejnou prioritou se načítají na základě FIFO.
- Požadavek na program pro specifickou zprávu.

Použitá metoda je určena požadavkem MQGET z aplikace.

IBM MQObjekty

Správci front definují vlastnosti objektů IBM MQ . Hodnoty těchto vlastností ovlivňují způsob, jakým produkt IBM MQ tyto objekty zpracovává. Objekty vytváříte a spravujete pomocí příkazů a rozhraní IBM MQ . V aplikacích používáte rozhraní MQI (Message Queue Interface) k řízení objektů. Objekty jsou identifikovány IBM MQ *deskriptorem objektu* (MQOD) při adresování z programu.

Administrace objektů




Administrace objektů zahrnuje následující úlohy:

- Spouštění a zastavování správců front.
- Vytváření objektů, zejména front, pro aplikace.
- Zobrazení nebo změna atributů objektů.
- Odstraňování objektů.
- Práce s kanály při vytváření komunikačních cest ke správcům front v jiných (vzdálených) systémech.
- Vytvoření *klastrů* správců front za účelem zjednodušení celkového procesu administrace a vyvážení pracovní zátěže.

S výjimkou dynamických front musí být objekty definovány pro správce front, než s nimi budete moci pracovat.


Použijete-li příkaz IBM MQ k provedení operace administrace objektu, správce front zkontroluje, zda máte požadovanou úroveň oprávnění k provedení operace. Podobně platí, že pokud aplikace používá k otevření objektu volání MQOPEN, správce front zkontroluje, zda má aplikace požadovanou úroveň oprávnění, než povolí přístup k danému objektu. Kontroly se provádějí na názvu otevíraného objektu.

Objekty můžete definovat a spravovat pomocí následujících metod:

- Příkazy PCF popsané v tématu [Programovatelné formáty příkazů](#) a [Automatizace administrativních úloh](#) .
- Příkazy MQSC popsané v části [Příkazy MQSC](#)
-  Operace IBM MQ for z/OS a ovládací panely popsané v tématu [Administrace IBM MQ for z/OS](#) .
-   IBM MQ Explorer (Windows a Linux pouze pro systémy Intel). Další informace viz [Úvod do produktu MQ Explorer](#).

Objekty můžete také spravovat pomocí následujících metod:

- Řídící příkazy, které se zapisují z klávesnice. Viz [Administrace IBM MQ for Multiplatforms pomocí řídicích příkazů](#).
- Volání rozhraní IBM MQ Administration Interface (MQAI) v programu. Viz [IBM MQ Rozhraní administrace \(MQAI\)](#).

 V případě posloupností příkazů systému IBM MQ v systému AIX, Linux, and Windows můžete pomocí prostředku MQSC spustit posloupnost příkazů umístěných v souboru. Další informace naleznete v tématu [Administrace IBM MQ pomocí příkazů MQSC](#).

Pro posloupnosti příkazů IBM MQ for IBM i , které pravidelně používáte, můžete psát CL programy. Další informace naleznete v tématu [Správa IBM MQ for IBM i pomocí příkazů CL](#).

Pro posloupnosti příkazů IBM MQ for z/OS , které pravidelně používáte, můžete psát administrační programy, které vytvářejí zprávy obsahující příkazy a vkládají tyto zprávy do vstupní fronty systémových příkazů. Správce front zpracovává zprávy v této frontě stejným způsobem, jakým zpracovává příkazy zadané z příkazového řádku nebo z operací a ovládacích panelů. Tato technika je popsána v tématu [Psaní programů pro správu produktu IBM MQa demonstrována v ukázkové aplikaci Mail Manager dodané s produktem IBM MQ for z/OS](#). Popis této ukázky viz [Ukázkové programy pro IBM MQ for z/OS](#) .

Atributy objektů

Vlastnosti objektu jsou definovány jeho atributy. Některé můžete zadat, jiné můžete pouze zobrazit.

Například maximální délka zprávy, kterou může fronta pojmout, je definována atributem **MaxMsgLength** ; tento atribut můžete uvést při vytváření fronty. Atribut **DefinitionType** uvádí, jak byla fronta vytvořena; můžete zobrazit pouze tento atribut.

V souboru IBM MQ existují dva způsoby, jak odkazovat na atribut:

- Pomocí názvu PCF, například **MaxMsgLength**.
- Pomocí názvu příkazu MQSC, například MAXMSGL.

Skupiny sdílení front

Správci front, kteří mají přístup ke stejné sadě sdílených front, tvoří skupinu s názvem *skupina sdílení front* (QSG) a vzájemně komunikují pomocí prostředku CF (coupling facility), který ukládá sdílené fronty. Všimněte si, že QSG není striktně objekt.

Sdílená fronta je typ lokální fronty se zprávami, ke kterým může přistupovat jeden nebo více správců front, kteří jsou ve skupině sdílení front. Nejedná se o stejnou frontu jako frontu sdílenou více než jednou aplikací s použitím stejného správce front.

Názvy skupin sdílení front jsou tvořeny nejvýše čtyřmi znaky. Název musí být v síti jedinečný a nesmí být shodný s žádným názvem správce front.

Důležité: Sdílené fronty a skupiny sdílení front jsou podporovány pouze v systému IBM MQ for z/OS.

Další informace viz [“Sdílené fronty a skupiny sdílení front” na stránce 161](#).

Předvolené objekty systému

Výchozí systémové objekty jsou sadou definic objektů, které jsou vytvářeny automaticky při každém vytvoření správce front. Můžete zkopírovat a upravit libovolnou z těchto definic objektů pro použití v aplikacích ve vaší instalaci.

Výchozí názvy objektů mají kmenovou frontu SYSTEM; například výchozí lokální fronta je SYSTEM.DEFAULT.LOCAL.QUEUE a výchozí přijímací kanál je SYSTEM.DEF.RECEIVER. Tyto objekty nelze přejmenovat; výchozí objekty těchto názvů jsou povinné.

Když definujete objekt, všechny atributy, které neuvedete explicitně, se zkopírují z příslušného výchozího objektu. Pokud například definujete lokální frontu, atributy, které neurčíte, budou převzaty z výchozí fronty SYSTEM.DEFAULT.LOCAL.QUEUE.

Další informace viz [Systémové a výchozí objekty](#) .

Typy objektů

Mnoho úloh administrace zahrnuje manipulaci s různými typy produktu IBM MQ *objekty*.

Informace o pojmenování objektů IBM MQ viz [“Pojmenování objektů IBM MQ” na stránce 35](#).

Informace o výchozích objektech vytvořených ve správci front viz [“Předvolené objekty systému”](#) na stránce 15.

Chcete-li získat informace o různých typech objektů IBM MQ , prohlédněte si následující:

Objekty ověřovacích informací

Objekt ověřovacích informací poskytuje definice nezbytné k provedení kontroly odvolání certifikátu.

Objekt ověřovacích informací správce front tvoří součást podpory produktu IBM MQ pro protokol TLS (Transport Layer Security). Poskytuje definice potřebné ke kontrole odvolaných certifikátů. Certifikační autority odvolávají certifikáty, které již nemohou být důvěryhodné.

K definování objektu ověřovacích informací můžete použít příkaz MQSC **DEFINE AUTHINFO** . Další informace o attributech objektů ověřovacích informací viz **DEFINE AUTHINFO**.

S objektem ověřovacích informací můžete použít následující řídicí příkazy IBM MQ :

- **setmqaut** (udělit nebo odvolat oprávnění)
- **dspmqaut** (autorizace zobrazení objektu)
- **dmpmqaut** (autorizace výpisu paměti)
- **rcrmqobj** (znovu vytvořit objekt)
- **rcdmqing** (záznam obrazu média)
- **dspmqfls** (zobrazované názvy souborů)

Přehled protokolu TLS a použití objektů ověřovacích informací naleznete v tématu [Koncepty TLS \(Transport Layer Security\)](#) a [Protokoly zabezpečení TLS v produktu IBM MQ](#).

Kanály

Kanály jsou objekty, které poskytují komunikační cestu z jednoho správce front do jiného.

Další informace viz [“Kanály”](#) na stránce 28.

Objekty informací o komunikaci

Výběrové vysílání produktu IBM MQ nabízí spolehlivé výběrové vysílání zpráv s nízkou latencí a vysokým stupněm větvení. K tomu, aby bylo možné použít přenos výběrovým vysíláním, je vyžadován objekt informací o komunikaci (COMMINFO).

Další informace viz [“IBM MQ Výběrové vysílání”](#) na stránce 102.

Objekt COMMINFO je objekt IBM MQ , který obsahuje atributy přidružené k přenosu výběrového vysílání. Další informace o těchto attributech viz [DEFINE COMMINFO](#). Další informace o vytvoření objektu COMMINFO naleznete v tématu [Začínáme s výběrovým vysíláním](#).


Moduly listener

Moduly listener jsou procesy, které přijímají síťové požadavky od jiných správců front nebo klientských aplikací a spouštějí přidružené kanály.

Procesy modulu listener lze spustit pomocí řídicího příkazu **runmqlsru** .

Objekty modulu listener jsou IBM MQ objekty, které umožňují spravovat spouštění a zastavování procesů modulu listener v rámci oboru správce front. Definováním atributů objektu modulu listener provedete následující:

- Nakonfigurujte proces modulu listener.
- Určete, zda se má proces modulu listener automaticky spouštět a zastavovat při spuštění a zastavení správce front.

Důležité:  Objekty modulu listener nejsou v systému IBM MQ for z/OS podporovány. Další informace o tom, jak produkt IBM MQ for z/OS implementuje naslouchání pomocí inicializátoru kanálu, viz [“Inicializátor kanálu v systému z/OS” na stránce 157.](#)


Seznamy názvů

Seznam názvů je objekt IBM MQ, který obsahuje seznam názvů klastrů, názvů front nebo názvů objektů ověřovacích informací. V klastru lze použít k identifikaci seznamu klastrů, pro které správce front uchovává úložiště.

Seznam názvů je objekt IBM MQ, který obsahuje seznam dalších objektů IBM MQ. Obvykle se seznamy názvů používají v aplikacích, jako jsou monitory spouštěčů, k identifikaci skupiny front. Výhodou použití seznamu názvů je, že je udržován nezávisle na aplikacích. Lze jej aktualizovat bez zastavení aplikací, které jej používají. Pokud jedna aplikace selže, seznam názvů nebude ovlivněn a ostatní aplikace jej budou moci nadále používat.

Seznamy názvů se také používají s klastry správců front k udržování seznamu klastrů, na které odkazuje více než jeden objekt IBM MQ.

Seznamy názvů můžete definovat a upravovat pomocí příkazů MQSC [DEFINE NAMELIST](#) a [ALTER NAMELIST](#).

Poznámka:  Případně v systému z/OS můžete použít operace IBM MQ for z/OS a ovládací panely.

Programy mohou pomocí rozhraní MQI zjistit, které fronty jsou v těchto seznamech názvů obsaženy. Za organizaci seznamů názvů odpovídá návrhář aplikací a administrátor systému.

Seznam atributů seznamu názvů, které lze použít, naleznete v části [Atributy pro seznamy názvů](#).


Definice procesů

Objekty definice procesu umožňují spouštět aplikace bez nutnosti zásahu operátora definováním atributů aplikace pro použití správcem front.

Objekt definice procesu definuje aplikaci, která se spouští v reakci na událost spouštěče ve správci front IBM MQ. Atributy definice procesu zahrnují ID aplikace, typ aplikace a data specifická pro aplikaci. Další informace naleznete v tématu *Inicializační fronty* v části [“Fronty používané pro specifické účely produktem IBM MQ” na stránce 26.](#)

Chcete-li povolit spuštění aplikace bez nutnosti zásahu operátora, jak je popsáno v tématu [Spuštění aplikací IBM MQ pomocí spouštěčů](#), musí být atributy aplikace známy správci front. Tyto atributy jsou definovány v objektu *definice procesu*.

Atribut **ProcessName** je opraven při vytvoření objektu. Jiné atributy však můžete změnit pomocí příkazů IBM MQ.

Poznámka:  Případně v systému z/OS můžete použít operace IBM MQ for z/OS a ovládací panely.

Pomocí příkazu [MQINQ-Inquire object attributes](#) můžete zjišťovat hodnoty všech atributů.

Seznam atributů definice procesu, které lze použít, viz [Atributy pro definice procesu](#).

Fronty

IBM MQ *fronta* je pojmenovaný objekt, do kterého mohou aplikace vkládat zprávy a ze kterého mohou aplikace získávat zprávy.

Další informace viz [“Fronty” na stránce 19.](#)

Správci front

Správci front systému IBM MQ poskytují aplikacím služby řazení do front a spravují fronty, které k nim patří.

Další informace viz [“Správci front” na stránce 27.](#)

Služby

Objekty *Service* představují způsob, jak definovat programy, které mají být spuštěny při spuštění nebo zastavení správce front.

Programy mohou být jednoho z následujících typů:

Servery

Server je objekt služby, který má parametr *SERVTYPE* uvedený jako *SERVER*. Objekt služby serveru je definice programu, který bude spuštěn při spuštění určeného správce front. Souběžně lze provádět pouze jednu instanci procesu serveru. Za běhu lze stav procesu serveru monitorovat pomocí příkazu *MQSC DISPLAY SVSTATUS*. Obvykle jsou objekty služeb serveru definicemi programů, jako jsou obslužné rutiny nedoručených dopisů nebo monitory spouštěčů, avšak programy, které lze spustit, nejsou omezeny na ty, které jsou dodávány s produktem IBM MQ. Dále lze definovat objekt služby serveru tak, aby obsahoval příkaz, který bude spuštěn při vypnutí uvedeného správce front za účelem ukončení programu.

Příkazy

Příkaz je objekt služby, který má parametr *SERVTYPE* zadaný jako *COMMAND*. Objekt služby příkazů je definice programu, který bude spuštěn při spuštění nebo zastavení určeného správce front. Souběžně lze provádět více instancí procesu příkazu. Objekty služeb příkazů se liší od objektů služeb serveru v tom, že po spuštění programu nebude správce front program monitorovat. Objekty služeb příkazů jsou obvykle definice programů, které jsou krátkodobé a provedou specifickou úlohu, například spuštění jedné nebo více jiných úloh.

Důležité:  Objekty služeb nejsou v systému IBM MQ for z/OS podporovány.

Další informace viz [Práce se službami.](#)

Paměťové třídy



Paměťová třída mapuje jednu nebo více front na sadu stránek.

To znamená, že zprávy pro tuto frontu jsou uloženy (podléhají ukládání do vyrovnávací paměti) v dané sadě stránek.

Paměťové třídy jsou podporovány pouze v systému IBM MQ for z/OS.

Další informace o úložných třídách naleznete v tématu [Plánování prostředí IBM MQ na webu z/OS.](#)

Objekty tématu

Objekt tématu je objekt IBM MQ, který umožňuje přiřadit k tématům specifické, jiné než výchozí atributy.

Téma je definováno publikováním aplikace nebo přihlášením k odběru konkrétního *řetězce tématu*. Řetězec tématu může určovat hierarchii témat tak, že je oddělíte znakem dopředného lomítka (/). To lze vizualizovat pomocí *stromu témat*. Pokud například aplikace publikuje do řetězců témat */Sport/American Football* a */Sport/Soccer*, vytvoří se strom témat, který má nadřazený uzel *Sport* se dvěma podřazenými prvky *American Football* a *Soccer*.

Témata dědí své atributy od prvního nadřazeného administrativního uzlu nalezeného ve stromu témat. Pokud v konkrétním stromu témat nejsou žádné uzly administrativních témat, zdědí všechna témata své atributy od základního objektu tématu *SYSTEM.BASE.TOPIC*.

Objekt tématu můžete vytvořit v libovolném uzlu stromu témat určením řetězce tématu uzlu v atributu TOPICSTR objektu tématu. Můžete také definovat další atributy pro uzel administrativního tématu. Další informace o těchto atributech naleznete v tématu [Příkazy MQSC](#) nebo [Automatizace administrace pomocí příkazů PCF](#). Každý objekt tématu standardně dědí své atributy ze svého nejbližšího nadřazeného uzlu administrativního tématu.

Objekty témat lze také použít ke skrytí celého stromu témat před vývojáři aplikací. Pokud je pro téma /Sport/American Football vytvořen objekt tématu s názvem FOOTBALL.US, může aplikace publikovat nebo odebrat objekt s názvem FOOTBALL.US namísto řetězce /Sport/American Football se stejným výsledkem.

Zadáte-li znak #, +, / nebo * v řetězci tématu v objektu tématu, bude tento znak považován za normální znak v řetězci a bude považován za součást řetězce tématu přidruženého k objektu tématu.

Další informace o objektech tématu viz [“Systém zpráv publikování/odběru”](#) na stránce 59.

Související pojmy

[“Úvod do řazení zpráv do fronty”](#) na stránce 5

Produkty IBM MQ umožňují programům vzájemnou komunikaci v síti různých komponent (procesorů, operačních systémů, subsystémů a komunikačních protokolů) pomocí konzistentního rozhraní API.

Související odkazy

[Příkazy MQSC](#)

Fronty

Úvod do front IBM MQ a atributů fronty.

Zprávy jsou uloženy ve frontě, takže pokud vkládající aplikace očekává odpověď na svou zprávu, může při čekání na tuto odpověď provádět jinou práci. Aplikace přistupují k frontě pomocí rozhraní MQI (Message Queue Interface), které je popsáno v tématu [Přehled rozhraní fronty zpráv](#).

Před vložením zprávy do fronty musí být tato fronta již vytvořena. Fronta je vlastněna správcem front a tento správce front může vlastnit mnoho front. Každá fronta však musí mít název, který je v rámci tohoto správce front jedinečný.

Fronta je udržována prostřednictvím správce front. Ve většině případů je každá fronta fyzicky spravována svým správcem front, ale to není pro aplikační program zřejmé. IBM MQ for z/OS Sdílené fronty mohou být spravovány libovolným správcem front ve skupině sdílení front.

Chcete-li vytvořit frontu, můžete použít příkazy IBM MQ (MQSC), příkazy PCF nebo rozhraní specifická pro platformu. Operační a ovládací panely systému IBM MQ for z/OS jsou například specifické pro platformu.

Z aplikace můžete *dynamicky* vytvářet lokální fronty pro dočasné úlohy. Můžete například vytvořit fronty *reply-to* (které nejsou potřeba po ukončení aplikace). Další informace viz [“Dynamické a modelové fronty”](#) na stránce 24.

Před použitím fronty je nutné frontu otevřít a určit, co s ní chcete provést. Můžete například otevřít frontu pro:

- Procházení pouze zpráv (neprobíhá jejich načítání)
- Načítání zpráv (a buď sdílení přístupu s jinými programy, nebo s výlučným přístupem)
- Vkládání zpráv do fronty
- Dotazování na atributy fronty
- Nastavení atributů fronty

Úplný seznam voleb, které můžete zadat při otevření fronty, naleznete v tématu [MQOPEN-Otevřít objekt](#).

Atributy front

Některé atributy fronty jsou uvedeny, když je fronta definována, a nelze je později změnit (například typ fronty). Ostatní atributy front lze seskupit do těch, které lze změnit:

- Správcem front během zpracování fronty (například aktuální hloubka fronty).
- Pouze příkazy (například textový popis fronty)
- Podle aplikací pomocí volání MQSET (například zda jsou ve frontě povoleny operace vložení).

Hodnoty všech atributů můžete najít pomocí volání MQINQ.

Atributy, které jsou společné pro více než jeden typ fronty, jsou:

QName

Název fronty.

QTYPE

Typ fronty.

QDesc

Textový popis fronty.

InhibitGet

Zda mají programy povoleno získat zprávy z fronty. Zprávy ze vzdálených front však nelze nikdy získat.

InhibitPut

Zda mají programy povoleno vkládat zprávy do fronty.

DefPriority

Výchozí priorita pro zprávy vkládané do fronty.

DefPersistence

Výchozí perzistence pro zprávy vkládané do fronty

Obor

Řídí, zda položka pro tuto frontu také existuje ve službě názvů.



Atribut **Scope** není v systému z/OS podporován.

Úplný popis těchto atributů naleznete v tématu [Atributy pro fronty](#).

Způsoby definování front

Fronty do produktu IBM MQ můžete definovat pomocí příkazu MQSC DEFINE nebo pomocí příkazu PCF Create Queue . Příkazy určují typ fronty a její atributy. Například lokální objekt fronty má atributy, které určují, co se stane, když aplikace odkazují na tuto frontu ve voláních MQI. Příklady atributů jsou:

- Zda mohou aplikace načítat zprávy z fronty (volba GET povolena)
- Zda mohou aplikace vkládat zprávy do fronty (PUT povoleno)
- Zda je přístup k frontě výlučný pro jednu aplikaci nebo sdílený mezi aplikacemi
- Maximální počet zpráv, které mohou být uloženy ve frontě současně (maximální hloubka fronty)
- Maximální délka zpráv, které lze vložit do fronty

Existují také různá rozhraní specifická pro platformu, která můžete použít k definování front.

Související pojmy

“Fronty klastru” na stránce 55

Fronta klastru je fronta, jejímž hostitelem je správce front klastru, a která je dostupná ostatním správcům front v klastru.

“Fronty nedoručené-dopis” na stránce 47

Fronta nedoručených zpráv (nebo fronta nedoručených zpráv) je fronta, do které jsou odesílány zprávy, pokud je nelze směřovat na správné místo určení. Každý správce front má obvykle frontu nedoručených zpráv.

[Automatizace administrace pomocí příkazů PCF](#)

[Práce s frontami v produktu IBM MQ Console](#)

Související úlohy

[Administrace produktu IBM MQ pomocí příkazů MQSC](#)

Související odkazy

“Porovnání mezi sdílenými frontami a frontami klastru” na stránce 55

Tyto informace jsou navrženy tak, aby vám pomohly porovnat sdílené fronty a fronty klastru a rozhodnout se, které by mohly být pro váš systém vhodnější.

Související informace

“Co je to sdílená fronta?” na stránce 161

Lokální fronty

Přenosové, iniciační, nedoručené, příkazové, výchozí, kanálové a fronty událostí jsou typy lokální fronty.

Fronta je programu známa jako *lokální*, pokud ji vlastní správce front, ke kterému je program připojen. Do lokální fronty můžete vkládat zprávy a získávat je z ní.

Objekt definice fronty obsahuje informace o definici fronty spolu s fyzickými zprávami vloženými do této fronty.

Každý správce front může mít některé lokální fronty, které používá pro speciální účely:

Přenosové fronty

Když aplikace odešle zprávu do vzdálené fronty, lokální správce front uloží zprávu do speciální lokální fronty nazvané *přenosová fronta*. Aplikace mohou vkládat zprávy přímo do přenosové fronty nebo nepřímo prostřednictvím definice vzdálené fronty.

Když správce front odesílá zprávy vzdálenému správci front, identifikuje přenosovou frontu pomocí následující posloupnosti:

1. Přenosová fronta uvedená v atributu XMITQ lokální definice vzdálené fronty.
2. Přenosová fronta se stejným názvem jako vzdálený správce front. Tato hodnota je výchozí hodnotou v XMITQ lokální definice vzdálené fronty.
3. Přenosová fronta uvedená v atributu DEFXMITQ lokálního správce front.

Agent kanálu zpráv je program kanálu přidružený k přenosové frontě a doručí zprávu do dalšího cíle. Dalším cílem je správce front, ke kterému je kanál zpráv připojen. Nejedná se nutně o stejného správce front jako konečné místo určení zprávy. Když je zpráva doručena do dalšího místa určení, je odstraněna z přenosové fronty. Je možné, že zpráva bude muset při cestě do konečného místa určení projít mnoha správci front. Musíte definovat přenosovou frontu v každém správci front na trase, přičemž každá obsahuje zprávy čekající na přenos do dalšího místa určení. Normální přenosová fronta uchovává zprávy pro další místo určení, ačkoli zprávy mohou mít různá případná místa určení. Přenosová fronta klastru obsahuje zprávy pro více cílů. `correlID` každé zprávy identifikuje kanál, na kterém je zpráva umístěna, aby ji přenesli do dalšího místa určení.

Ve správci front můžete definovat několik přenosových front. Můžete definovat několik přenosových front pro stejné místo určení, přičemž každá z nich se používá pro jinou provozní třídu. Můžete například vytvořit různé přenosové fronty pro malé zprávy a velké zprávy, které budou směřovat do stejného místa určení. Poté můžete přenést zprávy pomocí různých kanálů zpráv, aby velké zprávy nezadržely menší zprávy. Všechny zprávy do front klastru nebo témat klastru jsou standardně umístěny do jedné přenosové fronty klastru `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Jako volbu můžete změnit výchozí nastavení a oddělit přenos zpráv do různých správců front klastru do různých přenosových front klastru. Nastavíte-li atribut správce front `DEFCLXQ` na hodnotu `CHANNEL`, vytvoří každý odesílací kanál klastru samostatnou přenosovou frontu klastru. Jako alternativu můžete ručně definovat přenosové fronty klastru pro odesílací kanály klastru, které se mají použít.

Přenosové fronty mohou spustit agenta kanálu zpráv, aby odesílal zprávy dále; viz [Spuštění IBM MQ aplikací pomocí spouštěčů](#).



Pokud v systému IBM MQ for z/OS používáte řazení do front v rámci skupiny, je přenosová fronta obsluhována *agentem front v rámci skupiny*. Sdílená přenosová fronta se používá při použití front v rámci skupiny v systému IBM MQ for z/OS.

Inicializační fronty

Inicializační fronta je lokální fronta, do které správce front při výskytu události spouštěče ve frontě aplikací vloží zprávu spouštěče.

Spouštěcí událost je událost, která má způsobit, že program spustí zpracování fronty. Například, událost může být více než 10 příchozích zpráv. Další informace o tom, jak spouštění funguje, naleznete v tématu [Spuštění IBM MQ aplikací pomocí spouštěčů](#).

Fronta nedoručených zpráv (nedoručených zpráv)

Fronta nedoručených zpráv je lokální fronta, do které správce front vkládá zprávy, které nemůže doručit.

Když správce front vloží zprávu do fronty nedoručených zpráv, přidá do zprávy záhlaví. Informace v záhlaví zahrnují důvod, proč správce front vložil zprávu do fronty nedoručených zpráv. Obsahuje také místo určení původní zprávy, datum a čas, kdy správce front vložil zprávu do fronty nedoručených zpráv.

Aplikace mohou také používat frontu pro zprávy, které nemohou doručit. Další informace naleznete v tématu [Použití fronty nedoručených zpráv](#).

Fronta systémových příkazů

Fronta systémových příkazů je fronta, do které mohou vhodně autorizované aplikace odesílat příkazy systému IBM MQ . Tyto fronty přijímají příkazy PCF, MQSC a CL, jak jsou podporovány na vaší platformě, a jsou připraveny na to, aby je správce front provedl.



V systému IBM MQ for z/OS se fronta nazývá `SYSTEM.COMMAND.INPUT` ; na jiných platformách se nazývá `SYSTEM.ADMIN.COMMAND.QUEUE`. Přijaté příkazy se liší podle platformy. Podrobnosti lze najít v tématu [Odkaz na formáty programovatelných příkazů](#).

Výchozí systémové fronty

Výchozí systémové fronty obsahují počáteční definice front pro váš systém. Při vytváření definice fronty správce front zkopíruje definici z příslušné výchozí fronty systému. Vytvoření definice fronty se liší od vytvoření dynamické fronty. Definice dynamické fronty je založena na modelové frontě, kterou zvolíte jako šablonu pro dynamickou frontu.

fronty událostí

Fronty událostí zadržují zprávy událostí. Tyto zprávy jsou hlášeny správcem front nebo kanálem.

Vzdálené fronty

Pro program je fronta *vzdálená* , pokud je vlastněna jiným správcem front, než je správce front, ke kterému je program připojen.

Tam, kde bylo ustanoveno komunikační spojení, může program odeslat zprávu do vzdálené fronty. Program nikdy nemůže získat zprávu ze vzdálené fronty.

Objekt definice fronty vytvořený při definování vzdálené fronty obsahuje pouze informace, které lokální správce front potřebuje k vyhledání fronty, do které má zpráva přejít. Tento objekt je znám jako *lokální definice vzdálené fronty*. Všechny atributy vzdálené fronty jsou zadrženy správcem front, který ji vlastní, protože se jedná o lokální frontu pro tohoto správce front.

Chcete-li při otevírání vzdálené fronty identifikovat frontu, musíte zadat jednu z následujících možností:

- Název lokální definice, která definuje vzdálenou frontu. Z hlediska aplikace se jedná o otevření lokální fronty. Aplikace nemusí vědět, zda je fronta lokální nebo vzdálená.

Chcete-li vytvořit lokální definici vzdálené fronty na všech platformách kromě platformy IBM i, použijte příkaz `DEFINE QREMOTE`.

 V systému IBM i použijte příkaz `CRTMQMQ`.

- Název vzdáleného správce front a název fronty, jak je znám tomuto vzdálenému správci front.

Lokální definice vzdálených front mají kromě obecných atributů popsaných v části [“Atributy front”](#) na stránce 19 tři atributy. Tyto tři atributy jsou:

RemoteQName

Název, podle kterého jej zná vlastní správce front.

RemoteQmgrName

Název vlastního správce front.

XmitQName

Název lokální přenosové fronty, která se používá při předávání zpráv jiným správcům front.

Další informace o těchto attributech naleznete v tématu [Atributy pro fronty](#).


Pokud použijete volání `MQINQ` pro lokální definici vzdálené fronty, vrátí správce front pouze atributy lokální definice, tj. název vzdálené fronty, název vzdáleného správce front a název přenosové fronty, nikoli atributy odpovídající lokální fronty ve vzdáleném systému.

Viz také [Přenosová fronta](#).

Alias fronty

Alias fronta je objekt IBM MQ, který můžete použít pro přístup k jiné frontě nebo tématu. To znamená, že více než jeden program může pracovat se stejnou frontou a přistupovat k ní pomocí různých názvů.

Fronta, která je výsledkem rozlišení názvu aliasu známého jako základní fronta, může se jednat o kterýkoli z následujících typů front podporovaných platformou:

- Lokální fronta
- Lokální definice vzdálené fronty.
-  Sdílená fronta, což je typ lokální fronty, která je k dispozici pouze v systému IBM MQ for z/OS.
- Předdefinovaná fronta
- Dynamická fronta

Název aliasu může být také převedený na téma. Pokud aplikace v současné době vkládá zprávy do fronty, lze ji nastavit tak, aby se publikovala v tématu, a to tak, že se název fronty stane aliasem pro dané téma. Není nutná žádná změna kódu aplikace.

Poznámka: Alias nelze přímo převést na jiný alias ve stejném správci front.

Příkladem použití alias front je, že administrátor systému poskytne různá přístupová oprávnění k názvu základní fronty (tj. k frontě, do které se alias převádí) a k názvu alias fronty. To znamená, že program nebo uživatel může mít oprávnění používat alias frontu, ale ne základní frontu.

Alternativně lze autorizaci nastavit tak, aby blokovala operace vložení pro název aliasu, ale povolila je pro základní frontu.

V některých aplikacích znamená použití alias front, že administrátoři systému mohou snadno změnit definici objektu alias fronty, aniž by museli měnit aplikaci.

Produkt IBM MQ provádí kontroly autorizace s aliasem, když se programy pokusí použít tento název. Nekontroluje, zda je program autorizován pro přístup k názvu, na který se alias interpretuje. Program proto může být autorizován pro přístup k názvu alias fronty, ale ne k rozlišovanému názvu fronty.

Kromě obecných atributů fronty popsaných v tématu “Fronty” na stránce 19 mají alias fronty atribut **BaseQName**. Jedná se o název základní fronty, na kterou se alias převádí. Podrobnější popis tohoto atributu viz [BaseQName \(MQCHAR48\)](#).

Atributy *InhibitGet* a **InhibitPut** (viz “Fronty” na stránce 19) alias front patří k názvu aliasu. Pokud se například název alias-fronty ALIAS1 interpretuje jako název základní fronty BASE, blokování na ALIAS1 ovlivní pouze ALIAS1 a BASE nebude blokováno. Inhibice na BASE však také ovlivňují ALIAS1.

Atributy *DefPriority* a **DefPersistence** také patří k názvu aliasu. Můžete například přiřadit různé výchozí priority různým aliasům stejné základní fronty. Také můžete změnit tyto priority, aniž byste museli měnit aplikace, které používají aliasy.


Dynamické a modelové fronty

Tyto informace poskytují náhled na dynamické fronty, vlastnosti dočasných a trvalých dynamických front, použití dynamických front, některé aspekty při používání dynamických front a modelové fronty.

Když aplikační program vydá volání MQOPEN za účelem otevření modelové fronty, správce front dynamicky vytvoří instanci lokální fronty se stejnými atributy jako modelová fronta. V závislosti na hodnotě pole *DefinitionType* modelové fronty vytvoří správce front dočasnou nebo trvalou dynamickou frontu (viz [Vytvoření dynamických front](#)).

Vlastnosti dočasných dynamických front

Dočasné dynamické fronty mají následující vlastnosti:

-  Nemohou se jednat o sdílené fronty, přístupné ze správců front ve skupině sdílení front.
 - Všimněte si, že skupiny sdílení front jsou k dispozici pouze v systému IBM MQ for z/OS.
- Obsahují pouze přechodné zprávy.
- Jsou neopravitelné.
- Jsou odstraněny při spuštění správce front.
- Jsou odstraněny, když aplikace, která vydala volání MQOPEN, která vytvořila frontu, frontu zavře nebo ukončí.
 - Pokud jsou ve frontě nějaké potvrzené zprávy, jsou odstraněny.
 - Pokud v této chvíli existují nepotvrzená volání MQGET, MQPUT nebo MQPUT1 nevyřízená vůči frontě, je fronta označena jako logicky odstraněná a je pouze fyzicky odstraněna (po potvrzení těchto volání) jako součást zpracování zavření nebo při ukončení aplikace.
 - Je-li fronta v tuto chvíli používána (vytvářením nebo jinou aplikací), je označena jako logicky odstraněná a je fyzicky odstraněna pouze v případě, že byla zavřena poslední aplikací používající frontu.
 - Pokusy o přístup k logicky odstraněné frontě (jiné než k jejímu zavření) selhaly s kódem příčiny MQRC_Q_DELETED.
 - S MQCO_NONE, MQCO_DELETE a MQCO_DELETE_PURGE se zachází jako s MQCO_NONE, pokud jsou zadány ve volání MQCLOSE pro odpovídající volání MQOPEN, které vytvořilo frontu.

Vlastnosti trvalých dynamických front

Trvalé dynamické fronty mají následující vlastnosti:

- Obsahují trvalé nebo přechodné zprávy.
- Jsou obnovitelné v případě selhání systému.
- Jsou odstraněny, když aplikace (ne nutně ta, která vydala volání MQOPEN, jež vytvořilo frontu) úspěšně zavře frontu pomocí volby MQCO_DELETE nebo MQCO_DELETE_PURGE.
 - Požadavek na zavření s volbou MQCO_DELETE se nezdaří, pokud jsou ve frontě stále nějaké zprávy (potvrzené nebo nepotvrzené). Požadavek na zavření s volbou MQCO_DELETE_PURGE uspěje i v případě, že fronta obsahuje potvrzené zprávy (zprávy odstraňované jako součást

zavření), ale nezdaří se, pokud pro danou frontu existují nepotvrzená volání MQGET, MQPUT nebo MQPUT1 .

- Pokud je požadavek na odstranění úspěšný, ale fronta je používána (vytvářením nebo jinou aplikací), je fronta označena jako logicky odstraněná a je fyzicky odstraněna pouze, když je uzavřena poslední aplikací používající frontu.
- Nejsou odstraněny, pokud jsou uzavřeny aplikací, která nemá oprávnění k odstranění fronty, pokud uzavírající aplikace nevydala volání MQOPEN, které vytvořilo frontu. Kontroly autorizace jsou prováděny vůči identifikátoru uživatele (nebo alternativnímu identifikátoru uživatele, pokud byl zadán parametr MQOO_ALTERNATE_USER_AUTHORITY), který byl použit k ověření odpovídajícího volání MQOPEN.
- Lze je odstranit stejným způsobem jako normální frontu.

Použití dynamických front

Dynamické fronty můžete použít pro:

- Aplikace, které nevyžadují uchování front po ukončení aplikace.
- Aplikace, které vyžadují odpovědi na zprávy, které mají být zpracovány jinou aplikací. Tyto aplikace mohou dynamicky vytvářet frontu pro odpovědi otevřením modelové fronty. Aplikace klienta může například:
 1. Vytvořte dynamickou frontu.
 2. Zadejte jeho název do pole **ReplyToQ** struktury deskriptoru zpráv zprávy požadavku.
 3. Umístěte požadavek do fronty zpracovávané serverem.

Server pak může umístit zprávu odpovědi do fronty pro odpověď. Nakonec může klient zpracovat odpověď a zavřít frontu pro odpověď s volbou odstranění.

Aspekty při použití dynamických front

Při používání dynamických front zvažte následující body:

- V modelu klient-server musí každý klient vytvořit a používat vlastní dynamickou frontu pro odpovědi. Pokud je dynamická fronta pro odpověď sdílena mezi více než jedním klientem, může být odstranění fronty pro odpověď zpožděno, protože pro frontu existuje nepotvrzená aktivita nebo protože je fronta používána jiným klientem. Kromě toho může být fronta označena jako logicky odstraněná a nepřístupná pro následné požadavky rozhraní API (jiné než MQCLOSE).
- Pokud vaše aplikační prostředí vyžaduje, aby dynamické fronty byly sdíleny mezi aplikacemi, ujistěte se, že je fronta zavřená (s volbou odstranění) pouze v případě, že byla potvrzena veškerá aktivita vůči frontě. Toto by měl být poslední uživatel. To zajistí, že odstranění fronty nebude zpožděno, a minimalizuje dobu, po kterou je fronta nepřístupná, protože byla označena jako logicky odstraněná.

Modelové fronty

Modelová fronta je šablona definice fronty, kterou používáte při vytváření dynamické fronty.

Lokální frontu můžete vytvořit dynamicky z programu IBM MQ a pojmenovat modelovou frontu, kterou chcete použít jako šablonu pro atributy fronty. V tomto bodě můžete změnit některé atributy nové fronty. Nemůžete však změnit **DefinitionType**. Pokud například vyžadujete trvalou frontu, vyberte modelovou frontu s typem definice nastaveným na trvalou. Některé konverzační aplikace mohou používat dynamické fronty k ukládání odpovědí na své dotazy, protože po zpracování odpovědi pravděpodobně nemusí tyto fronty udržovat.

Název modelové fronty zadáváte v *deskriptoru objektu* (MQOD) volání MQOPEN. Pomocí atributů modelové fronty vytvoří správce front dynamicky lokální frontu.

Můžete zadat název (úplný) pro dynamickou frontu nebo kmen názvu (například ABC) a nechat správce front přidat do této fronty jedinečnou část, nebo můžete nechat správce front přiřadit úplný jedinečný název. Pokud správce front přiřadí název, vloží jej do struktury MQOD.

Nemůžete zadat volání MQPUT1 přímo do modelové fronty, ale můžete zadat MQPUT1 do dynamické fronty, která byla vytvořena otevřením modelové fronty.

Příkazy MQSET a MQINQ nelze zadat pro modelovou frontu. Při otevření modelové fronty s příkazem MQOOO_INQUIRE nebo MQOO_SET dojde k následnému volání MQINQ a MQSET pro dynamicky vytvořenou frontu.

Atributy modelové fronty jsou podmnožinou atributů lokální fronty. Podrobnější popis viz [Atributy pro fronty](#).

Fronty používané pro specifické účely produktem IBM MQ

Produkt IBM MQ používá některé lokální fronty pro specifické účely související s jeho provozem.

Tyto fronty musíte definovat dříve, než je bude moci produkt IBM MQ používat.

Inicializační fronty

Inicializační fronty jsou fronty, které se používají při spouštění. Správce front vloží zprávu spouštěče do inicializační fronty při výskytu události spouštěče. Událost spouštěče je logická kombinace podmínek, která je zjištěna správcem front. Událost spouštěče může být například generována, když počet zpráv ve frontě dosáhne předdefinované hloubky. Tato událost způsobí, že správce front vloží zprávu spouštěče do určené inicializační fronty. Tato zpráva spouštěče je načtena *monitorem spouštěče*, speciální aplikací, která monitoruje inicializační frontu. Monitor spouštěčů pak spustí aplikační program, který byl uveden ve zprávě spouštěče.

Má-li správce front používat spouštění, musí být pro tohoto správce front definována alespoň jedna inicializační fronta. Viz [Správa objektů pro spuštění](#), [runmqtrma](#) [Spuštění IBM MQ aplikací pomocí spouštěčů](#).

Přenosové fronty

Přenosové fronty jsou fronty, které dočasně ukládají zprávy určené pro vzdáleného správce front. Musíte definovat alespoň jednu přenosovou frontu pro každého vzdáleného správce front, do kterého má lokální správce front odesílat zprávy přímo. Tyto fronty se používají také ve vzdálené administraci; viz [Vzdálená administrace z lokálního správce front](#). Informace o použití přenosových front v distribuovaných frontách naleznete v tématu [IBM MQ techniky distribuovaných front](#).

Každý správce front může mít výchozí přenosovou frontu. Pokud správce front, který není součástí klastru, vloží zprávu do vzdálené fronty, je výchozí akcí použití výchozí přenosové fronty. Pokud existuje přenosová fronta se stejným názvem jako správce cílové fronty, zpráva se umístí do této přenosové fronty. Pokud existuje definice aliasu správce front, ve které se parametr **RQMNAME** shoduje s cílovým správcem front a je zadán parametr **XMITQ**, je zpráva umístěna do přenosové fronty s názvem **XMITQ**. Pokud neexistuje žádný parametr **XMITQ**, zpráva se umístí do lokální fronty uvedené ve zprávě.

Přenosové fronty klastru

Každý správce front v rámci klastru má přenosovou frontu klastru s názvem SYSTEM.CLUSTER.TRANSMIT.QUEUE a modelovou přenosovou frontu klastru SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. Definice těchto front jsou standardně vytvářeny při definování správce front. Je-li atribut správce front **DEFCLXQ** nastaven na hodnotu CHANNEL, bude pro každý vytvořený odesílací kanál klastru automaticky vytvořena trvalá přenosová fronta dynamického klastru. Fronty se nazývají SYSTEM.CLUSTER.TRANSMIT.Channe1Name. Přenosové fronty klastru můžete také definovat ručně.

Správce front, který je součástí klastru, odesílá zprávy v jedné z těchto front jiným správcům front, kteří jsou ve stejném klastru.

Během rozpoznávání názvů má přenosová fronta klastru přednost před výchozí přenosovou frontou a specifická přenosová fronta klastru má přednost před frontou SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Fronty nedoručené-dopis

Fronta nedoručených zpráv (nedoručených zpráv) je fronta, ve které jsou uloženy zprávy, které nelze směřovat do správných míst určení. Zprávu nelze směřovat, pokud je například cílová fronta plná. Zadaná fronta nedoručených zpráv se nazývá SYSTEM.DEAD.LETTER.QUEUE.

V případě distribuovaného řazení do front definujte frontu nedoručených zpráv pro každého příslušného správce front.

Fronty příkazů

Fronta příkazů `SYSTEM.ADMIN.COMMAND.QUEUE` je lokální fronta, do které mohou vhodně autorizované aplikace odesílat příkazy `MQSC` ke zpracování. Tyto příkazy jsou poté načteny komponentou `IBM MQ`, která se nazývá příkazový server. Příkazový server ověřuje příkazy, předává platné příkazy pro zpracování správcem front a vrací všechny odpovědi do příslušné fronty pro odpovědi.

Fronta příkazů je vytvořena automaticky pro každého správce front při vytvoření tohoto správce front.

Fronty pro odpovědi

Když aplikace odešle zprávu požadavku, aplikace, která přijme zprávu, může odeslat zprávu odpovědi odesílající aplikaci. Tato zpráva je vložena do fronty, která se nazývá fronta pro odpověď, což je obvykle lokální fronta pro odesílající aplikaci. Název fronty pro odpověď je určen odesílající aplikací jako součást deskriptoru zprávy.

fronty událostí

Události instrumentace lze použít k monitorování správců front nezávisle na aplikacích `MQI`.

Dojde-li k události přípravy nástrojů, správce front vloží zprávu události do fronty událostí. Tuto zprávu pak může číst aplikace monitorování, která může informovat administrátora nebo zahájit nějakou nápravnou akci, pokud událost označuje problém.

Poznámka: Události spouštěče se liší od událostí instrumentace. Události spouštěče nejsou způsobeny stejnými podmínkami a negenerují zprávy událostí.

Další informace o událostech instrumentace viz [Události instrumentace](#).

Správci front

Úvod do *správců front* a služeb řazení do front, které poskytují aplikacím.

Program musí mít připojení ke správci front, aby mohl používat služby tohoto správce front. Program může toto připojení vytvořit explicitně (pomocí volání `MQCONN` nebo `MQCONNX`) nebo může být připojení provedeno implicitně (to závisí na platformě a prostředí, ve kterém je program spuštěn).

Správce front `IBM MQ` zajišťuje následující akce:

- Atributy objektu se mění podle přijatých příkazů.
- Speciální události, jako např. události spouštěče nebo události přípravy nástrojů, jsou generovány, když jsou splněny příslušné podmínky.
- Zprávy jsou vloženy do správné fronty, jak požaduje aplikace, která provádí volání `MQPUT`. Aplikace je informována, pokud to nelze provést, a je uveden příslušný kód příčiny.

Každá fronta patří jednomu správci front a je pro tohoto správce front *lokální frontou*. Správce front, ke kterému je aplikace připojena, je pro tuto aplikaci *lokálním správcem front*. Pro aplikaci jsou fronty, které patří k lokálnímu správci front, lokálními frontami.

Vzdálená fronta je fronta, která patří jinému správci front. *Vzdálený správce front* je jakýkoli jiný správce front než lokální správce front. Vzdálený správce front může existovat ve vzdáleném počítači v síti nebo může existovat ve stejném počítači jako lokální správce front. Produkt `IBM MQ` podporuje více správců front ve stejném počítači.

Objekt správce front lze použít v některých voláních `MQI`. Můžete se například dotazovat na atributy objektu správce front pomocí volání `MQI MQINQ`.

Atributy správců front

Ke každému správci front je přidružena sada atributů (nebo vlastností), které definují jeho charakteristiky. Některé atributy správce front jsou při vytvoření opraveny; jiné můžete změnit pomocí příkazů `IBM MQ`. Pomocí volání `MQINQ` můžete zjistit hodnoty všech atributů, s výjimkou těch, které se používají pro šifrování `TLS` (Transport Layer Security).

Mezi pevné atributy patří:

- Název správce front
- Platforma, na které je spuštěn správce front (například Windows).
- Úroveň řídicích příkazů systému, které správce front podporuje
- Maximální priorita, kterou lze přiřadit zprávám zpracovávaným správcem front
- Název fronty, do které mohou programy odesílat příkazy IBM MQ
- Maximální délka zpráv, které může správce front zpracovat **z/OS** (opraveno pouze v IBM MQ for z/OS)
- Zda správce front podporuje synchronizaci při vložení programů a získání zpráv.

Atributy *changeable* zahrnují:

- Textový popis správce front
- Identifikátor znakové sady, kterou správce front používá pro znakové řetězce při zpracování volání MQI
- Časový interval, který správce front používá k omezení počtu zpráv spouštěče
- **z/OS** Časový interval, který správce front používá k určení, jak často mají být fronty skenovány pro zprávy s vypršenou platností (pouze IBM MQ for z/OS)
- Název fronty nedoručených zpráv (nedoručených zpráv) správce front
- Název výchozí přenosové fronty správce front
- Maximální počet otevřených popisovačů pro každé jedno připojení
- Povolení a zakázání různých kategorií hlášení událostí
- Maximální počet nepotvrzených zpráv v rámci pracovní jednotky

Správci front a správa pracovní zátěže

Můžete nastavit klastr správců front, který má pro stejnou frontu více než jednu definici (například správci front v klastru mohou být navzájem klony). Zprávy pro konkrétní frontu mohou být zpracovány libovolným správcem front, který je hostitelem instance fronty. Algoritmus správy pracovní zátěže rozhoduje, který správce front zpracuje zprávu, a rozloží tak pracovní zátěž mezi správce front. Další informace naleznete v tématu [Algoritmus správy pracovní zátěže klastru](#) .

Kanály

Kanál je logické komunikační propojení používané distribuovanými správci front mezi serverem IBM MQ MQI client a serverem IBM MQ nebo mezi dvěma servery IBM MQ .

Kanály se používají k přesouvání zpráv z jednoho správce front do jiného a chrání aplikace před základními komunikačními protokoly. Správci front mohou existovat ve stejném systému, v různých systémech na stejné platformě nebo na různých platformách. Odeslané zprávy mohou pocházet z mnoha míst:

- Uživatelem napsané aplikační programy, které přenášejí data z jednoho uzlu do jiného.
- Aplikace administrace napsané uživatelem, které používají příkazy PCF nebo MQAI.
- Soubor IBM MQ Explorer.
- Správci front, kteří odesílají zprávy událostí instrumentace jinému správci front.
- Správci front, kteří odesílají příkazy vzdálené administrace jinému správci front. Například pomocí příkazů MQSC nebo administrative REST API.

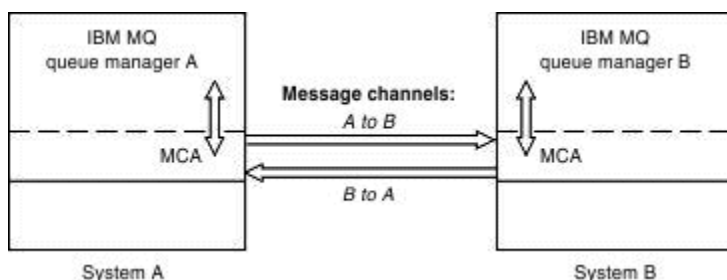
Kanál má dvě definice: jednu na každém konci připojení. Aby mohli správci front vzájemně komunikovat, je třeba definovat jeden objekt kanálu ve správci front, který má odesílat zprávy, a druhý, doplňkový, ve správci front, který je má přijímat. Na každém konci připojení musí být použit stejný *název kanálu* a použitý *typ kanálu* musí být kompatibilní.

V produktu IBM MQ existují tři kategorie kanálů s různými typy kanálů v rámci těchto kategorií:

- Kanály zpráv, které jsou jednosměrné, a přenos zpráv z jednoho správce front do jiného.
- Kanály MQI, které jsou obousměrné, přenášejí volání MQI z produktu IBM MQ MQI client do správce front a odezvy ze správce front do klienta IBM MQ .
- Kanály AMQP, které jsou obousměrné a připojují klienta AMQP ke správci front v počítači serveru. Produkt IBM MQ používá kanály AMQP k přenosu volání a odpovědí AMQP mezi aplikacemi AMQP a správci front.

Kanály zpráv

Účelem kanálu zpráv je přenos zpráv z jednoho správce front do jiného. Kanály zpráv nejsou vyžadovány prostředím klientského serveru.



Obrázek 2. Kanály zpráv mezi dvěma správci front

Kanál zpráv je jednosměrný odkaz. Chcete-li, aby vzdálený správce front odpovídal na zprávy odeslané lokálním správcem front, je třeba nastavit druhý kanál pro odesílání odpovědí zpět lokálnímu správci front.

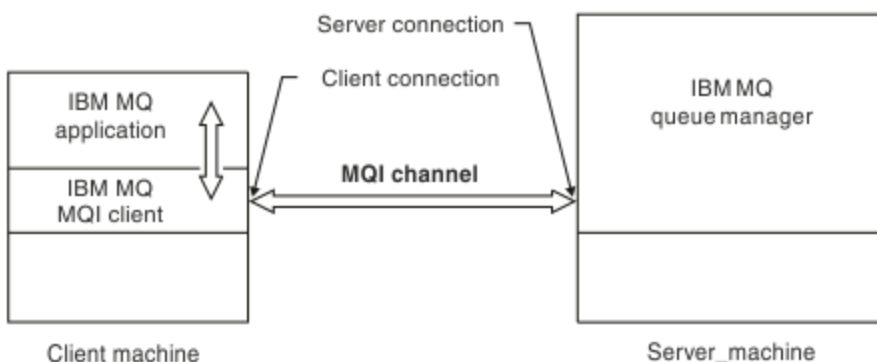
Kanál zpráv spojuje dva správce front pomocí *agentů kanálu zpráv* (MCA). Na každém konci kanálu je agent kanálu zpráv. Programu MCA můžete povolit přenos zpráv pomocí více podprocesů. Tento proces se nazývá *propojení procesů*. Pipelining umožňuje MCA efektivněji přenášet zprávy, což zlepšuje výkon kanálů. Další informace o propojení procesů viz [Atributy kanálů](#).

Další informace o kanálech naleznete v tématu [Volání a datové struktury ukončení kanálu](#) a ["Komponenty distribuovaného řazení do front"](#) na stránce 45.

Kanály MQI

Kanál rozhraní MQI (Message Queue Interface) připojuje IBM MQ MQI client ke správci front v počítači serveru a je vytvořen při volání MQCONN nebo MQCONNX z aplikace IBM MQ MQI client .

Jedná se o obousměrný odkaz a používá se pouze pro přenos volání MQI a odpovědí, včetně volání MQPUT obsahujících data zprávy a volání MQGET , jejichž výsledkem je vrácení dat zprávy. Existují různé způsoby vytváření a používání definic kanálů (viz [Definování kanálů MQI](#)).



Obrázek 3. Připojení klienta a připojení serveru v kanálu MQI



Kanál MQI lze použít k připojení klienta k jednomu správci front nebo ke správci front, který je součástí skupiny sdílení front (viz [Připojení klienta ke skupině sdílení front](#)).

Pro definice kanálu MQI existují dva typy kanálů. Definují obousměrný kanál MQI.

Kanál připojení klienta

Tento typ je pro IBM MQ MQI client.

Kanál připojení serveru

Tento typ je určen pro server, na kterém je spuštěn správce front, se kterým má komunikovat aplikace IBM MQ spuštěná v prostředí IBM MQ MQI client .

Kanály AMQP



K dispozici je pouze jeden typ kanálu AMQP.

Kanálem propojíte aplikaci systému zpráv AMQP se správcem front, aby si mohla aplikace vyměňovat zprávy s aplikacemi IBM MQ. Díky kanálu AMQP lze vyvinout aplikaci pomocí MQ Light a potom ji implementovat jako podnikovou aplikaci. Tím lze využít výhod zařízení úrovně enterprise poskytovaných produktem IBM MQ.

Kanály připojení klienta

Kanály připojení klienta jsou objekty, které poskytují komunikační cestu z IBM MQ MQI client do správce front.

Kanály připojení klienta se používají v distribuovaných frontách k přesouvání zpráv mezi správcem front a klientem. Chrání aplikace před základními komunikačními protokoly. Klient může existovat na stejné nebo jiné platformě ze správce front.

Definice kanálů

Popis jednotlivých typů kanálů naleznete v části [“Definice kanálů”](#) na stránce 30 .

Související pojmy

[“Distribuované řazení do front a klastry”](#) na stránce 41

Distribuované řazení do front znamená odesílání zpráv z jednoho správce front do jiného. Přijímající správce front může být na stejném počítači nebo jiném počítači; v blízkosti nebo na druhé straně světa. Může být spuštěn na stejné platformě jako lokální správce front nebo na libovolné platformě podporované produktem IBM MQ. Můžete ručně definovat všechna připojení v prostředí distribuovaných front, nebo můžete vytvořit klastr a nechat produkt IBM MQ definovat velkou část podrobností o připojení.

[Přehled rozhraní fronty zpráv](#)

Související úlohy

[Administrace vzdálených objektů IBM MQ](#)

[Zastavení kanálů MQI](#)

[Konfigurace připojení mezi serverem a klientem](#)

Související odkazy

[Volání uživatelské procedury kanálu a datové struktury](#)

[“Komunikace”](#) na stránce 34

Produkt IBM MQ MQI clients používá kanály MQI ke komunikaci se serverem.

Definice kanálů

Tabulky popisující různé typy kanálů zpráv a kanálů MQI, které produkt IBM MQ používá.

Mluví-li se o kanálech zpráv, je termín kanál často používán jako synonymum pro definici kanálu. Obvykle je z kontextu jasné, zda se jedná o celý kanál se dvěma konci nebo o definici kanálu, která má pouze jeden konec.

Kanály zpráv

Definice kanálů zpráv mohou být následujících typů:

Typ definice kanálu zpráv	Popis
Odesílatel	Termínem odesílací kanál je označován kanál zpráv, pomocí nějž správce front odesílá zprávy jiným správcům front. Chcete-li odesílat zprávy pomocí odesílacího kanálu, je nutné vytvořit pro partnerského správce front také přijímací kanál se shodným názvem, jaký má odesílací kanál. Pokud implementujete mechanismus "zpětného volání", můžete odesílací kanály také použít s žadatelskými kanály.
Server	Termínem kanál serveru je označován kanál zpráv, pomocí nějž správce front odesílá zprávy jiným správcům front. Chcete-li odesílat zprávy pomocí kanálu serveru, je nutné vytvořit pro partnerského správce front také přijímací kanál se shodným názvem, jaký má kanál serveru. Kanály serveru lze použít také spolu s žadatelskými kanály. V takovém případě vyžaduje definice žadatelského kanálu na druhém konci kanálu ke svému spuštění definici kanálu serveru. Server odesílá zprávy žadateli. Server může také inicializovat komunikaci, pokud má k dispozici údaje o názvu připojení partnerského kanálu.
Příjemce	Termínem přijímací kanál je označován kanál zpráv, pomocí nějž správce front přijímá zprávy od jiných správců front. Chcete-li přijímat zprávy pomocí přijímacího kanálu, je nutné vytvořit pro partnerského správce front také odesílací kanál nebo kanál serveru se shodným názvem, jaký má použitý přijímací kanál.
Žadatel	Termínem kanál žadatele je označován kanál zpráv, pomocí nějž správce front přijímá zprávy od jiných správců front. Kanál žadatele může požadovat spuštění partnerského kanálu definovaného na vzdáleném konci. Je-li partnerským kanálem kanál serveru, přijímá kanál serveru požadavek na spuštění a začne odesílat zprávy z přenosové fronty identifikované v definici kanálu serveru do kanálu žadatele. Je-li partnerským kanálem odesílací kanál, přijímá odesílací kanál požadavek na spuštění, ale pak zavře připojení se žadatelem. Odesílací kanál se pak spustí, vyjedná relaci s partnerským kanálem žadatele a začne odesílat zprávy z přenosové fronty identifikované v definici kanálu odesílatele. Tento druhý případ v podstatě poskytuje mechanismus zpětného volání v tom, že kanál žadatele požaduje, aby kanál odesílatele zavolal zpět.

Typ definice kanálu zpráv	Popis
Odesílatel klastru	Definice odesílacího kanálu klastru (CLUSDR) definuje odesílající konec kanálu, s jehož pomocí může správce front klastru odesílat informace o klastru do některého z úplných úložišť. Odesílací kanál klastru se používá k oznámení úložiště na veškeré změny stavu správce front, například na přidání nebo na odebrání fronty. Je používán také k přenosu zpráv. Samotní správci front mají odesílací kanály klastru, které jsou nasměrovány na sebe navzájem. Používají je ke vzájemné výměně změn stavu klastru. To, na které úplné úložiště odkazuje definice kanálu CLUSDR správce front, není příliš důležité. Po navázání počátečního kontaktu budou další objekty správce front klastru definovány automaticky podle potřeby tak, aby správce front mohl odesílat údaje o klastru do každého úplného úložiště a zprávy do každého správce front.
Příjemce klastru	Definice přijímacího kanálu klastru (CLUSRCVR) definuje přijímající konec kanálu, na kterém může správce front klastru přijímat zprávy od jiných správců front v klastru. Přijímací kanál klastru může také přenášet informace o klastru - údaje určené pro úložiště. Definováním přijímacího kanálu klastru dá správce front ostatním správcům front v klastru najevo, že je k dispozici pro příjem zpráv. Pro každého správce front klastru je nutný alespoň jeden přijímací kanál klastru.

Pro každý kanál je nutné definovat oba konce, aby každému konci kanálu odpovídala definice kanálu. Oba konce kanálu musí být kompatibilního typu.

Možné jsou následující kombinace definic kanálů:

- Odesílací-přijímací.
- Serveru-přijímací.
- Žadatelský-serveru.
- Žadatelský-odesílací (zpětné volání).
- Klastru-odesílací-klastru-přijímací.

Agenti kanálů zpráv

Každá vytvořená definice kanálu náleží k určitému správci front. Správce front může mít několik kanálů stejného typu nebo různých typů. Každému konci kanálu je přiřazen program označovaný jako agent kanálu zpráv (MCA). Na jednom konci kanálu přejímá volající agent MCA zprávy z přenosové fronty a odesílá je prostřednictvím kanálu. Na druhém konci kanálu tyto zprávy přijímá agent MCA odezvy a doručuje je do vzdáleného správce front.

Volajícího agenta MCA lze přidružit k odesílacímu kanálu, kanálu serveru nebo žadatelskému kanálu. Agent MCA odezvy lze přidružit ke kterémukoliv typu kanálu zpráv.

Produkt IBM MQ podporuje následující kombinace typů kanálů na obou koncích připojení:

Volající modul		Směr toku zpráv	Odpovídající	
Typ kanálu	Vyžadován modul listener?		Vyžadován modul listener?	Typ kanálu
Odesílatel	Ne	Od volajícího k odpovídajícímu	Ano	Příjemce
Server	Ne	Od volajícího k odpovídajícímu	Ano	Příjemce
Server	Ne	Od volajícího k odpovídajícímu	Ano	Žadatel
Žadatel	Ne	Od odpovídajícího k volajícímu	Ano	Server
Žadatel	Ano	Od odpovídajícího k volajícímu	Ano	Odesílatel

Kanály MQI

Kanály MQI mohou být některého z následujících typů:

Typ kanálu MQI	Popis
Připojení serveru	Kanál připojení serveru je obousměrný kanál MQI, který je používán pro připojení klienta IBM MQ k serveru IBM MQ. Kanál připojení serveru odpovídá konci kanálu v serveru.
Připojení klienta	Kanál připojení klienta je obousměrný kanál MQI, který je používán pro připojení klienta IBM MQ k serveru IBM MQ. Produkt IBM MQ Explorer používá připojení klienta také pro připojování ke vzdáleným správcům front. Kanál připojení klienta odpovídá konci kanálu v klientovi. Pokud vytvoříte kanál připojení klienta, bude v počítači, který je hostitelem správce front vytvořen odpovídající soubor. Poté je nutné zkopírovat tento soubor připojení klienta do počítače klienta produktu IBM MQ.

Podpora více podprocesů-propojení procesů

Volitelně můžete agentovi kanálu zpráv (MCA) povolit přenos zpráv pomocí více podprocesů. Tento proces, nazývaný *pipelining*, umožňuje agentovi MCA efektivněji přenášet zprávy s menším počtem stavů čekání, což zlepšuje výkon kanálu. Každý MCA je omezen na maximálně dva podprocesy.

Propojení procesů můžete řídit pomocí parametru *PipeLineLength* v souboru *qm.ini*. Tento parametr se přidává do sekce [Kanály](#).

Poznámka: Propojování procesů je účinné pouze pro kanály TCP/IP.

Používáte-li propojení procesů, musí být správci front na obou koncích kanálu nakonfigurováni tak, aby měli délku *PipeLine* větší než 1.

Aspekty uživatelské procedury kanálu

Propojení procesů může způsobit selhání některých uživatelských programů, protože:

- Uživatelské procedury nemusí být volány sériově.
- Uživatelské procedury mohou být volány střídavě z různých podprocesů.

Před použitím pipelingu zkontrolujte návrh svých výstupních programů:

- Východy musí být reentrant ve všech fázích jejich poprav.
- Při použití volání MQI nezapomeňte, že při vyvolání uživatelské procedury z různých podprocesů nelze použít stejný manipulátor MQI.





Zvažte uživatelskou proceduru zprávy, která otevře frontu a použije její manipulátor pro volání MQPUT při všech následných voláních uživatelské procedury. Tato operace selže v režimu propojení procesů, protože uživatelská procedura je volána z různých podprocesů. Chcete-li se tomuto selhání vyhnout, ponechte manipulátor fronty pro každý podproces a při každém vyvolání uživatelské procedury zkontrolujte identifikátor podprocesu.

Komunikace

Produkt IBM MQ MQI clients používá kanály MQI ke komunikaci se serverem.

Definice kanálu musí být vytvořena na konci IBM MQ MQI client i na konci serveru připojení. Postup vytvoření definic kanálů je vysvětlen v tématu [Definování kanálů MQI](#).

Možné přenosové protokoly jsou uvedeny v následující tabulce:

Platforma klienta	LU 6.2	TCP/IP	NetBIOS	SPX
 IBM i		Ano		
 Linux and Linux systémy  AIX	Ano ¹	Ano		
 Windows	Ano	Ano	Ano	Ano

Poznámka:

1.  Produkt LU6.2 není podporován na následujících platformách:

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

Přenosové protokoly-kombinace IBM MQ MQI client a serverových platform zobrazuje možné kombinace IBM MQ MQI client a serverových platform pomocí těchto přenosových protokolů.

Aplikace IBM MQ v systému IBM MQ MQI client může používat všechna volání MQI stejným způsobem, jako když je správce front lokální. **MQCONN** nebo **MQCONNX** přidruží aplikaci IBM MQ k vybranému správci front a vytvoří *manipulátor připojení*. Další volání používající tento manipulátor připojení jsou poté zpracována připojeným správcem front. Komunikace s produktem IBM MQ MQI client vyžaduje aktivní připojení mezi klientem a serverem, na rozdíl od komunikace mezi správcem front, která je nezávislá na připojení a nezávislá na čase.

Přenosový protokol je určen pomocí definice kanálu a neovlivňuje aplikaci. Aplikace Windows se může například připojit k jednomu správci front prostřednictvím protokolu TCP/IP a k jinému správci front prostřednictvím rozhraní NetBIOS.

Aspekty výkonu

Přenosový protokol, který používáte, může ovlivnit výkon systému klienta a serveru IBM MQ . V určitých situacích, kdy je přenos pomalý, můžete použít kompresi kanálu IBM MQ .

Pojmenování objektů IBM MQ


Konvence pojmenování přijatá pro objekty IBM MQ závisí na objektu. Na názvy počítačů a ID uživatelů, které používáte s produktem IBM MQ, se také vztahují některá omezení pojmenování.

Každá instance správce front je známa pod svým názvem. Tento název musí být jedinečný v rámci sítě propojených správců front, aby mohl jeden správce front jednoznačně identifikovat cílového správce front, do kterého je odeslána daná zpráva.

Pro ostatní typy objektů má každý objekt přidružený název a lze na něj tímto názvem odkazovat. Tyto názvy musí být v rámci jednoho správce front a typu objektu jedinečné. Můžete mít například frontu a proces se stejným názvem, ale nemůžete mít dvě fronty se stejným názvem.

V systému IBM MQ mohou mít názvy maximálně 48 znaků, s výjimkou *kanálů*, které mají maximálně 20 znaků. Další informace o pojmenování objektů IBM MQ viz [“Pravidla pro pojmenování objektů IBM MQ” na stránce 35](#).

Na názvy počítačů a ID uživatelů, které používáte s produktem IBM MQ, se také vztahují některá omezení pojmenování:

- Ujistěte se, že název počítače neobsahuje žádné mezery. Produkt IBM MQ nepodporuje názvy počítačů, které obsahují mezery. Pokud nainstalujete produkt IBM MQ na takový počítač, nebudete moci vytvořit žádné správce front.
- Pro autorizace systému IBM MQ nesmí být názvy ID uživatelů a skupin delší než 20 znaků (mezery nejsou povoleny).
-  Server IBM MQ for Windows nepodporuje připojení k serveru IBM MQ MQI client, pokud je klient spuštěn pod ID uživatele, které obsahuje znak @, například abc@d.

Související pojmy

[“IBM MQ NÁZVY SOUBORŮ” na stránce 38](#)

Každý objekt správce front, fronty, definice procesu, seznamu názvů, kanálu, kanálu připojení klienta, modulu listener, služby a ověřovacích informací produktu IBM MQ je reprezentován souborem. Vzhledem k tomu, že názvy objektů nejsou nutně platnými názvy souborů, správce front v případě potřeby převede název objektu na platný název souboru.

Související odkazy

[“Pravidla pro pojmenování objektů IBM MQ” na stránce 35](#)

Názvy objektů IBM MQ mají maximální délku a rozlišují velikost písmen. Ne všechny znaky jsou podporovány pro každý typ objektu a mnoho objektů má pravidla týkající se jedinečnosti názvů.

Pravidla pro pojmenování objektů IBM MQ

Názvy objektů IBM MQ mají maximální délku a rozlišují velikost písmen. Ne všechny znaky jsou podporovány pro každý typ objektu a mnoho objektů má pravidla týkající se jedinečnosti názvů.

Existuje mnoho různých typů objektů IBM MQ a objekty z každého typu mohou mít stejný název, protože existují v oddělených oborech názvů objektů: Například lokální fronta a kanál odesilatele mohou mít stejný název. Nicméně objekt nemůže mít stejný název jako jiný objekt ve stejném oboru názvů: například lokální fronta nemůže mít stejný název jako modelová fronta a kanál odesilatele nemůže mít stejný název jako kanál příjemce.

Následující objekty IBM MQ existují v oddělených prostorech jmen objektů:

- Ověřovací informace
- Kanál
- Kanál klienta
- Modul listener
- Seznam názvů
- Proces


- Fronta
- Služba
- Paměťová třída
- Předplatné
- Téma

Znaková délka názvů objektů

Obecně platí, že názvy objektů IBM MQ mohou mít délku až 48 znaků. Toto pravidlo platí pro následující objekty:

- Ověřovací informace
- Klastr
- Modul listener
- Seznam názvů
- Definice procesu
- Fronta
- Správce front
- Služba
- Předplatné
- Téma

Existují omezení:

1.  V systémech z/OS musí být správci front maximálně 4 znaky a musí být pouze velkými písmeny a číselnými znaky.
2. Maximální délka názvů objektů kanálu a názvů kanálů připojení klienta je 20 znaků. Další informace o kanálech viz [Definování kanálů](#).
3. Řetězce témat mohou být maximálně 10240 bajtů. Všechny názvy objektů IBM MQ rozlišují velikost písmen.
4. Názvy odběrů mohou být maximálně 10240 bajtů a mohou obsahovat mezery.
5. Maximální délka názvů paměťových tříd je 8 znaků.
6. Maximální délka názvů struktury prostředku CF je 12 znaků.

Znaky v názvech objektů

Platné znaky pro názvy objektů IBM MQ jsou:

Znaky	Omezení
Velká písmena A-Z	• Není

Znaky	Omezení
Malá písmena a-z	<ul style="list-style-type: none"> • Ve skriptech MQSC musí být názvy s malými písmeny uzavřeny v apostrofech. Tím zabráníte tomu, aby se malá písmena skládala na velká písmena. • Systémy používající EBCDIC Katakana nemohou v názvech objektů používat malá písmena a-z. • z/OS Při použití malých písmen v systémech z/OS mohou existovat omezení, například názvy správců front nemohou obsahovat malá písmena. • IBM i V systémech IBM i při použití příkazů CL musí být názvy s malými písmeny uzavřeny v apostrofech. Tím zabráníte tomu, aby se malá písmena skládala na velká písmena.
Čísla 0-9	<ul style="list-style-type: none"> • Není
Tečka (.)	<ul style="list-style-type: none"> • Není
Podtržítka (_)	<ul style="list-style-type: none"> • Multi Není • z/OS Vyvarujte se použití názvů s úvodními nebo koncovými podtržítka, protože nemohou být zpracovány operacemi IBM MQ for z/OS a ovládacími panely.
Lomítka (/)	<ul style="list-style-type: none"> • Windows V systémech Windows nemůže být prvním znakem názvu správce front dopředné lomítka. • IBM i V systémech IBM i při použití příkazů CL musí být názvy obsahující dopředné lomítka uzavřeny v apostrofech. • z/OS Není
Procento (%)	<ul style="list-style-type: none"> • ALW Není • z/OS Používáte-li produkt RACF jako externího správce zabezpečení pro produkt IBM MQ for z/OS, nepoužívejte v názvech objektů hodnotu%, protože tyto názvy nejsou zahrnuty do kontrol zabezpečení při použití generických profilů RACF . • IBM i V systémech IBM i při použití příkazů CL musí být názvy obsahující znak procenta uzavřeny v apostrofech.

Existují také některá obecná pravidla týkající se znaků v názvech objektů:

1. Vložené mezery ani mezery na začátku nejsou povoleny.
2. Znaky národního jazyka nejsou povoleny.
3. Jakýkoli název, který je menší než celá délka pole, může být zprava vyplněn mezerami. Všechny krátké názvy vrácené správcem front jsou vždy zprava vyplněny mezerami.

Názvy front

Název fronty má dvě části:

- Název správce front
- Lokální název fronty tak, jak je znám tomuto správci front

Každá část názvu fronty má délku 48 znaků.

Chcete-li odkazovat na lokální frontu, můžete vynechat název správce front (tím, že jej nahradíte prázdnými znaky nebo použijete úvodní znak null). Všechny názvy front vrácené programem produktem IBM MQ však obsahují název správce front.

z/OS Sdílená fronta, která je přístupná pro všechny správce front v příslušné skupině sdílení front, nemůže mít stejný název jako libovolná nesdílená lokální fronta ve stejné skupině sdílení front. Toto omezení zabraňuje možnosti, aby aplikace omylem otevřela sdílenou frontu, když má v úmyslu otevřít lokální frontu, nebo naopak. Sdílené fronty a skupiny sdílení front jsou k dispozici pouze v systému IBM MQ for z/OS.

Chcete-li odkazovat na vzdálenou frontu, musí program obsahovat název správce front v úplném názvu fronty nebo musí existovat lokální definice vzdálené fronty.

Pokud aplikace používá název fronty, může se jednat buď o název lokální fronty (nebo o alias k jedné), nebo o název lokální definice vzdálené fronty, ale aplikace nemusí vědět, která, pokud nepotřebuje získat zprávu z fronty (pokud musí být fronta lokální). Když aplikace otevře objekt fronty, provede volání MQOPEN funkci rozpoznávání názvů, která určí, ve které frontě se mají provést následné operace. Význam této skutečnosti spočívá v tom, že aplikace nemá žádnou vestavěnou závislost na konkrétních frontách, které jsou definovány v určitých umístěních v síti správců front. Proto, pokud administrátor systému přemístí fronty v síti a změní jejich definice, nemusí být aplikace, které tyto fronty používají, změněny.

Vyhrazené názvy objektů

Názvy objektů, které začínají na SYSTEM. , jsou vyhrazeny pro objekty definované správcem front. Pomocí příkazů **Alter**, **Define** a **Replace** můžete tyto definice objektů změnit tak, aby vyhovovaly vaší instalaci. Názvy definované pro produkt IBM MQ jsou uvedeny v plném rozsahu v poli [Názvy front](#).

z/OS V systému IBM MQ for z/OS je název struktury aplikace prostředku Coupling Facility CSQSYSAPPL vyhrazen.

Související pojmy

[Název instalace na systému AIX, Linux, and Windows](#)

IBM MQ NÁZVY SOUBORŮ

Každý objekt správce front, fronty, definice procesu, seznamu názvů, kanálu, kanálu připojení klienta, modulu listener, služby a ověřovacích informací produktu IBM MQ je reprezentován souborem. Vzhledem k tomu, že názvy objektů nejsou nutně platnými názvy souborů, správce front v případě potřeby převede název objektu na platný název souboru.

Výchozí cesta k adresáři správce front je následující:

- Předpona, která je definována v informacích o konfiguraci IBM MQ :
 - **Linux** **AIX** V systému AIX and Linux je výchozí předpona /var/mqm. Tato konfigurace je nakonfigurována v sekci DefaultPrefix konfiguračního souboru mqsc.ini .
 - **Windows** Na 32bitových systémech Windows je výchozí předpona C:\Program Files (x86)\IBM\WebSphere MQ. Na 64bitových systémech Windows je výchozí předpona C:\Program Files\IBM\MQ. Pro 32bitové i 64bitové instalace se datové adresáře instalují do adresáře C:\ProgramData\IBM\MQ. Tato konfigurace je nakonfigurována v sekci DefaultPrefix konfiguračního souboru mqsc.ini .

Je-li k dispozici, lze předponu změnit pomocí stránky vlastností IBM MQ v Průzkumníku IBM MQ , jinak upravte konfigurační soubor `mqsc.ini` ručně.

- Název správce front je převeden na platný název adresáře. Například správce front:

```
queue.manager
```

by byly zastoupeny jako:

```
queue!manager
```

Na tento proces se odkazuje jako na *transformaci názvu*.

V produktu IBM MQ můžete dát správci front název obsahující až 48 znaků.

Můžete například pojmenovat správce front:

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

Každý správce front je však reprezentován souborem a existují omezení maximální délky názvu souboru a znaků, které lze v názvu použít. V důsledku toho se názvy souborů reprezentujících objekty automaticky transformují tak, aby splňovaly požadavky systému souborů.

Pravidla, kterými se řídí transformace názvu správce front, jsou následující:

1. Transformovat jednotlivé znaky:
 - Z toho. Do které!
 - Od/do &
2. Pokud název stále není platný:
 - a. Oříznout na osm znaků
 - b. Připojit tříznakovou číselnou příponu

Předpokládejme například výchozí předponu a správce front s názvem `queue.manager`:

- **Windows** V systému Windows s NTFS nebo FAT32 se název správce front změní na:

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

- **Windows** V systému Windows s FAT se název správce front změní na:

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

- **Linux** **AIX** V systému AIX and Linux se název správce front změní na:

```
/var/mqm/mqmgs/queue!manager
```

Algoritmus transformace také rozlišuje mezi názvy, které se liší pouze velikostí písmen v systémech souborů, které nerozlišují velikost písmen.

Transformace názvu objektu

Názvy objektů nemusí být nutně platné názvy systémů souborů. Možná budete muset transformovat názvy objektů. Použitá metoda se liší od metody používané pro názvy správců front, protože ačkoli v každém počítači existuje pouze několik názvů správců front, pro každého správce front může existovat

velký počet dalších objektů. Fronty, definice procesů, seznamy názvů, kanály, kanály připojení klienta, moduly listener, služby a objekty ověřovacích informací jsou reprezentovány v systému souborů.

Když proces transformace vygeneruje nový název, neexistuje žádný jednoduchý vztah s původním názvem objektu. Příkaz **dspmqls** můžete použít k převodu mezi skutečnými a transformovanými názvy objektů.

Související odkazy

dspmqls (zobrazované názvy souborů)

Související informace

AllQueueSekce správců souboru mqz.ini

IBM i Názvy objektů na IBM i

Správce front má přidruženou knihovnu správců front, která má jedinečný název. Názvy správců front a názvy objektů může být nutné transformovat tak, aby splňovaly požadavky systému IBM i Integrated File System .

Při vytvoření správce front produkt IBM MQ přidruží k němu knihovnu správce front. Této knihovně správce front je udělen jedinečný název o délce nejvýše 10 znaků, který je z velké části založen na názvu správce front definovaném uživatelem. Správce front i knihovna správců front jsou umístěny do adresáře, který je také založen na názvu správce front s předponou /QIBM/UserData/mqm. Následuje příklad správce front, knihovny správce front a adresáře:

Název správce front	ORANŽOVÁ
Název knihovny správce front	QMORANGE
Adresář	/QIBM/UserData/mqm/ORANGE

Všechny názvy správců front a názvy knihoven správců front jsou zapsány do sekce v souboru /QIBM/UserData/mqm/mqs.ini.

IBM MQ Adresáře a soubory IFS

Produkt IBM i Integrated File System (IFS) používá produkt IBM MQ k ukládání dat ve velké míře. Další informace o IFS viz *Integrated File System*.

Každý objekt IBM MQ , například kanál nebo správce front, je reprezentován souborem. Vzhledem k tomu, že názvy objektů nejsou nutně platnými názvy souborů, správce front v případě potřeby převede název objektu na platný název souboru.

Cesta k adresáři správce front je vytvořena z následujícího:

- Předpona, která je definována v konfiguračním souboru správce front `qm.ini`. Výchozí předpona je /QIBM/UserData/mqm.
- Literál, `qmgrs`.
- Kódovaný název správce front, což je název správce front převedený na platný název adresáře. Například správce front `queue/manager` je reprezentován `queue&manager`.

Na tento proces se odkazuje jako na transformaci názvu.

Transformace názvu správce front IFS

V produktu IBM MQ můžete dát správci front název obsahující až 48 znaků.

Můžete například pojmenovat správce front `QUEUE/MANAGER/ACCOUNTING/SERVICES`. Stejným způsobem, jako je vytvořena knihovna pro každého správce front, je každý správce front také reprezentován souborem. Kvůli variantním identifikačním bodům v EBCDIC existují omezení znaků, které lze použít v názvu. V důsledku toho se názvy souborů IFS, které představují objekty, automaticky transformují tak, aby splňovaly požadavky systému souborů.

Při použití příkladu správce front s názvem `queue/manager`, transformaci znaku `/` na `&a` za předpokladu výchozí předpony se název správce front v souboru IBM MQ for IBM i změní na `/QIBM/UserData/mqm/qmgrs/queue&manager`.

Transformace názvu objektu

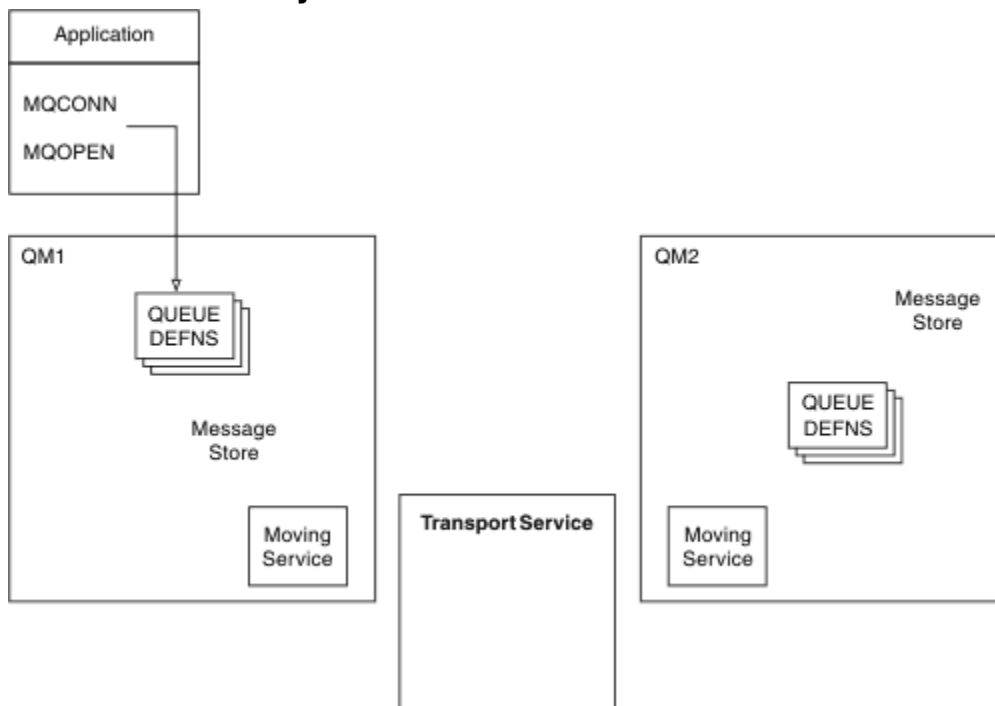
Názvy objektů nemusí být nutně platnými názvy systémů souborů, takže názvy objektů může být nutné transformovat. Použitá metoda se liší od metody pro názvy správců front, protože ačkoli pro každý počítač existuje pouze několik názvů správců front, pro každého správce front může existovat velký počet dalších objektů. V systému souborů jsou reprezentovány pouze definice procesů, fronty a seznamy názvů. Kanály nejsou těmito aspekty ovlivněny.

Když proces transformace vygeneruje nový název, neexistuje žádný jednoduchý vztah s původním názvem objektu. K zobrazení transformovaných názvů objektů IBM MQ můžete použít příkaz `DSPMQMOBJN`.

Distribuované řazení do front a klastry

Distribuované řazení do front znamená odesílání zpráv z jednoho správce front do jiného. Přijímající správce front může být na stejném počítači nebo jiném počítači; v blízkosti nebo na druhé straně světa. Může být spuštěn na stejné platformě jako lokální správce front nebo na libovolné platformě podporované produktem IBM MQ. Můžete ručně definovat všechna připojení v prostředí distribuovaných front, nebo můžete vytvořit klastr a nechat produkt IBM MQ definovat velkou část podrobností o připojení.

distribuované fronty



Obrázek 4. Přehled komponent distribuovaných front

V předchozím obrázku:

- Aplikace používá volání `MQCONN` pro připojení ke správci front. Aplikace poté pomocí volání `MQOPEN` otevře frontu, aby do ní mohla vkládat zprávy.
- Každý správce front má definici pro každou ze svých front. Může obsahovat definice *lokálních front* (tj. front, jejichž hostitelem je tento správce front) a definice *vzdálených front* (tj. front, jejichž hostitelem jsou jiní správci front).
- Pokud jsou zprávy určeny pro vzdálenou frontu, lokální správce front je uchovává v *přenosové frontě*, která je uchovává v úložišti zpráv, dokud je nelze předat vzdálenému správci front.

- Každý správce front obsahuje komunikační software, označovaný jako *služba přesunu*, který správce front používá ke komunikaci s ostatními správci front.
- *Služba přenosu* je nezávislá na správci front a může se jednat o některou z následujících položek (v závislosti na platformě):
 - Rozšířená komunikace SNA APPC (Systems Network Architecture Advanced Program-to-Program Communication)
 - Transmission Control Protocol/Internet Protocol (TCP/IP)
 - Síťový základní vstupní/výstupní systém (NetBIOS)
 - Sekvenční výměna paketů (SPX)

Komponenty potřebné k odeslání zprávy

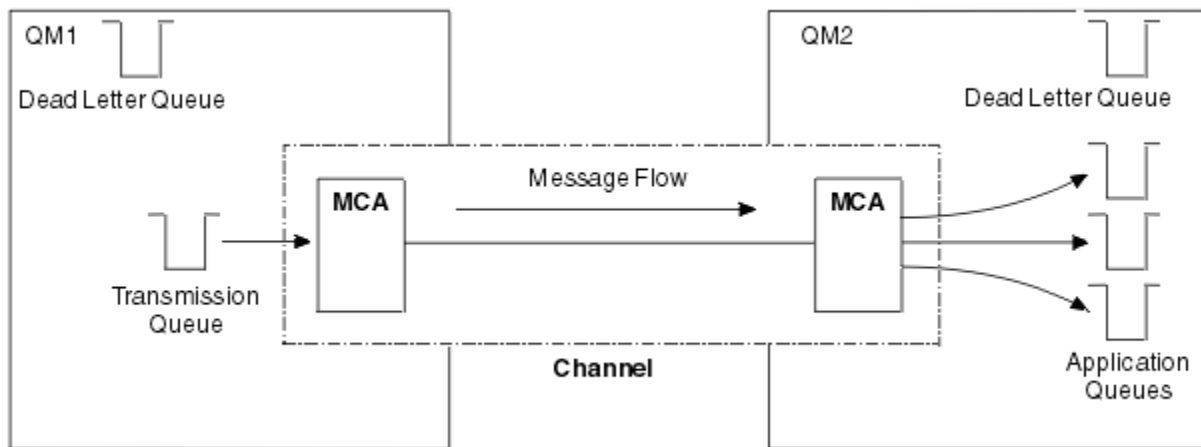
Má-li být zpráva odeslána vzdálenému správci front, lokální správce front potřebuje definice pro *přenosovou frontu* a *kanál*. Kanál je jednosměrné komunikační propojení mezi dvěma správci front. Může přenášet zprávy určené pro libovolný počet front ve vzdáleném správci front.

Každý konec kanálu má samostatnou definici, která jej definuje například jako odesílající nebo přijímající konec. Jednoduchý kanál se skládá z definice kanálu *sender* v lokálním správci front a z definice kanálu *receiver* ve vzdáleném správci front. Tyto dvě definice musí mít stejný název a společně tvoří jeden kanál.

Software, který obsluhuje odesílání a příjem zpráv, se nazývá *Message Channel Agent (MCA)*. Na každém konci kanálu je *agent kanálu zpráv (MCA)*.

Každý správce front by měl mít *frontu nedoručených zpráv* (označovanou také jako *fronta nedoručených zpráv*). Zprávy jsou vloženy do této fronty, pokud nemohou být doručeny do místa určení.

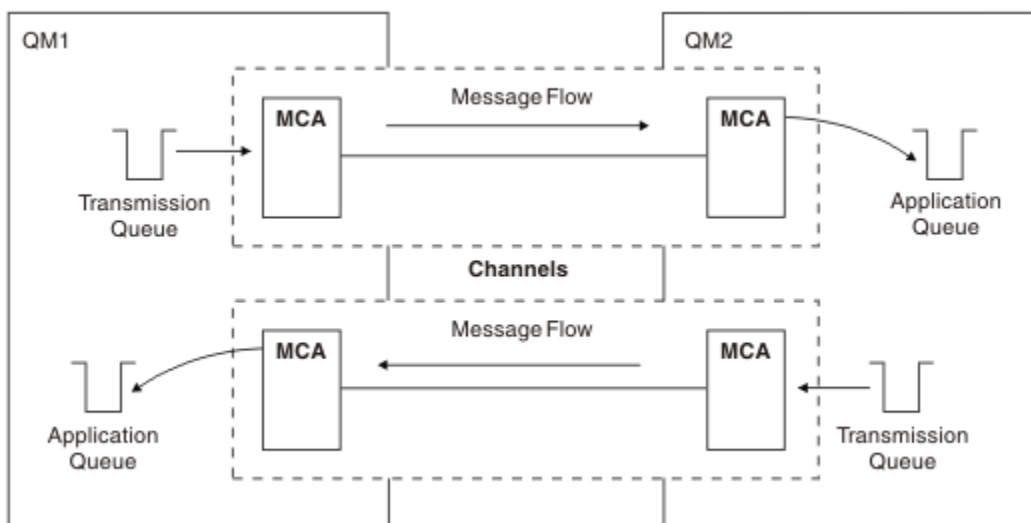
Následující obrázek ukazuje vztah mezi správci front, přenosovými frontami, kanály a MCA:



Obrázek 5. Odesílání zpráv

Komponenty potřebné k vrácení zprávy

Pokud vaše aplikace vyžaduje vrácení zpráv ze vzdáleného správce front, je třeba definovat jiný kanál, aby se mezi správci front spouštěli v opačném směru, jak ukazuje následující obrázek:



Obrázek 6. Odesílání zpráv v obou směrech

Klastry

Namísto ručního definování všech připojení v prostředí distribuovaných front můžete seskupit sadu správců front v klastru. V takovém případě mohou správci front zpřístupnit fronty, které jsou hostiteli jiných správců front v klastru, bez nutnosti explicitních definic kanálů, definic vzdálených front nebo přenosových front pro každé místo určení. Každý správce front v klastru má jednu přenosovou frontu, která přenáší zprávy do libovolného jiného správce front v klastru. Pro každého správce front je třeba definovat pouze jeden přijímací kanál klastru a jeden odesílací kanál klastru. Další kanály jsou automaticky spravovány klastrem.

Klient IBM MQ se může připojit ke správci front, který je součástí klastru, stejně jako se může připojit k jinému správci front. Stejně jako u ručně konfigurovaného distribuovaného řazení do front můžete pomocí volání MQPUT vložit zprávu do fronty v libovolném správci front. Volání MQGET se používá k načtení zpráv z lokální fronty.

Správci front na platformách, které podporují klastry, nemusí být součástí klastru. Můžete pokračovat v ruční konfiguraci distribuovaného řazení do front a také, nebo místo toho, pomocí klastrů.

Výhody použití klastrů

Klastrování poskytuje dvě klíčové výhody:

- Klastry zjednodušují administraci sítí IBM MQ, které obvykle vyžadují konfiguraci mnoha definic objektů pro kanály, přenosové fronty a vzdálené fronty. Tato situace platí zejména ve velkých, potenciálně se měnících sítích, kde je třeba propojit mnoho správců front. Tuto architekturu je obzvláště obtížné konfigurovat a aktivně udržovat.
- Klastry lze použít k rozdělení pracovní zátěže provozu zpráv mezi fronty a správce front v klastru. Taková distribuce umožňuje rozdělit pracovní zátěž zpráv jedné fronty mezi ekvivalentní instance této fronty umístěné ve více správcích front. Tuto distribuci pracovní zátěže lze použít k dosažení větší odolnosti vůči selháním systému a ke zlepšení výkonu škálování zvláště aktivních toků zpráv v systému. V takovém prostředí každá z instancí distribuovaných front využívá aplikace zpracovávající zprávy. Další informace naleznete v tématu [Použití klastrů pro správu pracovní zátěže](#).

Způsob směrování zpráv v klastru

Klastr lze považovat za síť správců front udržovanou svědomitým administrátorem systému. Kdykoli definujete frontu klastru, administrátor systému automaticky vytvoří odpovídající definice vzdálených front podle potřeby v ostatních správcích front.

Není třeba vytvářet definice přenosových front, protože produkt IBM MQ poskytuje přenosovou frontu pro každého správce front v klastru. Tuto jedinou přenosovou frontu lze použít k přenosu zpráv do

libovolného jiného správce front v klastru. Nejste omezeni na použití jedné přenosové fronty. Správce front může používat více přenosových front k oddělení zpráv předávaných jednotlivým správcům front v klastru. Správce front obvykle používá jednu přenosovou frontu klastru. Můžete změnit atribut správce front DEFCLXQtak, aby správce front používal pro každého správce front v klastru jinou přenosovou frontu klastru. Přenosové fronty klastru můžete také definovat ručně.

Všichni správci front, kteří se připojují ke klastru, souhlasí, že budou pracovat tímto způsobem. Odesílají informace o sobě a o frontách, které hostí, a přijímají informace o ostatních členech klastru.

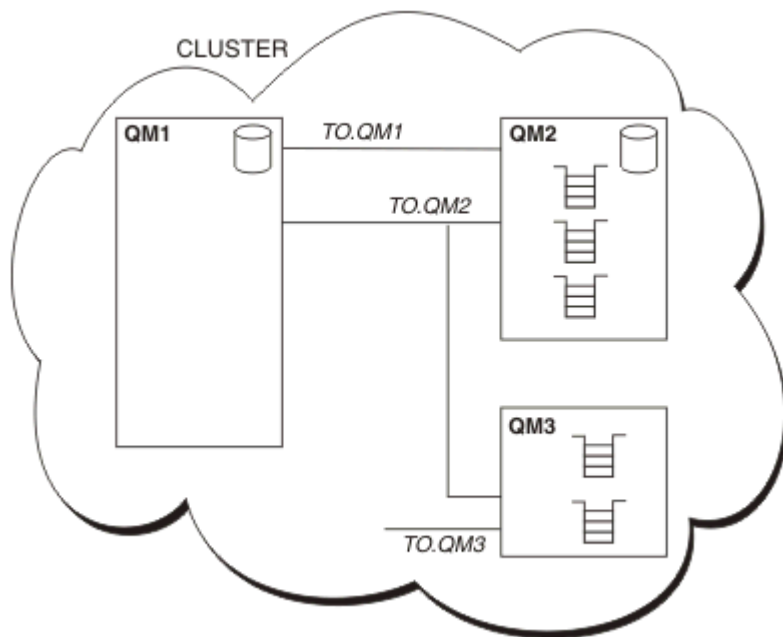
Chcete-li zajistit, aby při nedostupnosti správce front nebyly ztraceny žádné informace, zadejte v klastru dva správce front, kteří budou pracovat jako *úplná úložiště*. Tito správci front ukládají úplnou sadu informací o všech správcích front a frontách v klastru. Všichni ostatní správci front v klastru ukládají informace pouze o těch správcích front a frontách, se kterými si vyměňují zprávy. Tito správci front jsou označováni jako *dílčí úložiště*. Další informace viz [“Úložiště klastru”](#) na stránce 54.

Aby se správce front mohl stát součástí klastru, musí mít dva kanály: odesílací kanál klastru a přijímací kanál klastru:

- Odesílací kanál klastru je komunikační kanál jako odesílací kanál. Musíte ručně vytvořit jeden kanál odesilatele klastru ve správcí front, abyste jej mohli připojit k úplnému úložišti, které je již členem klastru.
- Přijímací kanál klastru je komunikační kanál jako přijímací kanál. Musíte ručně vytvořit jeden přijímací kanál klastru. Kanál se chová jako mechanismus pro správce front pro příjem komunikace klastru.

Všechny ostatní kanály potřebné pro komunikaci mezi tímto správcem front a ostatními členy klastru jsou poté vytvořeny automaticky.

Následující obrázek ukazuje komponenty klastru s názvem CLUSTER:



Obrázek 7. Klastř správců front

- KLASTR obsahuje tři správce front, QM1, QM2 a QM3.
- QM1 a QM2 jsou hostiteli úplných úložišť informací o správcích front a frontách v klastru.
- QM2 a QM3 jsou hostiteli některých front klastru, tj. front, které jsou přístupné pro všechny ostatní správce front v klastru.
- Každý správce front má přijímací kanál klastru s názvem TO.qmgr, na kterém může přijímat zprávy.
- Každý správce front má také odesílací kanál klastru, na kterém může odesílat informace jednomu ze správců front úložiště.

- QM1 a QM3 odesílají do úložiště na adrese QM2 a QM2 do úložiště na adrese QM1.

Komponenty distribuovaného řazení do front

Komponenty distribuovaných front jsou kanály zpráv, agenti kanálů zpráv, přenosové fronty, iniciátory kanálů a moduly listener a programy uživatelských procedur kanálu. Definice každého konce kanálu zpráv může být jednoho z několika typů.

Kanály zpráv jsou kanály, které přenášejí zprávy z jednoho správce front do jiného. Nepleťte si kanály zpráv s kanály MQI. Existují dva typy kanálu MQI: připojení serveru (SVRCONN) a připojení klienta (CLNTCONN). Další informace viz [Kanály](#).

Definice každého konce kanálu zpráv může být jednoho z následujících typů:

- Odesílatel (SDR)
- Přijímač (RCVR)
- Server (SVR)
- Žadatel (RQSTR)
- Odesílatel klastru (CLUSSDR)
- Příjemce klastru (CLUSRCVR)

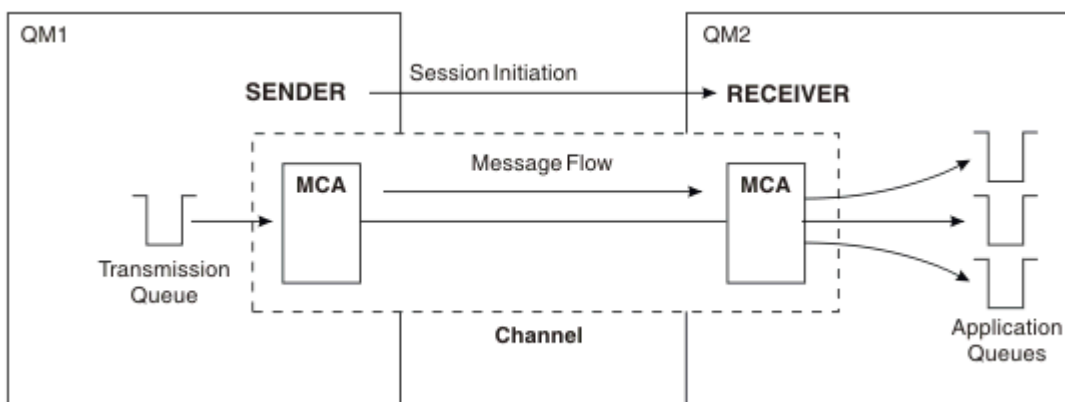
Kanál zpráv je definován pomocí jednoho z těchto typů definovaných na jednom konci a kompatibilního typu na druhém konci. Možné kombinace jsou:

- Odesílací-přijímací.
- Žadatelský-serveru.
- Žadatelský-odesílací (zpětné volání).
- Serveru-přijímací.
- Odesílatel klastru-příjemce klastru

Podrobné pokyny pro vytvoření kanálu odesílatele a příjemce jsou obsaženy v části [Definování kanálů](#). Příklady parametrů potřebných pro nastavení přijímacích kanálů odesílatele naleznete v tématu [Příklad informací o konfiguraci](#), které jsou použitelné pro vaši platformu. Parametry potřebné k definování kanálu libovolného typu viz [DEFINE CHANNEL](#).

Odesílací-přijímací kanály

Odesílatel v jednom systému spustí kanál, aby mohl odesílat zprávy do druhého systému. Odesílatel požaduje spuštění příjemce na druhém konci kanálu. Odesílatel odesílá zprávy z přenosové fronty příjemci. Příjemce vloží zprávy do cílové fronty. [Obrázek 8](#) na stránce 45 to ilustruje.

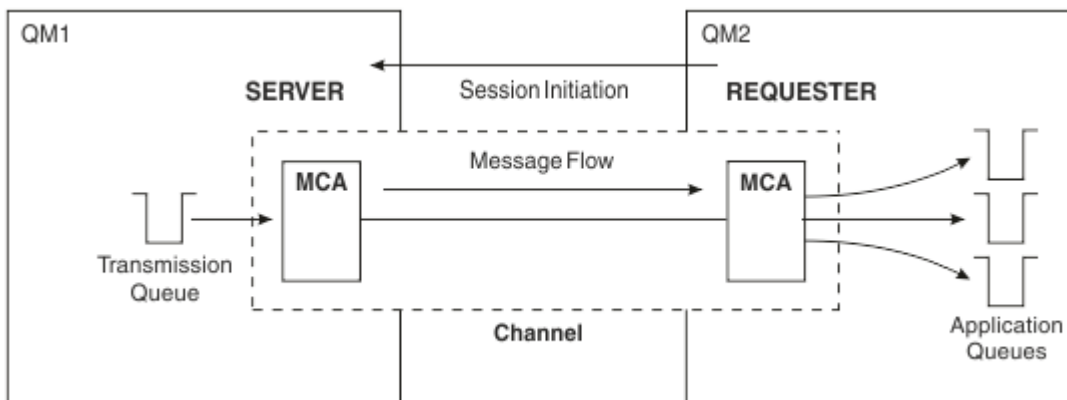


Obrázek 8. Kanál příjemce odesílatele

Kanály žadatele-server

Klient v jednom systému spustí kanál, aby mohl přijímat zprávy z druhého systému. Žadatel požaduje spuštění serveru na druhém konci kanálu. Server odesílá zprávy žadateli z přenosové fronty definované v jeho definici kanálu.

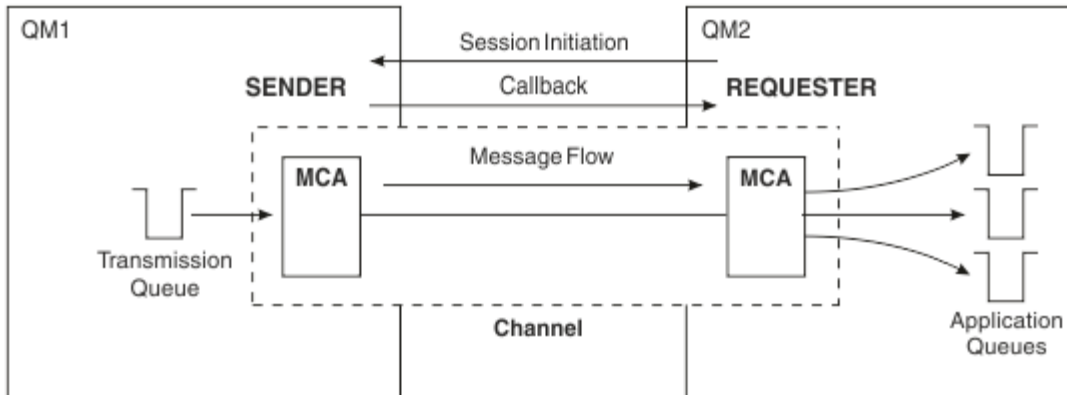
Kanál serveru může také zahájit komunikaci a odesílat zprávy žadateli. Toto platí pouze pro *úplné* servery, tj. kanály serveru, které mají název připojení partnera uvedeného v definici kanálu. Úplný server může být buď spuštěn žadatelem, nebo může zahájit komunikaci s žadatelem.



Obrázek 9. Kanál žadatele-server

Kanály žadatele-odesílající kanály

Žadatel spustí kanál a odesílatel ukončí volání. Odesílatel poté restartuje komunikaci podle informací v definici kanálu (známé jako *callback*). Odešle zprávy z přenosové fronty žadateli.



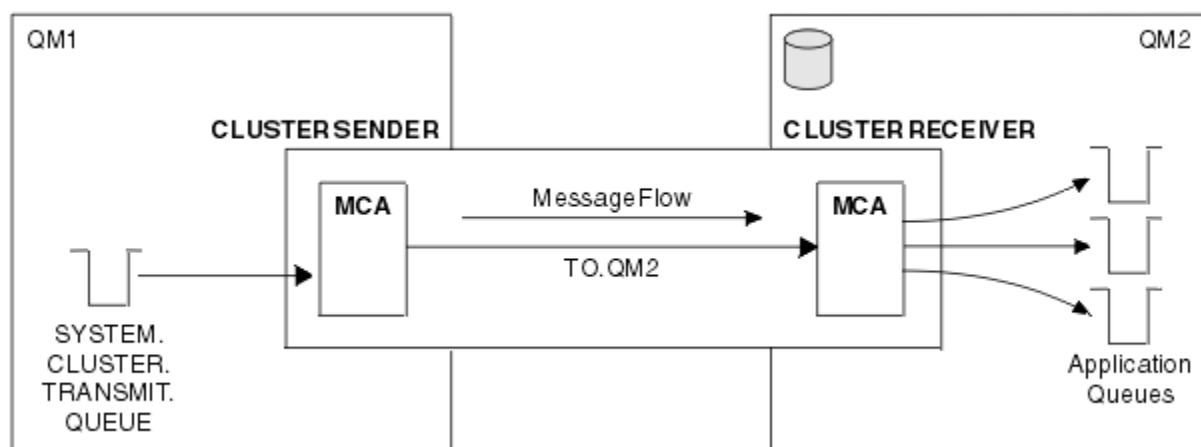
Obrázek 10. Kanál žadatele-odesílatel

Přijímací kanály serveru

Je to stejné jako příjemce odesílatele, ale týká se pouze *úplných* serverů, tj. kanálů serveru, které mají název připojení partnera uvedeného v definici kanálu. Spuštění kanálu musí být zahájeno na konci serveru propojení. Obrázek je podobný obrázku v části [Obrázek 8](#) na stránce 45.

Odesílací kanály klastru

V klastru má každý správce front odesílací kanál klastru, na kterém může odesílat informace o klastru jednomu ze správců front úplného úložiště. Správci front mohou také odesílat zprávy jiným správcům front v odesílacích kanálech klastru.



Obrázek 11. Odesílací kanál klastru

Přijímací kanály klastru

V klastru má každý správce front přijímací kanál klastru, na kterém může přijímat zprávy a informace o klastru. Obrázek je podobný obrázku v části [Obrázek 11](#) na stránce 47.

Fronty nedoručené-dopis

Fronta nedoručených zpráv (nebo fronta nedoručených zpráv) je fronta, do které jsou odesílány zprávy, pokud je nelze směřovat na správné místo určení. Každý správce front má obvykle frontu nedoručených zpráv.

Fronta nedoručených zpráv (DLQ), někdy označovaná jako *fronta nedoručených zpráv*, je fronta zadržení pro zprávy, které nelze doručit do cílových front, například proto, že fronta neexistuje nebo je plná. Fronty nedoručených zpráv se také používají na odesílajícím konci kanálu pro chyby převodu dat. Každý správce front v síti má obvykle lokální frontu, která má být použita jako fronta nedoručených zpráv, takže zprávy, které nemohou být doručeny do správného místa určení, mohou být uloženy pro pozdější načtení.

Zprávy mohou být do fronty DLQ vkládány správci front, agenti kanálů zpráv (MCA) a aplikacemi. Všechny zprávy ve frontě DLQ musí mít předponu se strukturou *záhlaví nedoručených zpráv* MQDLH. Pole *Příčina* struktury MQDLH obsahuje kód příčiny, který identifikuje, proč je zpráva v DLQ.

Obvykle byste měli definovat frontu nedoručených zpráv pro každého správce front. Pokud tak neučiníte a agent MCA nemůže vložit zprávu, bude ponechán v přenosové frontě a kanál bude zastaven. Také, pokud se jedná o rychlé, dočasné zprávy (viz [Rychlé, přechodné zprávy](#)), nelze doručit a v cílovém systému neexistuje žádná fronta nedoručených zpráv, tyto zprávy jsou vyřazeny.

Použití front nedoručených zpráv však může ovlivnit pořadí, ve kterém jsou zprávy doručovány, a proto se můžete rozhodnout je nepoužívat.

Související úlohy

[Práce s frontami nedoručených zpráv](#)

[Odstraňování problémů s nedoručených zpráv](#)

Související odkazy

[runmqdlq](#) (spustit obslužnou rutinu fronty nedoručených zpráv)

Definice vzdálené fronty

Definice vzdálených front jsou definice pro fronty vlastněné jiným správcem front.

Zatímco aplikace mohou načítat zprávy pouze z lokálních front, mohou vkládat zprávy do lokálních front nebo vzdálených front. Proto, stejně jako definice pro každou z lokálních front, může mít správce front *definice vzdálených front*. Výhodou definic vzdálených front je, že umožňují aplikaci vkládat zprávy

do vzdálené fronty bez nutnosti zadávat název vzdálené fronty, vzdáleného správce front nebo název přenosové fronty. Definice vzdálených front vám poskytují nezávislost umístění.

Pro definice vzdálených front existují další použití, která jsou popsána později.

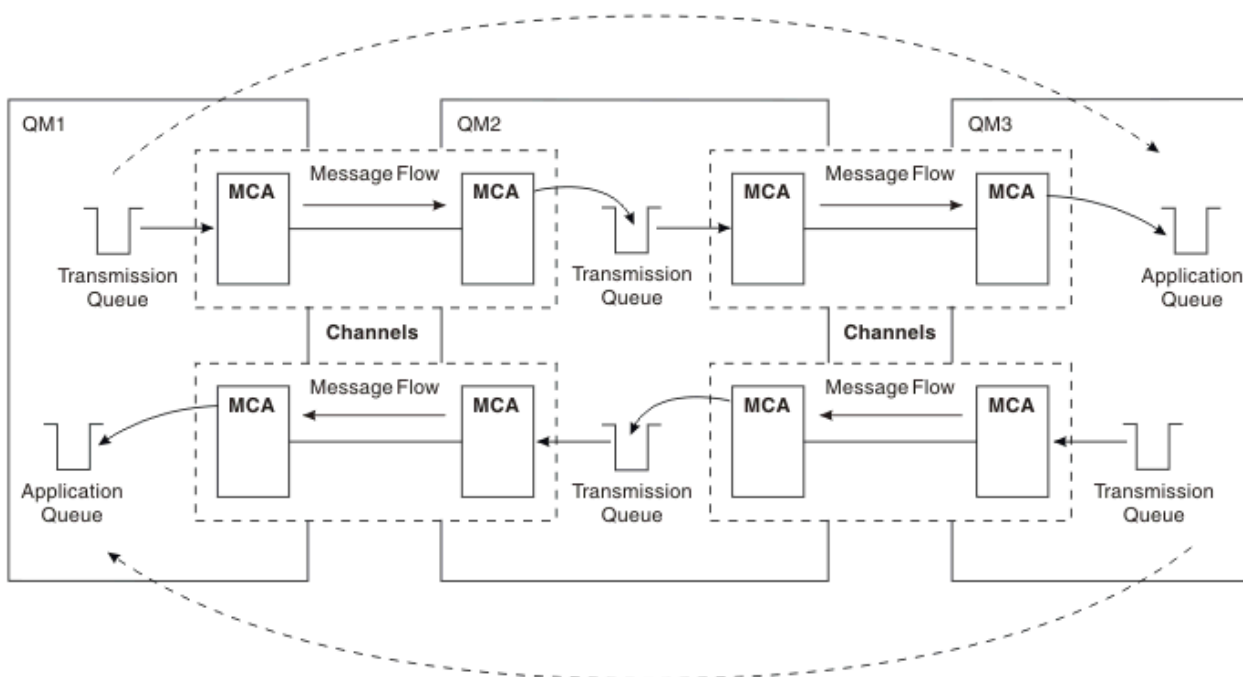
Jak se dostat ke vzdálenému správci front

Mezi každým zdrojovým a cílovým správcem front nemusí být vždy jeden kanál. Existuje celá řada dalších způsobů propojení mezi těmito dvěma kanály, včetně multi-hopping, sdílení kanálů, pomocí různých kanálů a klastrování.

Multi-hopping

Pokud mezi správcem zdrojové fronty a správcem cílové fronty neexistuje přímé komunikační spojení, je možné při cestě ke správci cílové fronty přejít prostřednictvím jednoho či více *intermediačních správců front*. Toto je známé jako *multi-hop*.

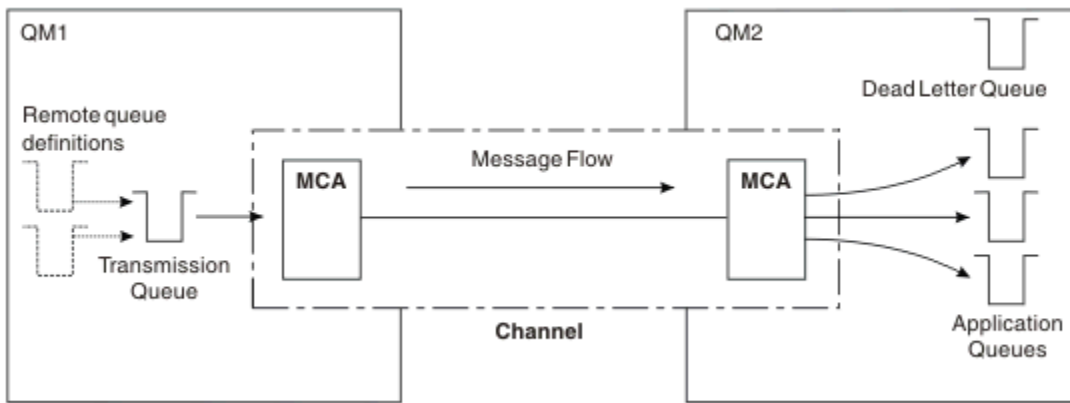
Je třeba definovat kanály mezi všemi správcem front a přenosovými frontami v intermediačních správcích front. To je zobrazeno v části [Obrázek 12 na stránce 48](#).



Obrázek 12. Průchod prostředními správci front

Sdílení kanálů

Jako návrhář aplikací můžete vynutit, aby aplikace zadaly název vzdáleného správce front spolu s názvem fronty, nebo vytvořit *definici vzdálené fronty* pro každou vzdálenou frontu. Tato definice obsahuje název vzdáleného správce front, název fronty a název přenosové fronty. V obou případech všechny zprávy ze všech aplikací adresujících fronty ve stejném vzdáleném umístění odesílají své zprávy prostřednictvím stejné přenosové fronty. To je zobrazeno v části [Obrázek 13 na stránce 49](#).



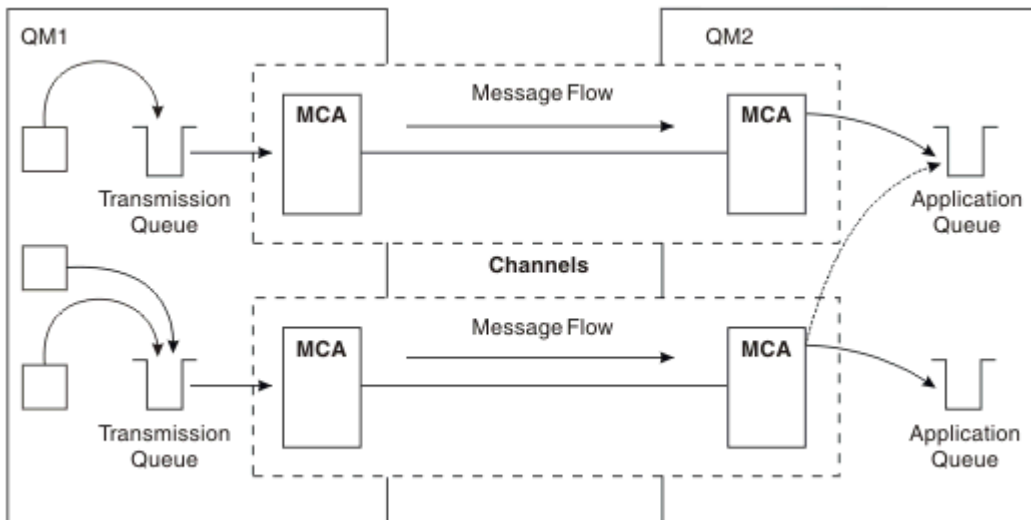
Obrázek 13. Sdílení přenosové fronty

Obrázek 13 na stránce 49 ilustruje, že zprávy z více aplikací do více vzdálených front mohou používat stejný kanál.

Použití různých kanálů

Máte-li mezi dvěma správci front zprávy různých typů, můžete mezi nimi definovat více než jeden kanál. Jsou chvíle, kdy potřebujete alternativní kanály, možná pro bezpečnostní účely, nebo vyměnit rychlost doručení proti naprosté části přenosu zpráv.

Chcete-li nastavit druhý kanál, musíte definovat jiný kanál a jinou přenosovou frontu a vytvořit definici vzdálené fronty určující umístění a název přenosové fronty. Aplikace pak mohou používat oba kanály, ale zprávy jsou stále doručovány do stejných cílových front. To je zobrazeno v části [Obrázek 14 na stránce 49](#).



Obrázek 14. Použití více kanálů

Používáte-li k určení přenosové fronty definice vzdálené fronty, vaše aplikace **nesmí** určovat umístění (tj. správce cílové fronty) samy. Pokud ano, správce front nepoužívá definice vzdálených front. Definice vzdálených front vám poskytují nezávislost umístění. Aplikace mohou vkládat zprávy do *logické* fronty, aniž by věděli, kde se fronta nachází, a můžete změnit *fyzickou* frontu, aniž byste museli měnit aplikace.

Použití klastrování

Každý správce front v rámci klastru definuje přijímací kanál klastru. Pokud chce jiný správce front odeslat zprávu tomuto správci front, definuje příslušný odesílací kanál klastru automaticky. Pokud například v klastru existuje více než jedna instance fronty, může být kanál odesílatele klastru definován pro všechny

správce front, kteří jsou hostiteli fronty. Produkt IBM MQ používá algoritmus správy pracovní zátěže, který používá rutinu round-robin k výběru dostupného správce front pro směrování zprávy. Další informace viz [Klastry](#).

Informace o adresování

Když aplikace vkládá zprávy, které jsou určeny pro vzdáleného správce front, lokální správce front k nim přidá záhlaví přenosu před jejich umístěním do přenosové fronty. Toto záhlaví obsahuje název cílové fronty a správce front, tj. *informace o adresování*.

V prostředí s jedním správcem front je adresa cílové fronty vytvořena, když aplikace otevře frontu pro vkládání zpráv. Vzhledem k tomu, že cílová fronta je ve stejném správcem front, není třeba žádné informace o adresování.

V distribuovaném prostředí front musí správce front znát nejen název cílové fronty, ale také umístění této fronty (tj. název správce front) a trasu do tohoto vzdáleného umístění (tj. přenosové fronty). Tyto informace o adresování jsou obsaženy v záhlaví přenosu. Přijímající kanál odebere záhlaví přenosu a použije v něm informace k vyhledání cílové fronty.

Pokud používáte definici vzdálené fronty, můžete se vyhnout tomu, aby aplikace zadaly název cílového správce front. Tato definice určuje název vzdálené fronty, název vzdáleného správce front, do kterého jsou zprávy určeny, a název přenosové fronty použité k přenosu zpráv.

Co jsou to aliasy?

Alias se používají k zajištění kvality služby pro zprávy. Alias správce front umožňuje administrátorovi systému změnit název cílového správce front, aniž byste museli měnit aplikace. Umožňuje také administrátorovi systému změnit trasu na cílového správce front nebo nastavit trasu, která zahrnuje průchod několika dalšími správci front (vícenásobné přeskokování). Alias fronty pro odpověď poskytuje kvalitu služby pro odpovědi.

Alias správce front a aliasy fronty pro odpovědi jsou vytvořeny pomocí definice vzdálené fronty s prázdným názvem RNAME. Tyto definice nedefinují skutečné fronty. Správce front je používá k rozlišování fyzických názvů front, názvů správců front a přenosových front.

Definice aliasů jsou charakterizovány tím, že mají prázdný název RNAME.

Rozlišení názvu fronty

Rozlišení názvů front se vyskytuje v každém správcem front při každém otevření fronty. Jeho účelem je identifikovat cílovou frontu, cílového správce front (který může být lokální) a trasu k tomuto správcem front (která může mít hodnotu null). Rozlišený název má tři části: název správce front, název fronty a, pokud je správce front vzdálený, přenosovou frontu.

Když existuje definice vzdálené fronty, nejsou odkazovány žádné definice aliasů. Název fronty zadaný aplikací je převeden na název cílové fronty, vzdáleného správce front a přenosové fronty určené v definici vzdálené fronty. Podrobnější informace o rozlišení názvů front naleznete v tématu [Rozlišení názvů front](#).

Pokud neexistuje žádná definice vzdálené fronty a název správce front je určen nebo interpretován službou názvů, správce front se pokusí zjistit, zda existuje definice aliasu správce front, která odpovídá zadanému názvu správce front. Pokud je k dispozici, informace v ní obsažené se použijí k vyřešení názvu správce front na název cílového správce front. Definici aliasu správce front lze také použít k určení přenosové fronty na cílového správce front.

Pokud převedený název fronty není lokální frontou, jsou v záhlaví přenosu jednotlivých zpráv vložených aplikací do přenosové fronty zahrnuty jak název správce front, tak název fronty.

Použitá přenosová fronta má obvykle stejný název jako vyřešený správce front, pokud ji nezmění definice vzdálené fronty nebo definice aliasu správce front. Pokud jste takovou přenosovou frontu nedefinovali, ale definovali jste výchozí přenosovou frontu, bude použita.

 Názvy správců front spuštěných v systému z/OS jsou omezeny na čtyři znaky.

Definice aliasů správce front

Definice aliasů správce front se použijí v případě, že aplikace, která otevře frontu pro vložení zprávy, určuje název fronty a název správce front.

Definice aliasů správce front mají tři použití:

- Při odesílání zpráv přemapování názvu správce front
- Při odesílání zpráv, změně nebo uvedení přenosové fronty
- Při příjmu zpráv se zjišťuje, zda je lokální správce front zamýšleným cílem pro tyto zprávy.

Odchozí zprávy-přemapování názvu správce front

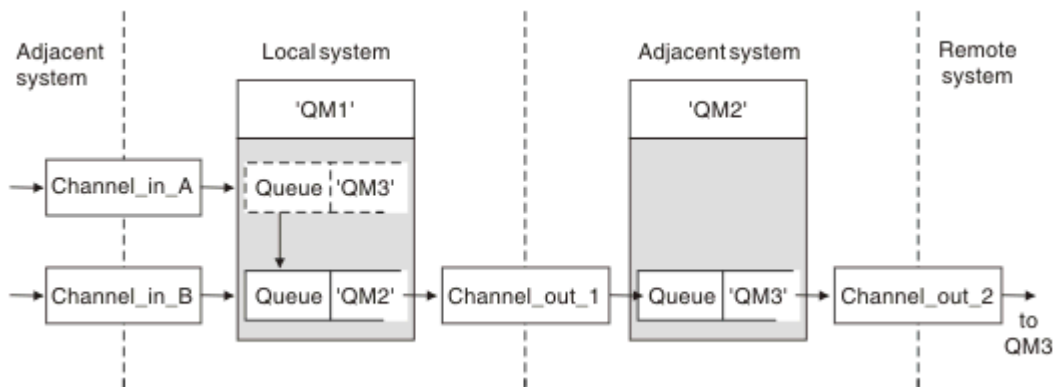
Definice aliasů správce front lze použít k opětovnému mapování názvu správce front určeného ve volání MQOPEN. Například volání MQOPEN určuje název fronty THISQ a název správce front YOURQM. V lokálním správci front existuje definice aliasu správce front, jako v následujícím příkladu:

```
DEFINE QREMOTE (YOURQM) RQMNAME (REALQM)
```

To ukazuje, že skutečný správce front, který má být použit, když aplikace vkládá zprávy do správce front YOURQM, je REALQM. Pokud je lokálním správcem front REALQM, vloží zprávy do fronty THISQ, což je lokální fronta. Pokud lokální správce front není nazván REALQM, směruje zprávu do přenosové fronty s názvem REALQM. Správce front změní záhlaví přenosu na hodnotu REALQM namísto hodnoty YOURQM.

Odchozí zprávy-změna nebo určení přenosové fronty

Obrázek 15 na stránce 51 ukazuje scénář, ve kterém zprávy přicházejí do správce front QM1 s přenosovými záhlavími zobrazujícími názvy front ve správci front QM3. V tomto scénáři je produkt QM3 dosažitelný pomocí vícenásobného přechodu přes QM2.



Obrázek 15. Alias správce front

Všechny zprávy pro systém QM3 jsou zachyceny v adresáři QM1 s aliasem správce front. Alias správce front má název QM3 a obsahuje definici QM3 prostřednictvím přenosové fronty QM2. Definice vypadá jako následující příklad:

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

Správce front vloží zprávy do přenosové fronty QM2, ale nezmění záhlaví přenosové fronty, protože název cílového správce front QM3 se nezmění.

Všechny zprávy přicházející do QM1 a zobrazující záhlaví přenosu obsahující název fronty v QM2 jsou také vloženy do přenosové fronty QM2. Tímto způsobem se zprávy s různými cíli shromažďují do společné přenosové fronty do příslušného sousedního systému pro další přenos do svých cílů.

Příchozí zprávy-určení místa určení

Přijímající agent MCA otevře frontu odkazovanou v záhlaví přenosu. Pokud existuje definice aliasu správce front se stejným názvem jako odkazovaný správce front, bude název správce front přijatý v záhlaví přenosu nahrazen názvem RQMNAME z této definice.

Tento proces má dvě použití:

- Směrování zpráv do jiného správce front
- Změna názvu správce front tak, aby byl stejný jako lokální správce front

Definice aliasu fronty pro odpověď

Definice aliasu fronty pro odpověď určuje alternativní názvy pro informace o odpovědi v deskriptoru zprávy. Výhodou je, že můžete změnit název fronty nebo správce front, aniž byste museli měnit aplikace.

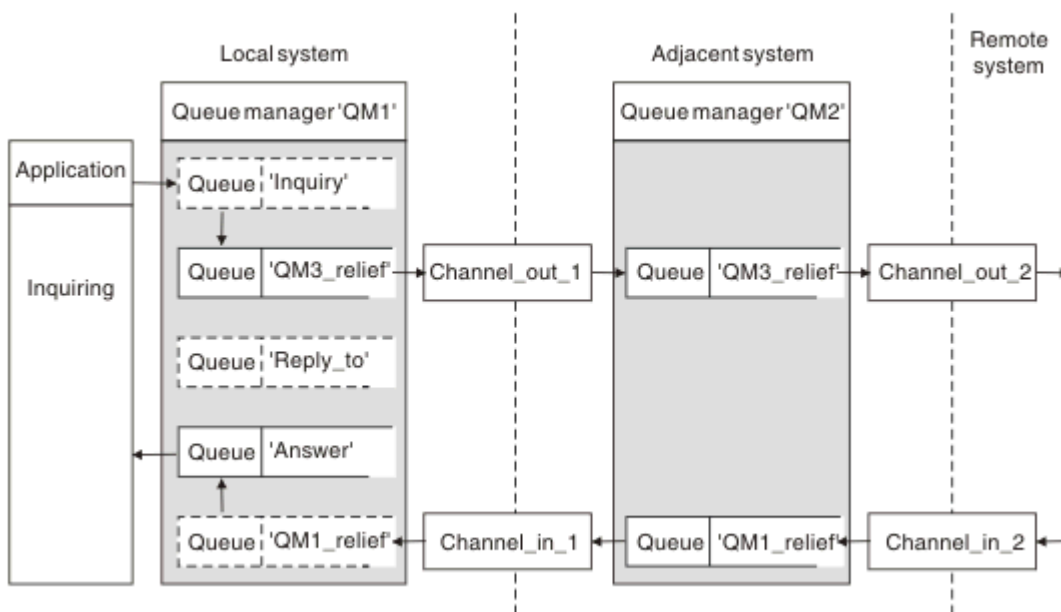
Rozlišení názvu fronty

Když aplikace odpoví na zprávu, použije data v *deskriptoru zprávy* zprávy, kterou přijala, aby zjistila název fronty, na kterou má odpovědět. Odesílající aplikace označuje, kam jsou odesílány odpovědi, a připojuje tyto informace ke svým zprávám. Tento koncept musí být koordinován jako součást návrhu aplikace.

Rozlišení názvu fronty se provádí na odesílajícím konci aplikace před vložením zprávy do fronty. Rozlišení názvu fronty se proto vyskytuje před interakcí se vzdálenou aplikací, do které je zpráva odesílána. Jedná se o jedinou situaci, kdy se rozlišování názvů provádí v době, kdy se fronta neotevřívá.

Rozlišení názvu fronty pomocí aliasu správce front

Aplikace obvykle určuje frontu pro odpověď a ponechává název správce front pro odpověď prázdný. Správce front dokončí svůj vlastní název v době vložení. Tato metoda funguje dobře s výjimkou případů, kdy chcete pro odpovědi použít alternativní kanál, například kanál, který používá přenosovou frontu QM1_relief místo výchozího návratového kanálu, který používá přenosovou frontu QM1. V této situaci se názvy správců front určené v záhlavích přenosové fronty neshodují s "skutečnými" názvy správců front, ale jsou znovu určeny pomocí definic aliasů správců front. Chcete-li vrátit odpovědi na alternativní trasy, je nutné mapovat také data fronty pro odpovědi pomocí definic aliasů fronty pro odpovědi.



Obrázek 16. Alias fronty pro odpověď použitý pro změnu umístění odpovědi

V příkladu v souboru [Obrázek 16](#) na stránce 52:

1. Aplikace vloží zprávu pomocí volání MQPUT a v deskriptoru zprávy uvede následující informace:

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

ReplyToSprávce front musí být prázdný, aby bylo možné použít alias fronty pro odpověď.

2. Vytvoříte definici aliasu fronty pro odpověď s názvem Reply_to, která obsahuje název Answer, a název správce front QM1_relief.

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

3. Zprávy jsou odesílány s deskriptorem zprávy zobrazujícím ReplyToQ= 'Answer' a ReplyToQMgr= 'QM1_relief'.
4. Specifikace aplikace musí obsahovat informace, které mají být nalezeny ve frontě Answer a nikoli Reply_to.

Chcete-li se připravit na odpovědi, musíte vytvořit paralelní návratový kanál a definovat:

- V QM2se jedná o přenosovou frontu s názvem QM1_relief.

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

- V QM1se jedná o alias správce front QM1_relief.

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

Tento alias správce front ukončí řetěz paralelních návratových kanálů a zachytí zprávy pro QM1.

Pokud si myslíte, že byste to mohli chtít provést někdy v budoucnu, ujistěte se, že aplikace používají název aliasu od začátku. Nyní se jedná o normální alias fronty pro frontu pro odpověď, ale později jej lze změnit na alias správce front.

Název fronty pro odpověď

Při pojmenovávání front pro odpovědi je třeba věnovat pozornost. Důvodem, proč aplikace vkládá do zprávy název fronty pro odpověď, je, že může určit frontu, do které jsou odesílány její odpovědi. Když vytvoříte definici aliasu fronty pro odpověď s tímto názvem, nemůžete mít skutečnou frontu pro odpověď (tj. definici lokální fronty) se stejným názvem. Proto musí definice aliasu fronty pro odpověď obsahovat nový název fronty i název správce front a specifikace aplikace musí obsahovat informace o tom, že její odpovědi byly nalezeny v této jiné frontě.

Aplikace nyní musí při vložení původní zprávy načíst zprávy z jiné fronty z fronty, kterou pojmenovaly jako frontu pro odpověď.

Komponenty klastru

Klastry se skládají ze správců front, úložišť klastrů, kanálů klastrů a front klastrů.

Informace o jednotlivých komponentách klastru naleznete v následujících dílčích tématech:

Související pojmy

[Porovnání klastrování a distribuovaného řazení do front](#)

Související úlohy

[Konfigurace klastru správců front](#)

[Nastavení nového klastru](#)

Úložiště klastru

Úložiště je kolekce informací o správcích front, kteří jsou členy klastru.

Informace o úložišti zahrnují názvy správců front, jejich umístění, kanály, fronty, které jsou hostiteli, a další informace. Informace jsou uloženy ve formě zpráv ve frontě s názvem

SYSTEM.CLUSTER.REPOSITORY.QUEUE. Tato fronta je jedním z výchozích objektů.

Multi

V systému Multiplatforms je definován při vytváření IBM MQ správce front.

z/OS

V systému IBM MQ for z/OS je definován jako součást přizpůsobení správce front.

Úplné úložiště a dílčí úložiště

Zpravidla dva správci front v klastru obsahují úplné úložiště. Všichni zbývající správci front zadržují dílčí úložiště.

Správce front, který je hostitelem úplné sady informací o každém správcí front v klastru, má úplné úložiště. Ostatní správci front v klastru mají dílčí úložiště obsahující podmnožinu informací v úplných úložištích.

Dílčí úložiště obsahuje informace pouze o správcích front, se kterými si musí správce front vyměňovat zprávy. Správci front požadují aktualizace potřebných informací, takže pokud se změní, správce front úplného úložiště jim odešle nové informace. Po většinu času obsahuje dílčí úložiště všechny informace, které musí správce front provést v rámci klastru. Pokud některý správce front potřebuje další informace, vyžádá si je z úplného úložiště a poté provede aktualizaci svého dílčího úložiště. Správci front používají frontu SYSTEM.CLUSTER.COMMAND.QUEUE k vyžádání a přijímání aktualizací do úložišť.

Při migraci správců front, kteří jsou členy klastru, proveďte migraci úplných úložišť před dílčími úložišti. Důvodem je, že starší úložiště nemůže ukládat novější atributy zavedené v novější verzi. Toleruje je, ale neskládá je.

Správce front klastru

Správce front klastru je správce front, který je členem klastru.

Jeden správce front může být členem více klastrů. Každý správce front klastru musí mít název, který je jedinečný ve všech klastrech, jejichž je členem.

Správce front klastru může hostovat fronty, které inzeruje ostatním správcům front v klastru. Nemusí to však dělat. Místo toho může odesílat zprávy do front hostovaných jinde v klastru a přijímat pouze odpovědi, které jsou k němu výslovně směřovány.

z/OS

V produktu IBM MQ for z/OS může být správce front klastru členem skupiny sdílení front. V tomto případě sdílí své definice front s ostatními správcí front ve stejné skupině sdílení front.

Správci front klastru jsou autonomní. Mají úplnou kontrolu nad frontami a kanály, které definují. Jejich definice nemohou být upraveny jinými správcí front (jinými než správci front ve stejné skupině sdílení front). Správci front úložiště neřídí definice v jiných správcích front v klastru. Obsahují úplnou sadu všech definic pro použití v případě potřeby. Klaster je federací správců front.

Po vytvoření nebo změně definice ve správcí front klastru jsou informace odeslány správcí front úplného úložiště. Další úložiště v klastru jsou aktualizována později.

Správce front úplného úložiště

Správce front úplného úložiště je správce front klastru, který obsahuje úplnou reprezentaci prostředků klastru. Chcete-li zajistit dostupnost, nastavte v každém klastru dva nebo více správců front úplného úložiště. Správci front úplného úložiště obdrží informace odeslané ostatními správcí front v klastru a aktualizují svá úložiště. Vzájemně si odesílají zprávy, aby si byli jisti, že jsou oba průběžně aktualizovány s novými informacemi o klastru.

Správci front a úložiště

Každý klastr má alespoň jednoho (nejlépe dva) správce front, který obsahuje úplná úložiště informací o správcích front, frontách a kanálech v klastru. Tato úložiště také obsahují požadavky od ostatních správců front v klastru na aktualizace informací.

Ostatní správci front obsahují dílčí úložiště obsahující informace o podmnožině front a správcích front, s nimiž musí komunikovat. Správci front sestavují svá dílčí úložiště tím, že se dotazují, když nejprve potřebují přistupovat k jiné frontě nebo správci front. Požádají o oznámení všech nových informací týkajících se této fronty nebo správce front.

Každý správce front ukládá své informace o úložišti do zpráv ve frontě s názvem `SYSTEM.CLUSTER.REPOSITORY.QUEUE`. Správci front si vyměňují informace o úložišti ve zprávách ve frontě s názvem `SYSTEM.CLUSTER.COMMAND.QUEUE`.

Každý správce front, který se připojí ke klastru, definuje odesílací kanál klastru `CLUSDR` pro jedno z úložišť. Okamžitě se dozví, kteří další správci front v klastru obsahují úplná úložiště. Od té doby může správce front požadovat informace z libovolného úložiště. Když správce front odešle informace do vybraného úložiště, odešle také informace do jiného úložiště (pokud existuje).


Úplné úložiště je aktualizováno, když správce front, který je jeho hostitelem, obdrží nové informace od jednoho ze správců front, kteří jsou k němu propojeni. Nové informace jsou také odeslány do jiného úložiště, aby se snížilo riziko jejich zpoždění v případě, že je správce front úložiště mimo provoz. Vzhledem k tomu, že všechny informace jsou odesílány dvakrát, musí úložiště vyřadit duplikáty. Každá položka informací obsahuje pořadové číslo, které úložiště používají k identifikaci duplikátů. Všechna úložiště jsou udržována v kroku mezi sebou výměnou zpráv.

Fronty klastru

Fronta klastru je fronta, jejímž hostitelem je správce front klastru, a která je dostupná ostatním správcům front v klastru.

Definice fronty klastru se oznamuje ostatním správcům front v klastru. Ostatní správci front v klastru mohou vkládat zprávy do fronty klastru, aniž by potřebovali odpovídající definici vzdálené fronty. Fronta klastru může být oznámena ve více než jednom klastru pomocí seznamu názvů klastrů.

Po oznámení fronty může každý správce front v klastru do ní vkládat zprávy. Chcete-li správce front vložit zprávu, musí z úplných úložišť zjistit, kdo je hostitelem této fronty. Pak přidá do zprávy informace o směrování a vloží zprávu do přenosové fronty klastru.

 Fronta klastru může být fronta, kterou sdílí členové skupiny sdílení front v produktu IBM MQ for z/OS.

Související úlohy

[Definování front klastru](#)

Porovnání mezi sdílenými frontami a frontami klastru

Tyto informace jsou navrženy tak, aby vám pomohly porovnat sdílené fronty a fronty klastru a rozhodnout se, které by mohly být pro váš systém vhodnější.

Náklady na inicializátor kanálu

Ve frontách klastru jsou zprávy odesílány kanály, takže kromě nákladů na aplikace umožňují také náklady na iniciátory kanálů. V síti jsou náklady, protože kanály dostávají a ukládají zprávy. Tyto náklady nejsou k dispozici pro sdílené fronty, které proto při přesouvání zpráv mezi správci front ve skupině sdílení front využívají méně výkonu zpracování než fronty klastru.

Dostupnost zpráv

Při vkládání do fronty odesílají fronty klastru zprávu jednomu ze správců front s aktivními kanály připojenými ke správci front. Pokud ve vzdáleném správci front nefungují aplikace používané ke

zpracování zpráv, nebudou zprávy zpracovány a budou čekat na spuštění aplikací. Podobně platí, že pokud je správce front vypnutý, nejsou žádné zprávy ve správci front k dispozici, dokud se správce front znovu nespustí. Tyto instance zobrazují nižší dostupnost zpráv než při použití sdílených front.

Při použití sdílených front může každá aplikace ve skupině sdílení front získat odeslané zprávy. Pokud ukončíte jednoho správce front ve skupině sdílení front, budou pro ostatní správce front k dispozici zprávy, které zajistí vyšší dostupnost zpráv než při použití front klastru.

Kapacita

Prostředek Coupling Facility je dražší než disk; proto jsou náklady na uložení 1 000 000 zpráv v lokální frontě nižší než náklady na prostředek Coupling Facility s dostatečnou kapacitou pro uložení stejného počtu zpráv.

Odesílání jiným správcům front

Zprávy sdílené fronty jsou k dispozici pouze v rámci skupiny sdílení front. Chcete-li použít správce front mimo skupinu sdílení front, musíte použít kanály. Klastrování lze použít k vyvážení pracovní zátěže mezi více vzdálenými distribuovanými správci front.

Vyrovňování zátěže

Pomocí klastrování můžete určit váhu, pro kterou kanály a správci front získají část odeslaných zpráv. Můžete například odeslat 60% zpráv jednomu správci front a 40% zpráv jinému správci front. Tato instance nezávisí na schopnosti vzdáleného správce front zpracovat práci. Systém s prvním správcem front může být přetížený a systém s druhým správcem front může být nečinný, ale většina zpráv stále přejde do prvního správce front.

Se sdílenými frontami mohou dva systémy CICS získávat zprávy. Pokud je jeden systém přetížený, druhý systém převezme většinu pracovní zátěže.

Kanály klastru

V každém úplném úložišti ručně definujete přijímací kanál klastru a sadu odesílacích kanálů klastru pro připojení ke všem ostatním úplným úložištům v klastru. Když přidáte dílčí úložiště, ručně definujete přijímací kanál klastru a jeden odesílací kanál klastru, který se připojí k jednomu z úplných úložišť. Další odesílací kanály klastru jsou v případě potřeby automaticky definovány klastrem. Automaticky definované odesílací kanály klastru přebírá své atributy z odpovídající definice přijímacího kanálu klastru v přijímajícím správci front.

Přijímací kanál klastru: CLUSRCVR

Definice kanálu CLUSRCVR definuje konec kanálu, na kterém může správce front klastru přijímat zprávy od jiných správců front v klastru.

Pro každého správce front klastru musíte definovat alespoň jeden kanál CLUSRCVR. Definováním kanálu CLUSRCVR správce front zobrazí ostatní správce front klastru, které jsou k dispozici pro příjem zpráv.

Definice kanálu CLUSRCVR také umožňuje ostatním správcům front automaticky definovat odpovídající definice odesílacího kanálu klastru. Viz část [“Automaticky definované odesílací kanály klastru”](#) na stránce 57 tohoto článku.

Odesílací kanál klastru: CLUSSDR

Ručně definujete kanál CLUSSDR z každého správce front úplného úložiště do každého jiného správce front úplného úložiště v klastru. Všechny aktualizace vyměňované úplnými úložišti proudí výhradně na těchto kanálech. Ručním definováním těchto kanálů můžete explicitně řídit síť úplných úložišť.

Při přidávání správce front dílčího úložiště do klastru ručně definujete jeden kanál CLUSSDR pro připojení k jednomu z úplných úložišť. Nezáleží na tom, které úplné úložiště vyberete, protože po provedení počátečního kontaktu jsou další objekty správce front klastru pro vašeho správce front, včetně kanálů

CLUSSDR , definovány automaticky podle potřeby. To umožňuje správci front odesílat informace o klastru do libovolného úplného úložiště a odesílat zprávy všem správcům front v klastru.

Jak je vysvětleno v části tohoto článku, automaticky definované odesílací kanály jsou založeny na konfiguraci přijímacího kanálu klastru. Proto by měly být všechny vlastnosti kanálu, které jste nastavili na kanálech klastru, nastaveny stejně na odpovídajících kanálech CLUSSDR a přijímacích kanálech klastru nebo pouze na přijímacích kanálech klastru.

Kanály CLUSSDR byste měli definovat ručně pouze z výše uvedených důvodů. To znamená, že nejprve připojte dílčí úložiště k úplnému úložišti, nebo připojte dvě úplná úložiště dohromady. Ruční konfigurace kanálu CLUSSDR , který se připojuje k částečnému úložišti nebo ke správci front, který není v klastru, způsobí vydání chybových zpráv, například AMQ9427 a AMQ9428 . Ačkoli to může být někdy nevyhnutelné jako dočasná situace, například při úpravě umístění úplného úložiště, ruční definice by měla být odstraněna co nejdříve.

Automaticky definované odesílací kanály klastru

Při přidávání správce front dílčího úložiště do klastru obvykle ručně definujete pouze dva kanály klastru ve správci front:

- Odesílatel klastru (CLUSSDR) kanál ke správci front úplného úložiště pro klastr.
- Přijemce klastru (CLUSRCVR) kanál.

Definovaný kanál CLUSSDR umožňuje správci front navázat počáteční kontakt s klastrem. Po provedení počátečního kontaktu jsou další kanály CLUSSDR v případě potřeby automaticky definovány klastrem.

Automaticky definovaný kanál CLUSSDR přebírá své atributy z odpovídající definice kanálu CLUSRCVR v přijímacím správci front. I když existuje ručně definovaný kanál CLUSSDR , použijí se atributy z automaticky definovaného kanálu CLUSSDR . Předpokládáme například, že definujete kanál CLUSRCVR bez zadání čísla portu v parametru **CONNNAME** a ručně definujete kanál CLUSSDR , který určuje číslo portu. Když automaticky definovaný kanál CLUSSDR nahradí ručně definovaný kanál, číslo portu (převzaté z kanálu CLUSRCVR) bude prázdné. Použijte se výchozí číslo portu a kanál selže.

Pokud existují rozdíly v konfiguraci mezi ručně definovaným kanálem CLUSSDR a odpovídající definicí kanálu CLUSRCVR , některé rozdíly se projeví okamžitě (například parametry vyrovnávání pracovní zátěže) a některé se projeví pouze při restartování kanálu (například konfigurace TLS).

Abyste se vyhnuli nejasnostem, dodržujte pokud možno následující pokyny:

- Kanály CLUSSDR definujte pouze ručně tak, aby ukazovaly na úplná úložiště.
- Pokud jste ručně definovali kanály CLUSSDR , nakonfigurujte je tak, aby se shodovaly s odpovídající definicí kanálu CLUSRCVR v přijímacím správci front.

Viz také [Práce s automaticky definovanými kanály](#).

Související pojmy

[Práce s automaticky definovanými kanály](#)

[Práce s přenosovými frontami klastru a odesílacími kanály klastru](#)

Související úlohy

[Nastavení nového klastru](#)

[Přidání správce front do klastru](#)

Témata klastru

Témata klastru jsou administrativní témata s definovaným atributem **cluster** . Informace o tématech klastru se publikují na všechny členy klastru a v kombinaci s lokálními tématy vytváří části prostoru témat, které pokrývají více správců front. Tato konfigurace umožňuje publikovat zprávy k tématu na jednom správci front, a doručení těchto zpráv do odběrů na ostatních správcích front v klastru.

Když definujete na správci front téma klastru, odešle se definice tématu klastru do správců front úplného úložiště. Úplná úložiště následně šíří definici tématu klastru na všechny správce front v klastru, čímž zpřístupní toto téma klastru vydavatelům i odběratelům ve všech správcích front klastru. Správci front,

na kterém jste vytvořili téma klastru, se říká hostitel tématu klastru. Téma klastru může následně použít libovolný správce front v klastru, ale veškeré změny v tomto tématu klastru se musí provádět na tom správci front, na kterém bylo toto téma nadefinováno (na hostiteli), načež se tyto změny rozšíří na všechny členy klastru prostřednictvím úplných úložišť.

Chcete-li získat informace o konfiguraci témat klastru pro použití *přímého směrování* nebo *směrování hostitele tématu* o dědičnosti klastrovaných témat a odběrech se zástupnými znaky, prohlédněte si téma Definování témat klastru.

Informace o příkazech používaných k zobrazení témat klastru viz související informace.

Související pojmy

Práce s administrativními tématy

Práce s odběry

Související odkazy


DISPLAYTOPIC-zobrazení

DISPLAYTPSTATUS

DISPLAYSUB-zobrazení

Výchozí objekty klastru

 Na platformě Multiplatforms jsou výchozí objekty klastru zahrnuty do sady výchozích

objektů automaticky vytvořených při definování správce front.  V systému z/OS lze výchozí definice objektů klastru nalézt v ukázkách přizpůsobení.

Poznámka: Spuštěním příkazů MQSC nebo PCF můžete změnit výchozí definice kanálů stejným způsobem jako libovolnou jinou definici kanálu. Neměňte výchozí definice front, s výjimkou SYSTEM.CLUSTER.HISTORY.QUEUE.

SYSTEM.CLUSTER.COMMAND.QUEUE

Každý správce front v klastru má lokální frontu s názvem SYSTEM.CLUSTER.COMMAND.QUEUE, která se používá k přenosu zpráv do úplného úložiště. Zpráva obsahuje nové nebo změněné informace o správci front nebo požadavky na informace o jiných správcích front. SYSTEM.CLUSTER.COMMAND.QUEUE je obvykle prázdný.

SYSTEM.CLUSTER.HISTORY.QUEUE

Každý správce front v klastru má lokální frontu s názvem SYSTEM.CLUSTER.HISTORY.QUEUE. SYSTEM.CLUSTER.HISTORY.QUEUE se používá k uložení historie informací o stavu klastru pro servisní účely.

Ve výchozím nastavení objektu je parametr SYSTEM.CLUSTER.HISTORY.QUEUE nastaven na hodnotu PUT (ENABLED). Chcete-li potlačit kolekci historie, změňte nastavení na PUT (DISABLED).

SYSTEM.CLUSTER.REPOSITORY.QUEUE

Každý správce front v klastru má lokální frontu s názvem SYSTEM.CLUSTER.REPOSITORY.QUEUE. Tato fronta se používá k uložení všech informací o úplném úložišti. Tato fronta není obvykle prázdná.

SYSTEM.CLUSTER.TRANSMIT.QUEUE

Každý správce front má definici pro lokální frontu s názvem SYSTEM.CLUSTER.TRANSMIT.QUEUE. SYSTEM.CLUSTER.TRANSMIT.QUEUE je výchozí přenosová fronta pro všechny zprávy pro všechny fronty a správce front v rámci klastrů. Výchozí přenosovou frontu pro každý odesílací kanál klastru můžete změnit na hodnotu SYSTEM.CLUSTER.TRANSMIT.ChannelName změnou atributu správce front DEFCLXQ. Nelze odstranit SYSTEM.CLUSTER.TRANSMIT.QUEUE. Používá se také k definování kontrol autorizace, zda je použita výchozí přenosová fronta SYSTEM.CLUSTER.TRANSMIT.QUEUE nebo SYSTEM.CLUSTER.TRANSMIT.ChannelName.

SYSTEM.DEF.CLUSRCVR

Každý klastr má výchozí definici kanálu CLUSRCVR s názvem SYSTEM.DEF.CLUSRCVR. Parametr SYSTEM.DEF.CLUSRCVR se používá k zadání výchozích hodnot pro všechny atributy, které neurčíte při vytváření přijímacího kanálu klastru ve správci front v klastru.

SYSTEM. DEF . CLUSSDR

Každý klastr má výchozí definici kanálu CLUSSDR s názvem SYSTEM. DEF . CLUSSDR. SYSTEM. DEF . CLUSSDR se používá k zadání výchozích hodnot pro všechny atributy, které neurčíte při vytváření odesílacího kanálu klastru ve správci front v klastru.

Související pojmy

[Práce s výchozími objekty klastru](#)

System zpráv publikování/odběru

Systém zpráv typu publikování/odběr vám umožňuje oddělit poskytovatele informací od spotřebitelů těchto informací. Odesílající a přijímající aplikace nemusí vědět nic o sobě pro informace, které mají být odeslány a přijaty.

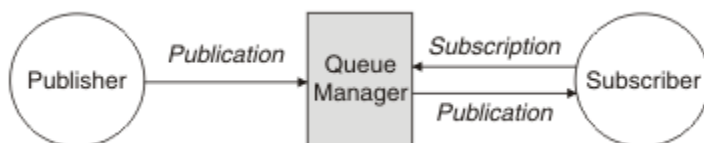
Než může aplikace IBM MQ typu point-to-point odeslat zprávu jiné aplikaci, musí o této aplikaci něco vědět. Například musí znát název fronty, do které mají být informace odeslány, a může také zadat název správce front.

Produkt IBM MQ publikování/odběr odstraňuje potřebu, aby vaše aplikace o cílové aplikaci něco věděla. Vše, co musí odesílající aplikace udělat, je toto:

- Vložte zprávu IBM MQ , která obsahuje informace požadované aplikací.
- Přiřaďte zprávu k tématu, které označuje předmět informací.
- Nechte produkt IBM MQ , aby zpracovala distribuci těchto informací.

Podobně nemusí cílová aplikace vědět nic o zdroji informací, které obdrží.

Následující obrázek ukazuje nejjednodušší systém publikování/odběru. Existuje jeden vydavatel, jeden správce front a jeden odběratel. Odběr je proveden odběratelem ve správci front, publikování je odesláno vydavatelem správci front a publikování je poté předáno správcem front odběrateli.



Obrázek 17. Jednoduchá konfigurace publikování/odběru

Typický systém publikování/odběru má více než jednoho vydavatele a více než jednoho odběratele v mnoha různých tématech a často má více než jednoho správce front. Aplikace může být vydavatelem i odběratelem.

Dalším významným rozdílem mezi systémem zpráv publikování/odběru a systémem point-to-point je, že zprávu odeslanou do fronty point-to-point zpracovává pouze jedna přijímající aplikace. Zpráva publikovaná do tématu publikování/odběru, kde více než jeden odběratel zaregistroval zájem, je zpracována každým zainteresovaným odběratelem.

Komponenty publikování/odběru

Publikování/odběr je mechanismus, pomocí kterého mohou odběratelé přijímat informace od vydavatelů ve formě zpráv. Interakce mezi vydavatelem a odběrateli jsou řízeny správci front pomocí standardních prostředků IBM MQ .

Typický systém publikování/odběru má více než jednoho vydavatele a více než jednoho odběratele v mnoha různých tématech a často má více než jednoho správce front. Aplikace může být vydavatelem i odběratelem.

Poskytovatel informací se nazývá *vydavatel*. Vydavatelé poskytují informace o předmětu, aniž by museli vědět něco o aplikacích, které se o tyto informace zajímají. Vydavatelé generují tyto informace ve formě zpráv nazývaných *publikace* , které chtějí publikovat, a definují téma těchto zpráv.

Příjemce informací se nazývají *odběratel*. Odběratelé vytvářejí *odběry*, které popisují téma, o které má odběratel zájem. Odběr tedy určuje, která publikování budou předána odběrateli. Odběratelé mohou vytvořit více odběrů a mohou přijímat informace od mnoha různých vydavatelů.

Publikované informace jsou odeslány ve zprávě IBM MQ a předmět informací je identifikován v *tématu*. Vydavatel určuje téma při publikování informací a odběratel určuje témata, o nichž chce přijímat publikování. Odběrateli jsou odesílány informace pouze o tématech, která odebírá.

Jedná se o existenci témat, která umožňují poskytovatelům a spotřebitelům informací oddělit v systému zpráv publikování/odběru tím, že se odstraní potřeba zahrnout do každé zprávy specifické místo určení, jak je vyžadováno v systému zpráv typu point-to-point.

Interakce mezi vydavatelem a odběrateli jsou všechny řízeny správcem front. Správce front přijímá zprávy od vydavatelů a odběry od odběratelů (pro řadu témat). Úlohou správce front je směřovat publikované zprávy na odběratele, kteří zaregistrovali zájem o téma zpráv.

Standardní prostředky systému IBM MQ se používají k distribuci zpráv, takže vaše aplikace mohou používat všechny funkce, které jsou k dispozici pro existující aplikace systému IBM MQ. To znamená, že trvalé zprávy můžete použít k jednorázově zajištěnému doručení a že vaše zprávy mohou být součástí transakční jednotky práce, abyste zajistili, že budou doručeny odběrateli pouze v případě, že jsou potvrzeny vydavatelem.

Vydavatelé a publikace

V produktu IBM MQ publish/subscribe je vydavatel aplikace, která zpřístupňuje informace o zadaném tématu správci front ve formě standardní zprávy IBM MQ nazvané publikování. Vydavatel může publikovat informace o více než jednom tématu.

Vydavatelé používají slovo MQPUT k vložení zprávy do dříve otevřeného tématu. Tato zpráva je publikování. Lokální správce front poté přesměruje publikování na všechny odběratele, kteří mají odběry k tématu publikování. Publikovanou zprávu může spotřebovat více než jeden odběratel.

Kromě distribuce publikací všem lokálním odběratelům, kteří mají příslušné odběry, může správce front také distribuovat publikování všem ostatním správcům front, kteří jsou k němu připojeni, a to buď přímo, nebo prostřednictvím sítě správců front, kteří mají odběratele daného tématu.

V síti publikování/odběru produktu IBM MQ může být publikační aplikace také odběratelem.

Publikování v synchronizačním bodu

Vydavatelé mohou vydávat volání MQPUT nebo MQPUT1 v synchronizačním bodu, aby zahrnuli všechny zprávy doručené odběratelům v transakci. Pokud je zadána volba MQPMO_RETAIN nebo volba doručení tématu NPMSGDLV nebo PMSGDLV s hodnotami ALL nebo ALLDUR, použije správce front interní volání MQPUT nebo MQPUT1 v synchronizačním bodu v rozsahu volání MQPUT nebo MQPUT1 vydavatele.

Informace o stavu a událostech

Publikace lze kategorizovat jako státní publikace, například jako aktuální cenu akcií, nebo jako publikace událostí, například obchod s těmito akciemi.

Publikování stavu

Státní publikace obsahují informace o aktuálním stavu něčeho, jako je cena akcií nebo aktuální skóre ve fotbalovém zápase. Pokud se něco stane (například změní se cena akcie nebo dojde ke změně skóre), předchozí informace o stavu již není potřeba, protože je nahrazena novou informací.

Odběratel bude chtít při spuštění obdržet aktuální verzi informací o stavu a při každé změně stavu mu budou odeslány nové informace.

Pokud publikování obsahuje informace o stavu, je často publikováno jako zachované publikování. Nový odběratel obvykle požaduje informace o aktuálním stavu okamžitě; odběratel nechce čekat na událost, která způsobí opětovné publikování informací. Odběratelé při přihlášení k odběru automaticky obdrží

zachované publikování tématu, pokud odběratel nepoužívá volby MQSO_PUBLICATIONS_ON_REQUEST nebo MQSO_NEW_PUBLICATIONS_ONLY.

Publikování událostí

Publikace o událostech obsahují informace o jednotlivých událostech, které se vyskytnou, jako je například obchod s některými akciemi nebo vstřelení určitého cíle. Každá událost je na ostatních událostech nezávislá.

Odběratel bude chtít přijímat informace o událostech v jejich rámci.

Zachovaná publikování

Ve výchozím nastavení je publikování po odeslání všem zainteresovaným odběratelům zrušeno. Vydavatel však může určit, že kopie publikování bude zachována, aby mohla být odeslána budoucím odběratelům, kteří zaregistrují zájem o dané téma.

Odstranění publikací poté, co byly zaslány všem zainteresovaným odběratelům, je vhodné pro informace o události, ale ne vždy je vhodné pro informace o stavu. Ponecháním zprávy nemusí noví odběratelé čekat na opětovné publikování informací, než obdrží informace o počátečním stavu. Například předplatitel s předplatným ceny akcií by okamžitě obdržel aktuální cenu, aniž by čekal na změnu ceny akcií (a tudíž by byl znovu publikován).

Správce front může pro každé téma zachovat pouze jedno publikování, takže existující zachované publikování tématu bude odstraněno, jakmile do správce front dorazí nové zachované publikování. Avšak odstranění existujícího publikování se nemusí vyskytnout synchronně s příchodem nového zachovaného publikování. Proto, je-li to možné, nesmí více než jeden vydavatel zasílat zachovaná publikování k libovolnému tématu.

Odběratelé mohou určit, že nechtějí přijímat zachovaná publikování, pomocí volby odběru MQSO_NEW_PUBLICATIONS_ONLY. Stávající odběratelé mohou požádat o zaslání duplicitních kopií zachovaných publikování.

Mohou nastat situace, kdy nebudete chtít zachovat publikování, a to i v případě informací o stavu:

- Pokud jsou všechny odběry tématu provedeny před provedením publikování v rámci daného tématu a neočekáváte nebo nepovolíte nové odběry, není nutné publikace uchovávat, protože jsou doručeny úplné sadě odběratelů při prvním publikování.
- Pokud se publikování vyskytují často, například každou sekundu, nový odběratel (nebo odběratel zotavující se ze selhání) obdrží aktuální stav téměř okamžitě po svém počátečním odběru, takže není nutné tato publikování uchovávat.
- Pokud jsou publikace velké, může se stát, že budete potřebovat značné množství úložného prostoru pro uložení zachovaného publikování pro každé téma. V prostředí s více správci front jsou uchovaná publikování uložena všemi správci front v síti, kteří mají odpovídající odběr.

Při rozhodování o tom, zda použít zachovaná publikování, zvažte, jak se odebírající aplikace zotaví ze selhání. Pokud vydavatel nepoužívá zachovaná publikování, může být nutné, aby aplikace odběratele lokálně uložila aktuální stav.

Chcete-li zajistit zachování publikování, použijte volbu MQPMO_RETAIN put-message. Je-li použita tato volba a publikování nelze zachovat, zpráva nebude publikována a volání se nezdaří s hodnotou MQRC_PUT_NOT_ZACHOVÁNO.

Pokud je zpráva zachované publikování, je to označeno vlastností zprávy MQIsRetained . Perzistence zprávy je taková, jaká byla, když byla původně publikována.

Související pojmy

[Aspekty návrhu pro zachovaná publikování v klastrech publikování/odběru](#)

Publikování v synchronizačním bodu

V produktu IBM MQ publish/subscribe může být synchronizační bod používán vydavatelem nebo interně správcem front.

Vydavatelé používají synchronizační bod, když vydávají volání MQPUT/MQPUT1 s volbou MQPMO_SYNCPOINT. Všechny zprávy doručené odběratelům se započítávají do maximálního počtu nepotvrzených zpráv v jednotce práce work.The MAXUMSGS správce front určuje tento limit. Pokud je dosaženo limitu, vydavatel obdrží kód příčiny 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_DOSAŽENÉ .

Pokud vydavatel zadá volání MQPUT/MQPUT1 s použitím volby MQPMO_NO_SYNCPOINT s volbou MQPMO_RETAIN nebo volby doručení tématu NPMSGDLV/PMSGDLV s hodnotami ALL nebo ALLDUR, správce front použije interní synchronizační body k zajištění toho, aby byly zprávy doručovány podle požadavku. Vydavatel může přijmout kód příčiny 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_DOSAŽENÉ , pokud je dosaženo limitu v rozsahu volání vydavatele MQPUT/MQPUT1 .

Odběratelé a odběry

V produktu IBM MQ publish/subscribe je odběratel aplikace, která požaduje informace o specifickém tématu od správce front v síti publikování/odběru. Odběratel může přijímat zprávy o stejných nebo různých tématech od více než jednoho vydavatele.

Odběry lze vytvořit ručně pomocí příkazu MQSC nebo pomocí aplikací. Tyto odběry jsou vydávány lokálnímu správci front a obsahují informace o publikacích, které chce odběratel přijímat:

- Téma, o které má odběratel zájem; při použití zástupných znaků lze toto téma interpretovat jako více témat.
- Volitelný řetězec výběru, který má být použit na publikované zprávy.
- Popisovač fronty (známý jako *fronta odběratele*), do které mají být umístěna vybraná publikování, a volitelné CorrelId.

Lokální správce front ukládá informace o odběru a při přijetí publikování tyto informace skenuje a zjišťuje, zda existuje odběr, který odpovídá tématu publikování a řetězci výběru. Pro každý odpovídající odběr správce front směřuje publikování do fronty odběratele odběratele. Informace, které správce front ukládá o odběrech, lze zobrazit pomocí příkazů DIS SUB a DIS SBSTATUS.

Odběr je odstraněn pouze v případě, že dojde k jedné z následujících událostí:

- Odběratel se odhlásí pomocí volání MQCLOSE (pokud byl odběr proveden netrvale).
- Odběr vyprší.
- Odběr odstraní administrátor systému pomocí příkazu DELETE SUB.
- Aplikace odběratele skončí (pokud byl odběr proveden netrvale).
- Správce front je zastaven nebo restartován (pokud byl odběr proveden netrvale).

Při získávání zpráv použijte příslušné volby pro volání MQGET. Pokud vaše aplikace zpracovává pouze zprávy pro jeden odběr, měli byste použít minimálně `get-by-correlid`, jak ukazuje ukázkový program `C amqssbxa.c` a nespravovaný odběratel MQ. Objekt **CorrelId** , který má být použit, je vrácen z modulu MQSUB v modulu MQSD.SubCorrelId pole.

Související pojmy

Klonované a sdílené odběry

Související odkazy

Příklady, jak definovat vlastnost sharedSubscription

Spravované fronty a publikování/odběr

Při vytváření odběru můžete zvolit použití spravovaného řazení do fronty. Používáte-li spravované řazení do fronty, bude při vytváření odběru automaticky vytvořena fronta odběrů. Spravované fronty jsou automaticky vyřízeny v souladu s trvanlivostí odběru. Použití spravovaných front znamená, že se nemusíte starat o vytváření front pro příjem publikování a všechna nespoteřovaná publikování jsou automaticky odebrána z front odběratele, pokud je připojení k trvalému odběru uzavřeno.

Pokud aplikace nemusí jako svou frontu odběratele používat konkrétní frontu, místo určení pro publikování, která přijímá, může používat *spravované odběry* s použitím volby odběru MQSO_MANAGED.

Pokud vytvoříte spravovaný odběr, vrátí správce front odběrateli popisovač objektu pro frontu odběratele, kterou správce front vytvoří, kde budou přijímána publikování. Důvodem je skutečnost, že *spravované odběry* jsou takové odběry, které produkt IBM MQ zpracovává. Popisovač objektu fronty bude vrácen, což vám umožní procházet, získat nebo zjistit frontu (není možné vložit do nebo nastavit atributy spravované fronty, pokud vám nebyl výslovně udělen přístup k dočasným dynamickým frontám).

Trvanlivost odběru určuje, zda spravovaná fronta zůstane zachována i po přerušení připojení odebírající aplikace ke správci front.

Spravované odběry jsou užitečné zejména při použití s netrvalými odběry, protože po ukončení připojení aplikace by nespoteřované zprávy jinak zůstaly ve frontě odběratele, která zabírá místo ve správci front, neomezeně dlouho. Pokud používáte spravovaný odběr, bude spravovaná fronta dočasnou dynamickou frontou a jako taková bude odstraněna spolu s nespoteřovanými zprávami, když je připojení přerušeno z některého z následujících důvodů:

- Použije se MQCLOSE s MQCO_REMOVE_SUB a spravovaný Hobj je zavřený.
- Došlo ke ztrátě připojení k aplikaci s použitím netrvalého odběru (MQSO_NON_TRVALÉ).
- předplatné je odebráno, protože vypršela jeho platnost a spravovaný Hobj je uzavřen.

Spravované odběry lze také použít s trvalými odběry, ale je možné, že budete chtít ponechat nespoteřované zprávy ve frontě odběratele, aby je bylo možné načíst při opětovném otevření připojení. Z tohoto důvodu mají spravované fronty pro trvalé odběry podobu trvalé dynamické fronty a zůstanou zachovány po přerušení připojení odebírající aplikace ke správci front.

Chcete-li použít trvalou dynamickou spravovanou frontu, můžete nastavit vypršení platnosti odběru, aby i po přerušení připojení fronta nadále existovala, ale nebude existovat neomezeně dlouho.

Pokud spravovanou frontu odstraníte, obdržíte chybovou zprávu.

Vytvořené spravované fronty jsou pojmenovány čísly na konci (časová razítka), takže každá z nich je jedinečná.

Trvanlivost předplatného

Odběry lze konfigurovat jako trvalé nebo netrvalé. Trvanlivost odběrů určuje, co se stane s odběry, když se odebírající aplikace odpojí od správce front.

Trvalé odběry

Trvalé odběry nadále existují i po zavření připojení odebírající aplikace ke správci front. Je-li odběr trvalý, po odpojení odebírající aplikace zůstává odběr na místě a může být použit odebírající aplikací při opětovném připojení s požadavkem na odběr znovu pomocí **SubName**, který byl vrácen při vytvoření odběru.

Při trvalém přihlášení k odběru název odběru (**SubName**) je vyžadováno. Názvy odběrů musí být v rámci správce front jedinečné, aby je bylo možné použít k identifikaci odběru. Tento způsob identifikace je nezbytný při zadávání odběru, který chcete obnovit, pokud jste buď úmyslně uzavřeli připojení k odběru (pomocí volby MQCO_KEEP_SUB), nebo jste byli odpojeni od správce front. Můžete obnovit existující odběr pomocí volání MQSUB s volbou MQSO_RESUME. Názvy odběrů se zobrazí také v případě, že použijete příkaz DISPLAY SBSTATUS s hodnotou SUBTYPE ALL nebo ADMIN.

Pokud aplikace již nevyžaduje trvalý odběr, lze ji odebrat pomocí volání funkce MQCLOSE s volbou MQCO_REMOVE_SUB nebo ji lze odstranit ručně pomocí příkazu MQSC DELETE SUB.

Pomocí atributu tématu **DURSUB** můžete určit, zda mají být v tématu provedeny trvalé odběry.

Při návratu z volání MQSUB s použitím volby MQSO_RESUME je vypršení platnosti odběru nastaveno na původní vypršení platnosti odběru, nikoli na zbývající dobu vypršení platnosti.

Správce front pokračuje v odesílání publikací s cílem uspokojit trvalý odběr i v případě, že aplikace odběratele není připojena. To vede k sestavení zpráv ve frontě odběratele. Nejjednodušším způsobem, jak se tomuto problému vyhnout, je použít trvalé předplatné, kdykoli je to vhodné. Je-li však nutné používat trvalé odběry, je možné se vyhnout sestavení zpráv, pokud se odběratel přihlásí k odběru pomocí volby Zadržené publikace. Odběratel pak může řídit, kdy obdrží publikování pomocí volání MQSUBRQ.

Dočasné odběry

Dočasné odběry existují pouze v případě, že připojení odebírající aplikace ke správci front zůstává otevřené. Tento odběr je odebrán, když se odběratelská aplikace odpojí od správce front, ať už úmyslně nebo kvůli ztrátě připojení. Po zavření připojení jsou informace o odběru odebrány ze správce front a již nejsou zobrazeny, pokud zobrazujete odběry pomocí příkazu DISPLAY SBSTATUS. Do fronty odběratele nejsou vloženy žádné další zprávy.

Co se stane s nespoteřovaným publikováním ve frontě odběratele pro netrvalé odběry, je určeno následujícím způsobem.

- Pokud odebírající aplikace používá [spravované místo určení](#), všechna publikování, která nebyla využita, budou automaticky odebrána.
- Pokud odebírající aplikace poskytuje manipulátor své vlastní frontě odběratele při přihlášení k odběru, nespoteřované zprávy se neodeberou automaticky. Je odpovědností aplikace vymazat frontu, je-li to vhodné. Pokud je fronta sdílena více než jedním odběratelem nebo jinými aplikacemi dvoubodového spojení, nemusí být vhodné frontu zcela vymazat.

Ačkoli to není vyžadováno pro dočasné odběry, je název odběru, je-li zadán, používán správcem front. Názvy odběrů musí být v rámci správce front jedinečné, aby je bylo možné použít k identifikaci odběru.

Související pojmy

[Klonované a sdílené odběry](#)

Související úlohy

[Použití sdílených odběrů JMS 2.0](#)

Související odkazy

[Příklady, jak definovat vlastnost sharedSubscription](#)

Řetězce výběru

Řetězec výběru je výraz, který se použije na publikování, aby se určilo, zda odpovídá odběru. Řetězce výběru mohou obsahovat zástupné znaky.

Když se přihlásíte k odběru, můžete kromě určení tématu určit řetězec výběru pro výběr publikování podle jejich vlastností zprávy.

Výběrový řetězec je vyhodnocen proti zprávě, kterou vložil vydavatel před tím, než je upraven pro doručení každému odběrateli. Při použití polí v řetězci výběru, která mohou být upravena jako součást operace publikování, buďte opatrní. Například pole MQMD UserIdentifier, MsgIda CorrelId.

Řetězce výběru by neměly odkazovat na žádné pole vlastností zprávy přidané správcem front jako součást operace publikování (viz [Vlastnosti zprávy publikování/odběru](#)), s výjimkou vlastnosti zprávy MQTopicString, která obsahuje řetězec tématu pro publikování.

Související pojmy

[Pravidla a omezení řetězce výběru](#)

Témata

Téma je předmětem informací publikovaných ve zprávě publikování/odběru.

Zprávy v dvoubodových systémech jsou odesílány na specifickou cílovou adresu. Zprávy v systémech publikování/odběru založených na předmětu se odesílají odběratelům na základě předmětu, který popisuje obsah zprávy. V systémech založených na obsahu jsou zprávy odesílány odběratelům na základě obsahu samotné zprávy.

Systém publikování/odběru IBM MQ je systém publikování/odběru založený na předmětu. Vydavatel vytvoří zprávu a publikuje ji pomocí řetězce tématu, který nejlépe odpovídá předmětu publikace. Chcete-li přijímat publikování, odběratel vytvoří odběr se vzorem odpovídajícím řetězci tématu pro výběr témat publikování. Správce front doručuje publikování odběratelům, kteří mají odběry odpovídající tématu publikování, a jsou autorizováni přijímat publikování. Článek [“Řetězce tématu”](#) na stránce 65 popisuje syntaxi řetězců témat, které identifikují předmět publikace. Odběratelé také vytvářejí řetězce témat pro výběr témat, která mají být přijímána. Řetězce témat, které vytvoří odběratelé, mohou obsahovat

dvě alternativní schémata zástupných znaků, která se budou shodovat s řetězcí témat v publikacích. Porovnávání vzorů je popsáno v tématu [“Schémata zástupných znaků”](#) na stránce 66.

V případě publikování/odběru na základě předmětu jsou vydavatelé nebo administrátoři odpovědní za klasifikaci předmětů do témat. Předměty jsou obvykle uspořádány hierarchicky do stromů témat s použitím znaku ' / ' k vytvoření dílčích témat v řetězcí tématu. Příklady stromů témat viz [“Stromy témat”](#) na stránce 72 . Témata jsou uzly ve stromu témat. Témata mohou být koncové uzly bez dalších dílčích témat nebo přechodné uzly s dílčími tématy.

Souběžně s uspořádáním předmětů do hierarchického stromu témat můžete přidružit témata k objektům administrativních témat. Atributy přiřadíte k tématu, například zda je téma distribuováno v klastru, jeho přidružením k objektu administrativního tématu. Přidružení se provede pojmenováním tématu pomocí atributu TOPICSTR objektu administrativního tématu. Pokud explicitně nepřidružíte objekt administrativního tématu k tématu, téma zdědí atributy svého nejbližšího předchůdce ve stromu témat, který jste *přiřadili* k objektu administrativního tématu. Pokud jste vůbec nedefinovali žádná nadřazená témata, zdědí je od SYSTEM . BASE . TOPIC. Objekty administrativních témat jsou popsány v tématu [“Objekty administrativních témat”](#) na stránce 73.

Poznámka: I když zdědíte všechny atributy tématu z SYSTEM . BASE . TOPIC, definujte kořenové téma pro témata, která přímo dědí ze systému SYSTEM . BASE . TOPIC. Například v prostoru tématu stavů USA, USA/Alabama USA/Alaskaatd. je kořenovým tématem téma USA . Hlavním účelem kořenového tématu je vytvořit diskrétní, nepřekrývající se prostory témat, aby se zabránilo publikování, která by se shodovala s nesprávným odběry. To také znamená, že můžete změnit atributy svého kořenového tématu tak, aby ovlivňovaly celý prostor tématu. Můžete například nastavit název atributu **CLUSTER** .

Odkazujete-li na téma jako na vydavatele nebo odběratele, máte možnost dodat řetězec tématu nebo odkazovat na objekt tématu. Nebo můžete provést obojí. V takovém případě řetězec tématu, který zadáte, definuje dílčí téma objektu tématu. Správce front identifikuje téma připojením řetězce tématu k předponě řetězce tématu uvedené v objektu tématu a vložením dalšího řetězce ' / ' mezi dva řetězce tématu, například *řetězec řetězec/objekt tématu*. [“Kombinování řetězců témat”](#) na stránce 70 to dále popisuje. Výsledný řetězec tématu se používá k identifikaci tématu a jeho přidružení k objektu administrativního tématu. Objekt administrativního tématu nemusí být nutně stejný objekt tématu jako objekt tématu odpovídající hlavnímu tématu.

V publikování/odběru založeném na obsahu můžete definovat, které zprávy chcete přijmout, zadáním řetězců výběru, které prohledávají obsah každé zprávy. Produkt IBM MQ poskytuje intermediační formu publikování/odběru na základě obsahu pomocí selektorů zpráv, které skenují vlastnosti zprávy, a nikoli úplný obsah zprávy, viz [Selektory](#). Archetypální použití selektorů zpráv je přihlásit se k odběru tématu a poté kvalifikovat výběr pomocí číselné vlastnosti. Selektor vám umožňuje určit, že máte zájem o hodnoty pouze v určitém rozsahu; něco, co nemůžete dělat pomocí znakových nebo tématických zástupných znaků. Pokud potřebujete filtrovat na základě úplného obsahu zprávy, musíte použít IBM Integration Bus.

Řetězce tématu

Informace o popisku, které publikujete jako téma pomocí řetězce tématu. Přihlaste se k odběru skupin témat pomocí znakových řetězců témat nebo řetězců témat se zástupnými znaky.

Témata

Řetězec tématu je znakový řetězec, který identifikuje téma zprávy publikování/odběru. Při vytváření řetězce tématu můžete použít libovolné znaky, které se vám líbí.



Tři znaky mají v produktu IBM WebSphere MQ 7 publikování/odběr speciální význam. Jsou povoleny kdekoli v řetězcí tématu, ale používají se s opatrností. Použití speciálních znaků je vysvětleno v části [“Schéma zástupných znaků na základě tématu”](#) na stránce 67.

dopředné lomítko (/)

Oddělovač úrovně tématu. Pomocí znaku ' / ' můžete strukturovat téma do stromu témat.

Pokud můžete, vyvarujte se prázdných úrovní tématu ' / / '. Ty odpovídají uzlům v hierarchii témat bez řetězce tématu. Úvodní nebo koncová hodnota ' / ' v řetězci tématu odpovídá počátečnímu nebo koncovému prázdnému uzlu a měli byste se také vyvarovat.

Znak přepočtu klíče (#)

Používá se v kombinaci s ' / ' k vytvoření zástupného znaku pro více úrovní v odběrech. V řetězcích témat používaných k pojmenování publikovaných témat buďte opatrní při použití ' # ' sousedících s ' / ' . [“Příklady řetězců témat”](#) na stránce 66 ukazuje rozumné použití ' # ' .

Řetězce ' . . . / # / . . . ' , ' # / . . . ' a ' . . . / # ' mají v řetězcích tématu odběru speciální význam. Řetězce odpovídají všem tématům na jedné nebo více úrovních v hierarchii témat. Pokud jste tedy vytvořili téma s jednou z těchto posloupností, nemohli jste se k němu přihlásit, aniž byste se přihlásili k odběru všech témat na více úrovních v hierarchii témat.

Znaménko plus (+)

Používá se v kombinaci s ' / ' k vytvoření zástupného znaku na jedné úrovni v odběrech. V řetězcích témat používaných k pojmenování publikovaných témat buďte opatrní při použití ' + ' sousedících s ' / ' .

Řetězce ' . . . / + / . . . ' , ' + / . . . ' a ' . . . / + ' mají v řetězcích tématu odběru speciální význam. Řetězce odpovídají všem tématům na jedné úrovni v hierarchii témat. Pokud jste tedy vytvořili téma s jednou z těchto posloupností, nemohli jste se k němu přihlásit, aniž byste se přihlásili k odběru všech témat na jedné úrovni v hierarchii témat.

Příklady řetězců témat

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

Související odkazy

[TOPIC](#)

Schémat zástupných znaků

K přihlášení k odběru více témat se používají dvě schémata zástupných znaků. Volba schématu je volba předplatného.

MQSO_WILDCARD_TOPIC

Vyberte témata, která se mají přihlásit k odběru pomocí schématu zástupného znaku založeného na tématu.

Jedná se o předvolbu, pokud není explicitně vybráno žádné schéma se zástupnými znaky.

MQSO_WILDCARD_CHAR

Vyberte témata, která se mají přihlásit k odběru pomocí znakově orientovaného schématu se zástupnými znaky.

Nastavte schéma zadáním parametru **wschema** v příkazu DEFINE SUB. Další informace viz [DEFINE SUB](#).

Poznámka: Odběry, které byly vytvořeny před produktem IBM WebSphere MQ 7.0 , vždy používají schéma se zástupnými znaky na základě znaků.

Příklady

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

Schéma zástupných znaků na základě tématu

Zástupné znaky na základě témat umožňují odběratelům přihlásit se k odběru více než jednoho tématu najednou.

Zástupné znaky založené na tématu jsou výkonnou funkcí systému témat v produktu IBM MQ publish/subscribe. Zástupné znaky více úrovní a zástupné znaky jedné úrovně lze použít pro odběry, nemohou ale být použity v rámci tématu vydavatelem zprávy.

Schéma se zástupnými znaky na základě tématu vám umožňuje vybrat publikování seskupená podle úrovně tématu. Pro každou úroveň v hierarchii témat můžete určit, zda se má řetězec v odběru pro danou úroveň tématu přesně shodovat s řetězcem v publikování či nikoli. Například odběr IBM/+/Results vybere všechna témata,

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

Existují dva typy zástupných znaků.

Víceúrovňový zástupný znak

- Zástupný znak pro více úrovní se používá v odběrech. Při použití v publikaci se s ním zachází jako s literálem.
- Zástupný znak s více úrovněmi '#' se používá pro shodu s libovolným počtem úrovní v rámci tématu. Pokud se například pomocí ukázkového stromu témat přihlásíte k odběru produktu 'USA/Alaska/#', obdržíte zprávy o tématech 'USA/Alaska' a 'USA/Alaska/Juneau'.
- Zástupný znak s více úrovněmi může představovat nula nebo více úrovní. Proto se 'USA/#' může také shodovat s singulárním 'USA', kde '#' představuje nulové úrovně. Oddělovač úrovně tématu nemá v tomto kontextu žádný význam, protože neexistují žádné úrovně, které by bylo možné oddělit.
- Zástupný znak pro více úrovní je platný pouze v případě, že je zadán samostatně nebo vedle oddělovacího znaku úrovně tématu. Proto '#' a 'USA/#' jsou platná témata, kde je znak '#' považován za zástupný znak. Ačkoli je 'USA#' také platný řetězec tématu, není znak '#' považován za zástupný znak a nemá žádný speciální význam. Další informace viz [“Pokud zástupné znaky založené na tématu nejsou zástupné znaky” na stránce 69.](#)

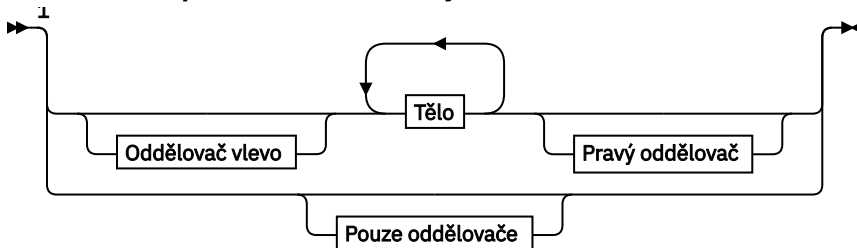
Zástupný znak na jedné úrovni

- V odběrech se používá jeden zástupný znak. Při použití v publikaci se s ním zachází jako s literálem.
- Zástupný znak na jedné úrovni '+' odpovídá jedné a pouze jedné úrovni tématu. Například 'USA/+' odpovídá 'USA/Alabama', ale ne 'USA/Alabama/Auburn'. Vzhledem k tomu, že zástupný znak na jedné úrovni odpovídá pouze jedné úrovni, 'USA/+' neodpovídá 'USA'.
- Zástupný znak na jedné úrovni lze použít na libovolné úrovni stromu témat a ve spojení se zástupným znakem na více úrovních. Vedle oddělovače úrovně tématu musí být uveden zástupný znak na jedné úrovni, s výjimkou případu, kdy je určen samostatně. Proto '+' a 'USA/+' jsou platná témata, kde je znak '+' považován za zástupný znak. Ačkoli je 'USA+' také platný řetězec tématu, není znak '+' považován za zástupný znak a nemá žádný speciální význam. Další informace viz [“Pokud zástupné znaky založené na tématu nejsou zástupné znaky” na stránce 69.](#)

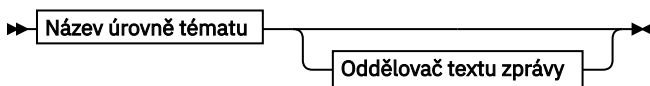
Syntaxe pro schéma se zástupnými znaky založené na tématu neobsahuje žádné řídicí znaky. To, zda jsou '#' a '+' považovány za zástupné znaky, či nikoli, závisí na jejich kontextu. Další informace viz [“Pokud zástupné znaky založené na tématu nejsou zástupné znaky” na stránce 69.](#)

Poznámka: Začátek a konec řetězce tématu je zpracován zvláštním způsobem. Pomocí '\$' označte konec řetězce, pak '\$#/...' je víceúrovňový zástupný znak a '\$#/...' je prázdný uzel v kořenovém adresáři následovaný zástupným znakem pro více úrovní.

Řetězec zástupného znaku založený na tématu



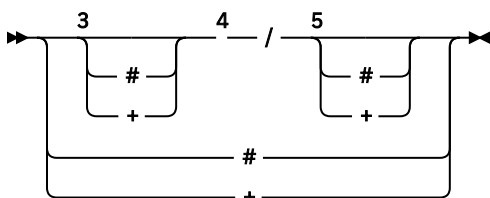
Tělo



Název úrovně tématu



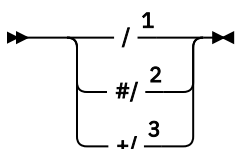
Pouze oddělovače



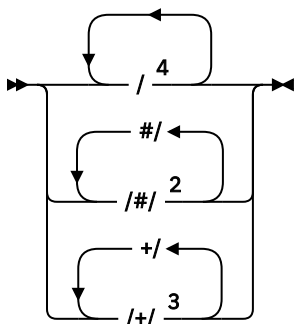
Poznámky:

- ¹ Řetězec tématu s nulovou nebo nulovou délkou je neplatný.
- ² Doporučuje se nepoužívat v řetězcích názvů úrovně *, ?, % pro kompatibilitu mezi schémata se zástupnými znaky založenými na znaku a na tématu.
- ³ Tyto případy jsou ekvivalentní vzoru *levý oddělovač*.
- ⁴ / bez zástupných znaků odpovídá jednomu prázdnému tématu.
- ⁵ Tyto případy jsou ekvivalentní vzoru *pravý oddělovač*.
- ⁶ Odpovídá každému tématu.
- ⁷ Odpovídá každému tématu, kde je pouze jedna úroveň.

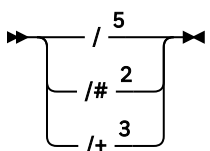
Levý oddělovač



Oddělovač těla



Pravý oddělovač



Poznámky:

- 1 Řetězec tématu začíná prázdným tématem.
- 2 Odpovídá nule nebo více úrovním. Více řetězců shody s více úrovněmi má stejný efekt jako jeden řetězec shody s více úrovněmi.
- 3 Odpovídá přesně jedné úrovni.
- 4 // je prázdné téma-objekt tématu bez řetězce tématu.
- 5 Řetězec tématu končí prázdným tématem.

Pokud zástupné znaky založené na tématu nejsou zástupné znaky

Zástupné znaky '+' a '#' nemají žádný speciální význam, pokud jsou smíchány s jinými znaky (včetně sebe) na úrovni tématu.

To znamená, že témata, která obsahují znaky '+' nebo '#' spolu s dalšími znaky na úrovni tématu, mohou být publikována.

Zvažte například následující dvě témata:

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

V prvním příkladu jsou znaky '+' a '#' považovány za zástupné znaky, a proto nejsou platné v řetězci tématu, do kterého mají být publikovány, ale jsou platné v odběru.

Ve druhém příkladu nejsou znaky '+' a '#' považovány za zástupné znaky, a proto může být řetězec tématu publikován i přihlášen k odběru.

Příklady

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

Schéma zástupných znaků založené na znacích

Znakově orientované schéma se zástupnými znaky vám umožňuje vybrat témata na základě tradičního porovnávání znaků.

Můžete vybrat všechna témata na více úrovních v hierarchii témat pomocí řetězce '*'. Použití '*' ve znakově orientovaném schématu se zástupnými znaky je ekvivalentní použití řetězce se zástupnými znaky založenými na tématu '#'

'x*/y' je ekvivalentem 'x#/y' ve schématu založeném na tématu a vybírá všechna témata v hierarchii témat mezi úrovněmi 'x a y', kde 'x' a 'y' jsou názvy témat, které nejsou v sadě úrovní vrácených zástupným znakem.

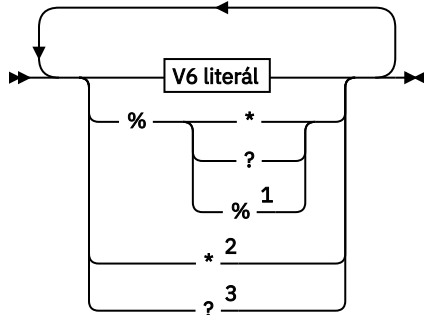
'/+' ve schématu založeném na tématu nemá přesný ekvivalent ve znakově orientovaném schématu. 'IBM*/Results' také vybere 'IBM/Patents/Software/Results'. Pouze v případě, že sada názvů témat na každé úrovni hierarchie je jedinečná, můžete vždy vytvářet dotazy se dvěma schématy, která poskytují identické shody.

Používá se obecně, '*' a '?' ve znakově orientovaném schématu nemají žádné ekvivalenty ve schématu založeném na tématu. Schéma založené na tématu neprovádí částečné porovnávání pomocí zástupných

znaků. Odběr se zástupnými znaky ve znacích ' IBM/*ware/Results ' nemá ekvivalent založený na tématu.

Poznámka: Shody používající zástupné znaky jsou pomalejší než shody používající odběry založené na tématu.

Řetězec zástupných znaků založený na znacích



V6 literál

► Jakýkoli znak Unicode kromě *,? a% ◄

Poznámky:

- ¹ Znamená změnu významu následujícího znaku tak, aby byl považován za literál. Za '%' musí následovat řetězec '*', '?' nebo '%'. Viz téma "Příklady řetězců témat" na stránce 66.
- ² Znamená shodu s žádným nebo více znaky v odběru.
- ³ Znamená, že se přesně shoduje s jedním znakem v předplatném.

Příklady

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

Kombinování řetězců témat

Při vytváření odběrů nebo otevírání témat, abyste do nich mohli publikovat zprávy, lze řetězec tématu vytvořit kombinací dvou samostatných řetězců dílčích témat nebo "dílčích témat". Jedno dílčí téma je poskytováno aplikací nebo administrativním příkazem jako řetězec tématu a druhé je řetězec tématu přidružený k objektu tématu. Můžete použít buď podtéma jako řetězec tématu samostatně, nebo je zkombinovat do nového názvu tématu.

Pokud například definujete odběr pomocí příkazu MQSC **DEFINE SUB**, příkaz může jako atribut použít buď **TOPICSTR** (řetězec tématu), nebo **TOPICOBJ** (objekt tématu), nebo obojí dohromady. Je-li zadán pouze řetězec **TOPICOBJ**, použije se jako řetězec tématu řetězec tématu přidružený k tomuto objektu tématu. Je-li uveden pouze parametr **TOPICSTR**, použije se jako řetězec tématu. Pokud jsou poskytnuty obě, jsou zřetězeny tak, aby tvořily jeden řetězec tématu ve tvaru **TOPICOBJ / TOPICSTR**, kde **TOPICOBJ** nakonfigurovaný řetězec tématu je vždy první a dvě části řetězce jsou vždy odděleny znakem "/" .

Podobně v programu MQI je úplný název tématu vytvořen produktem MQOPEN. Skládá se ze dvou polí používaných ve voláních MQI publikování/odběru v uvedeném pořadí:

1. Atribut **TOPICSTR** objektu tématu, pojmenovaný v poli **ObjectName** .
2. Parametr **ObjectString** definující dílčí téma poskytované aplikací.

Výsledný řetězec tématu je vrácen v parametru **ResObjectString** .

Tato pole jsou považována za přítomná, pokud první znak každého pole není mezerový nebo nulový znak a délka pole je větší než nula. Pokud je přítomno pouze jedno z polí, použije se jako název tématu beze změny. Pokud žádné z polí nemá hodnotu, volání selže s kódem příčiny

MQRC_UNKNOWN_OBJECT_NAME nebo MQRC_TOPIC_STRING_ERROR , pokud není úplný název tématu platný.

Jsou-li přítomna obě pole, je mezi dva prvky výsledného kombinovaného názvu tématu vložen znak "/" .

V následující tabulce jsou uvedeny příklady zřetězení řetězců témat:

<i>Tabulka 2. Příklady zřetězení řetězců témat</i>			
TOPICSTR objektu tématu	Řetězec tématu poskytnutý aplikací nebo příkazem DEFINE SUB	Úplný název tématu	Komentář
Fotbal/výsledky	' '	Fotbal/výsledky	Parametr TOPICSTR objektu tématu se používá samostatně.
' '	Fotbal/výsledky	Fotbal/výsledky	ObjectString/TOPICSTR se používá samostatně.
Fotbalové hřiště	Skóre	Fotbal/výsledky	Znak "/" se přidá do bodu zřetězení.
Fotbalové hřiště	/Skóre	Fotbalové//Skóre	Mezi těmito dvěma řetězci je vytvořen 'prázdný uzel'. Liší se od "Fotbalu/skóre".
/Fotbalové hřiště	Skóre	/Fotbalové hřiště/skóre	Téma začíná 'prázdným uzlem'. Liší se od "Fotbalu/skóre".

Znak "/" je považován za speciální znak poskytující strukturu pro úplný název tématu v souboru "Stromy témat" na stránce 72. Znak "/" nesmí být použit z žádné jiné příčiny, protože je ovlivněna struktura stromu témat. Téma "/Football" není stejné jako téma "Football".

Poznámka: Pokud při vytváření odběru použijete objekt tématu, bude hodnota řetězce tématu objektu tématu v odběru stanovena v definovaném čase. Žádná další změna objektu tématu neovlivní řetězec tématu, pro který je definován odběr.

Zástupné znaky v řetězcích témat

Následující zástupné znaky jsou speciální znaky:

- znak plus (+)
- znak čísla (#)
- hvězdička (*)
- otazník (?)

Zástupné znaky mají speciální význam pouze v případě, že je používá odběr. Tyto znaky nejsou při použití na jiném místě považovány za neplatné, je však třeba se ujistit, že chápete, jak se používají, a při publikování nebo definování objektů tématu byste raději tyto znaky v řetězcích témat nepoužívali.

Publikujete-li v řetězci tématu s řetězcem # nebo + smíchaným s jinými znaky (včetně sebe) v rámci úrovně tématu, lze řetězec tématu přihlásit k odběru pomocí jednoho ze schémat se zástupnými znaky.

Pokud publikujete v řetězci tématu s # nebo + jako jediným znakem mezi dvěma znaky / , nemůže být řetězec tématu explicitně přihlášen k odběru aplikací pomocí schématu se zástupnými znaky MQSO_WILDCARD_TOPIC. Tato situace vede k tomu, že aplikace získává více publikací, než se očekávalo.

V řetězci tématu definovaného objektu tématu byste neměli používat zástupný znak. Pokud tak učiníte, bude se s tímto znakem zacházet jako s literálovým znakem v případě, že objekt používá vydavatel, a jako se zástupným znakem v případě, že jej používá odběr. To může vést ke zmatku.

Příklad úseku kódu

Tento úsek kódu extrahovaný z ukázkového programu [Příklad 2: Vydavatel na téma proměnné](#) kombinuje objekt tématu s řetězcem tématu proměnné:

```
MQOD td = {MQOD_DEFAULT}; /* Object Descriptor */
td.ObjectType = MQOT_TOPIC; /* Object is a topic */
td.Version = MQOD_VERSION_4; /* Descriptor needs to be V4 */
strcpy(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);
```

Stromy témat

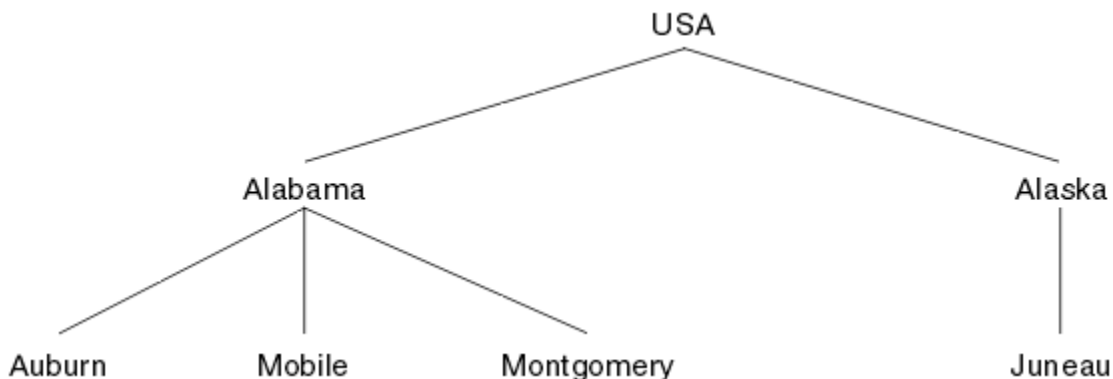
Každé téma, které definujete, je prvkem, neboli uzlem, stromu témat. Strom témat může být prázdný, aby mohl začínat nebo obsahovat témata, která byla dříve definována pomocí příkazů MQSC nebo PCF. Nové téma můžete definovat buď pomocí příkazů pro vytvoření tématu, nebo zadáním tématu poprvé v publikování či odběru.

Ačkoli můžete k definování řetězce tématu použít libovolný znakový řetězec, doporučuje se vybrat řetězec tématu, který se vejde do hierarchické stromové struktury. Promyšlený návrh stings témat a tématických stromů vám může pomoci s následujícími operacemi:

- Přihlášení k odběru více témat.
- Vytváření zásad zabezpečení.

Ačkoli můžete vytvořit strom témat jako plochou, lineární strukturu, je lepší sestavit strom témat v hierarchické struktuře s jedním nebo více kořenovými tématy. Další informace o plánování zabezpečení a tématech naleznete v tématu [Publikovat/odebírat zabezpečení](#).

Obrázek 18 na stránce 72 ukazuje příklad stromu témat s jedním kořenovým tématem.



Obrázek 18. Příklad stromu témat

Každý znakový řetězec na obrázku představuje uzel ve stromu témat. Úplný řetězec tématu je vytvořen agregací uzlů z jedné nebo více úrovní ve stromu témat. Úrovně jsou odděleny znakem "/". Formát plně určeného řetězce tématu je: "root/level2/level3".

Platná témata ve stromu témat zobrazeném v souboru [Obrázek 18 na stránce 72](#) jsou:

- "Spojené státy"
- "USA/Alabama"
- "USA/Aljaška"
- "USA/Alabama/Auburn"

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

Při návrhu řetězců témat a stromů témat nezapomeňte, že správce front neinterpretuje samotný řetězec tématu ani se z něj nepokouší odvodit význam. Jednoduše používá řetězec tématu k odeslání vybraných zpráv odběratelům daného tématu.

Na konstrukci a obsah stromu témat se vztahují následující zásady:

- Počet úrovní ve stromu témat není nijak omezen.
- Délka názvu úrovně ve stromu témat není omezena.
- Může existovat libovolný počet "kořenových" uzlů; to znamená, že může existovat libovolný počet stromů témat.

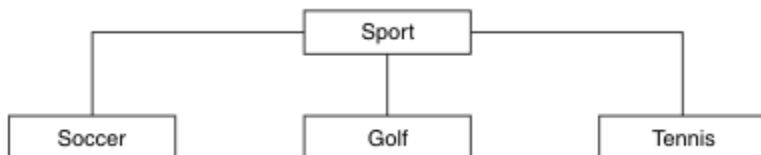
Související úlohy

Snížení počtu nechtěných témat ve stromu témat

Objekty administrativních témat

Pomocí objektu administrativního tématu můžete k tématům přiřadit specifické, jiné než výchozí atributy.

Obrázek 19 na stránce 73 ukazuje, jak lze téma vysoké úrovně produktu Sport rozdělené do samostatných témat pokrývajících různé sporty vizualizovat jako strom témat:



Obrázek 19. Vizualizace stromu témat

Obrázek 20 na stránce 73 ukazuje, jak lze dále rozdělit strom témat, aby se oddělily různé typy informací o jednotlivých sportech:



Obrázek 20. Rozšířený strom témat

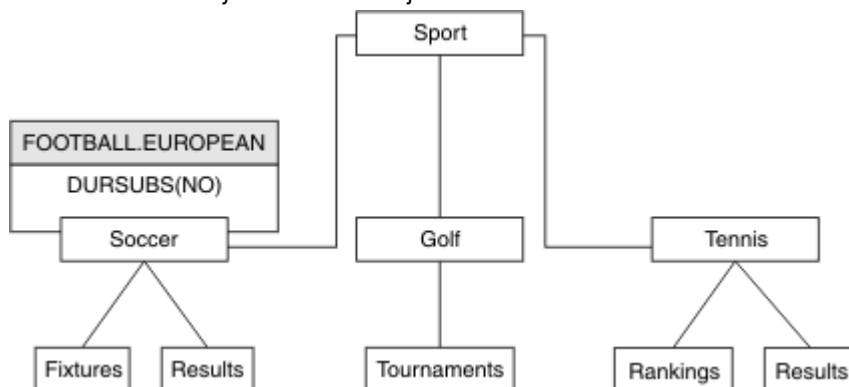
Chcete-li vytvořit zobrazený strom témat, není třeba definovat žádné objekty administrativních témat. Každý z uzlů v tomto stromu je definován řetězcem tématu vytvořeným v operaci publikování nebo odběru. Každé téma ve stromu dědí své atributy od svého nadřazeného prvku. Atributy jsou zděděny z nadřazeného objektu tématu, protože standardně jsou všechny atributy nastaveny na ASPARENT. V tomto příkladu má každé téma stejné atributy jako téma Sport. Téma Sport nemá žádný objekt administrativního tématu a dědí jeho atributy od `SYSTEM.BASE.TOPIC`.

Všimněte si, že není dobrým zvykem udělovat oprávnění pro uživatele, kteří nejsou uživateli mqm, v kořenovém uzlu stromu témat, což je `SYSTEM.BASE.TOPIC`, protože oprávnění jsou zděděná, ale nelze je omezit. Proto tím, že dáte oprávnění na této úrovni, dáváte oprávnění celému stromu. Měli byste udělit oprávnění na nižší úrovni tématu v hierarchii.

Objekty administrativních témat lze použít k definování specifických atributů pro konkrétní uzly ve stromu témat. V následujícím příkladu je objekt administrativního tématu definován tak, aby nastavil vlastnost trvalých odběrů DURSUB tématu fotbal na hodnotu NO:

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

Strom témat lze nyní vizualizovat jako:



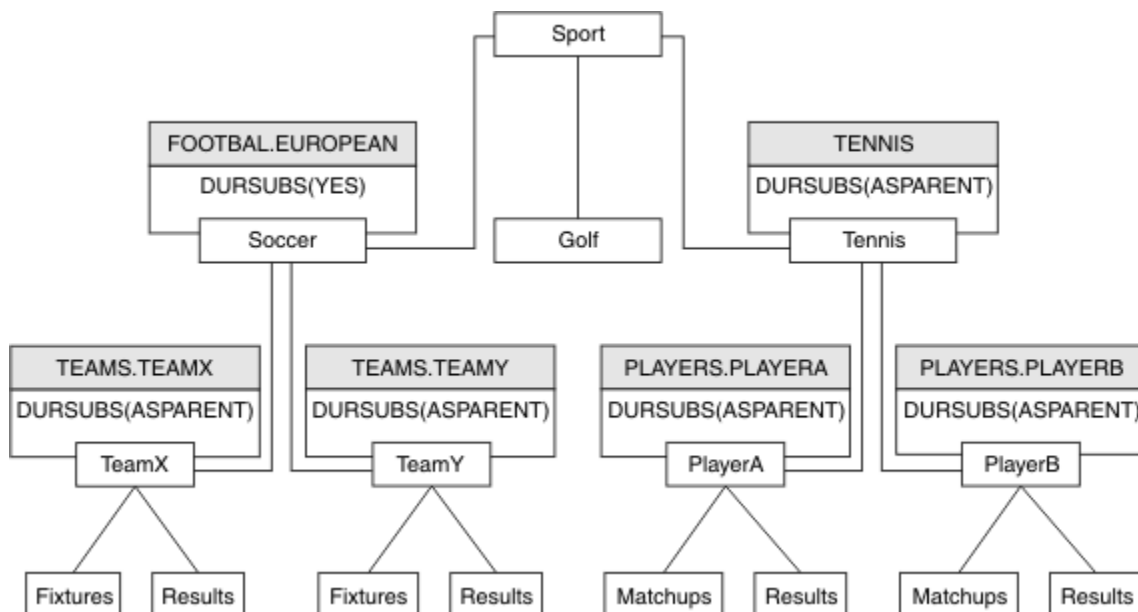
Obrázek 21. Vizualizace objektu administrativního tématu přidruženého k tématu Sport/fotbal

Všechny aplikace, které se přihlašují k odběru témat pod oknem ve stromu, mohou i nadále používat řetězce témat, které používaly před přidáním objektu administrativního tématu. Aplikace však může být nyní zapsána pro odběr pomocí názvu objektu FOOTBALL.EUROPEAN namísto řetězce /Sport/Soccer. Chcete-li se například přihlásit k odběru produktu /Sport/Soccer/Results, může aplikace zadat MQSD.ObjectName jako FOOTBALL.EUROPEAN a MQSD.ObjectString jako Results.

Pomocí této funkce můžete skrýt část stromu témat před vývojáři aplikací. Definujte objekt administrativního tématu v konkrétním uzlu ve stromu témat, pak mohou vývojáři aplikací definovat svá vlastní témata jako podřízené prvky uzlu. Vývojáři musí vědět o nadřazeném tématu, ale ne o žádných jiných uzlech v nadřazeném stromu.

Dědění atributů

Pokud má strom témat mnoho objektů administrativních témat, každý objekt administrativních témat standardně zdědí své atributy od svého nejbližšího nadřazeného administrativního tématu. Předchozí příklad byl rozšířen v souboru [Obrázek 22 na stránce 75](#):



Obrázek 22. Strom témat s několika objekty administrativních témat

Pomocí dědičnosti můžete například všem podřízeným tématům /Sport/Soccer poskytnout vlastnost, že odběry jsou dočasné. Změňte atribut DURSUB FOOTBALL . EUROPEAN na NO.

Tento atribut lze nastavit pomocí následujícího příkazu:

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

Všechny objekty administrativních témat podřízených témat Sport/Soccer mají vlastnost DURSUB nastavenou na výchozí hodnotu ASPARENT. Po změně hodnoty vlastnosti DURSUB FOOTBALL . EUROPEAN na NO podřízená témata Sport/Soccer zdědí hodnotu vlastnosti DURSUB NO. Všechna podřízená témata Sport/Tennis zdědí hodnotu objektu DURSUB from SYSTEM . BASE . TOPIC . SYSTEM . BASE . TOPIC má hodnotu YES.

Pokus o vytvoření trvalého odběru tématu Sport/Soccer/TeamX/Results by nyní selhal, avšak pokus o vytvoření trvalého odběru pro produkt Sport/Tennis/PlayerB/Results by byl úspěšný.

Řízení použití zástupných znaků pomocí vlastnosti ZÁSTUPNÝ znak

Pomocí vlastnosti MQSC Topic ZÁSTUPNÝ znak nebo ekvivalentní vlastnosti PCF Téma WildcardOperation můžete řídit doručování publikací do aplikací odběratele, které používají názvy řetězců témat se zástupnými znaky. Vlastnost ZÁSTUPNÝ znak může mít jednu ze dvou možných hodnot:

WILDCARD

Chování odběrů používajících zástupné znaky s ohledem na toto téma.

PASSTHRU

U odběrů registrovaných pro téma, které používá zástupné znaky a které je méně specifické než řetězec tématu v tomto objektu tématu, jsou poskytnuty publikace k tomuto tématu a k řetězcům tématu, které jsou specifičtější než toto téma.

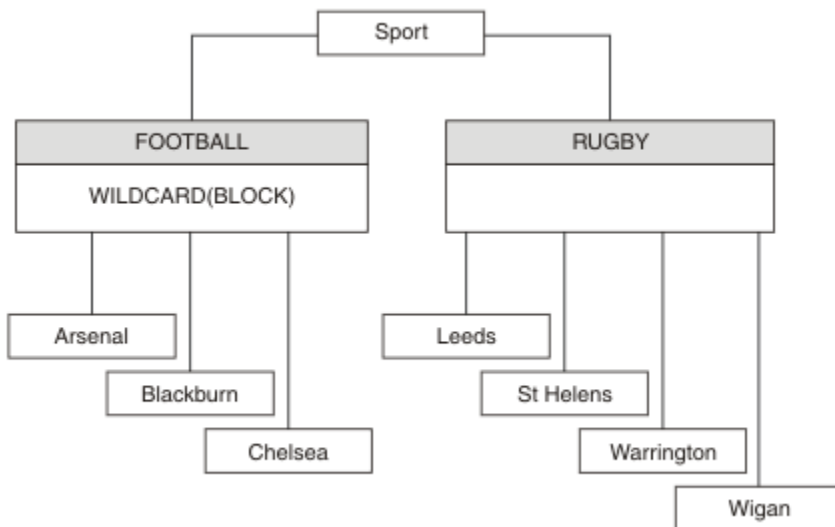
BLOCK

U odběrů registrovaných pro téma, které používá zástupné znaky a které je méně specifické než řetězec tématu v tomto objektu tématu, nejsou poskytnuty publikace k tomuto tématu ani k řetězcům tématu, které jsou specifičtější než toto téma.

Hodnota tohoto atributu se použije při definici odběrů. Když tento atribut změníte, sada témat pokrytých existujícími odběry nebude touto změnou ovlivněna. Tento scénář platí i v případě, že se změnila topologie, tj. když byly objekty tématu vytvořeny nebo odstraněny. Sada témat odpovídajících odběrům, které byly vytvořeny po této změně atributu WILDCARD, se vytvoří s použitím této nové,

upravené topologie. Pokud chcete vynutit opětovné vyhodnocení odpovídající sady témat pro existující odběry, musíte restartovat správce front.

V příkladu “Příklad: Vytvoření klastru publikování/odběru Sport” na stránce 79 můžete postupovat podle kroků a vytvořit stromovou strukturu témat zobrazenou v souboru [Obrázek 23 na stránce 76](#).



Obrázek 23. Strom témat, který používá vlastnost ZÁSTUPNÝ znak BLOCK .

Odběratel používající řetězec tématu se zástupnými znaky # obdrží všechna publikování do tématu Sport a podstromu Sport /Rugby . Odběratel neobdrží žádná publikování do podstromu Sport/Football , protože hodnota vlastnosti ZÁSTUPNÝ znak tématu Sport/Football je BLOCK.

PASSTHRU je výchozí nastavení. Hodnotu vlastnosti ZÁSTUPNÝ znak PASSTHRU můžete nastavit na uzly ve stromu Sport . Pokud uzly nemají hodnotu vlastnosti ZÁSTUPNÝ znak BLOCK, nastavení PASSTHRU nezmění chování pozorované odběrateli pro uzly ve stromu Sports .

V tomto příkladu vytvořte odběry, abyste viděli, jak nastavení zástupných znaků ovlivňuje publikování, která jsou dodána; viz [Obrázek 27 na stránce 81](#). Chcete-li vytvořit některá publikování, spusťte příkaz publish v adresáři [Obrázek 30 na stránce 82](#) .

pub QMA

Obrázek 24. Publikovat do QMA

Výsledky jsou uvedeny v části [Tabulka 3 na stránce 76](#). Všimněte si, jak nastavení hodnoty vlastnosti ZÁSTUPNÝ znak BLOCKzabrání odběru se zástupnými znaky v přijímání publikování do témat v rámci rozsahu zástupného znaku.

Tabulka 3. Publikace přijaté na QMA			
Předplatně	Řetězec tématu	Přijatá publikování	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace na fotbalový podstrom blokován WILDCARD (BLOCK) na Sports/ Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/ Football zabraňuje odběru se zástupnými znaky na Arsenal

Tabulka 3. Publikace přijaté na QMA (pokračování)			
Předplatně	Řetězec tématu	Přijatá publikování	Notes
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Výchozí nastavení WILDCARD na Sports/Rugby nezabrání odběru se zástupnými znaky na Leeds.

Poznámka:

Předpokládejme, že odběr obsahuje zástupný znak, který odpovídá objektu tématu s hodnotou vlastnosti ZÁSTUPNÝ BLOCK. Pokud má odběr také řetězec tématu napravo od odpovídajícího zástupného znaku, odběr nikdy neobdrží publikování. Sada publikací, které nejsou blokovány, jsou publikace pro témata, která jsou nadřazená blokování zástupnému znaku. Publikace k tématům, která jsou podřazenými tématy tématu s hodnotou vlastnosti BLOCK, jsou blokovány zástupným znakem. Proto řetězce tématu odběru, které obsahují téma napravo od zástupného znaku, nikdy neobdrží žádná publikování, která by se shodovala.

Nastavení vlastnosti WILDCARD na hodnotu BLOCK neznamena, že se nemůžete přihlásit k odběru pomocí řetězce tématu, který obsahuje zástupné znaky. Takové předplatné je normální. Odběr má explicitní téma, které odpovídá tématu s objektem tématu, který má hodnotu vlastnosti ZÁSTUPNÝ znak BLOCK. Používá zástupné znaky pro témata, která jsou nadřazenými nebo podřazenými položkami tématu s hodnotou vlastnosti ZÁSTUPNÝ znak BLOCK. V příkladu v souboru [Obrázek 23 na stránce 76](#) může odběr, jako např. Sports/Football/#, přijímat publikování.

Zástupné znaky a témata klastru

Definice témat klastru jsou šířeny do všech správců front v klastru. Odběr tématu klastru v jednom správci front v klastru má za následek vytvoření proxy odběrů správce front. Proxy odběr je vytvořen v každém jiném správci front v klastru. Odběry používající řetězce témat obsahující zástupné znaky v kombinaci s tématy klastru mohou obtížně předpovídat chování. Chování je vysvětleno v následujícím příkladu.

V klastru nastaveném pro tento příklad, "Příklad: Vytvoření klastru publikování/odběru Sport" na stránce 79, má QMB stejnou sadu odběrů jako QMA, ale QMB neobdržel žádná publikování po publikování vydavatele v produktu QMA, viz [Obrázek 24 na stránce 76](#). Ačkoli témata Sports/Football a Sports/Rugby jsou témata klastru, odběry definované v souboru fullsubs.tst neodkazují na téma klastru. Z adresáře QMB do adresáře QMA nejsou šířeny žádné proxy odběry. Bez proxy odběrů nejsou do adresáře QMA předávány žádné publikace do adresáře QMB.

Některé odběry, například Sports/#/Leeds, se mohou jevit jako odkaz na téma klastru, v tomto případě Sports/Rugby. Odběr Sports/#/Leeds se skutečně interpretuje na objekt tématu SYSTEM.BASE.TOPIC.

Pravidlo pro vyřešení objektu tématu, na který odkazuje odběr, například Sports/#/Leeds, je následující. Oříznete řetězec tématu na první zástupný znak. Projděte řetězec tématu a vyhledejte první téma, které má přidružený objekt administrativního tématu. Objekt tématu může určovat název klastru nebo definovat lokální objekt tématu. V příkladu Sports/#/Leeds je řetězec tématu po oříznutí Sports, který nemá žádný objekt tématu, a proto Sports/#/Leeds dědí od SYSTEM.BASE.TOPIC, což je lokální objekt tématu.

Chcete-li zjistit, jak může přihlášení k odběru klastrovaných témat změnit způsob fungování šíření zástupných znaků, spusťte dávkový skript upsubs.bat. Skript vymaže fronty odběrů a přidá odběry témat klastru do souboru fullsubs.tst. Spusťte znovu puba.bat, abyste vytvořili dávku publikací; viz [Obrázek 24 na stránce 76](#).

[Tabulka 4 na stránce 78](#) zobrazuje výsledek přidání dvou nových odběrů do stejného správce front, v němž byla publikování publikována. Výsledek je takový, jak se očekávalo, nové odběry obdrží vždy jedno publikování a počet publikování přijatých ostatními odběry se nezmění. Neočekávané výsledky se vyskytují v jiném správci front klastru; viz [Tabulka 5 na stránce 78](#).

<i>Tabulka 4. Publikace přijaté na QMA</i>			
Předplatně	Řetězec tématu	Přijatá publikování	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace na fotbalový podstrom blokovan WILDCARD (BLOCK) na Sports/ Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/ Football zabraňuje odběru se zástupnými znaky na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Výchozí nastavení WILDCARD na Sports/Rugby nezabrání odběru se zástupnými znaky na Leeds.
FARSENAL	Sports/Football/ Arsenal	Sports/Football/ Arsenal	Produkt Arsenal obdrží publikování, protože odběr neobsahuje zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikaci v každém případě.

Tabulka 5 na stránce 78 zobrazuje výsledky přidání dvou nových odběrů v systému QMB a publikování v systému QMA. Připomeňme, že produkt QMB neobdržel žádná publikování bez těchto dvou nových odběrů. Podle očekávání obdrží oba nové odběry publikování, protože témata Sports/FootBall a Sports/Rugby jsou témata klastru. QMB předal proxy odběry pro Sports/Football/Arsenal a Sports/Rugby/Leeds na QMA, které poté odeslaly publikování na adresu QMB.

Neočekávaným výsledkem je, že dva odběry Sports/# a Sports/#/Leeds, které dříve neobdržely žádná publikování, nyní přijímají publikování. Důvodem je skutečnost, že publikace Sports/Football/ Arsenal a Sports/Rugby/Leeds předané produktu QMB pro ostatní odběry jsou nyní k dispozici pro všechny odběratele připojené k produktu QMB. Následně odběry lokálních témat Sports/# a Sports/#/Leeds obdrží publikaci Sports/Rugby/Leeds. Produkt Sports/#/Arsenal nadále nepřijímá publikování, protože sport/fotbal má hodnotu vlastnosti ZÁSTUPNÝ znak nastavenou na BLOCK.

<i>Tabulka 5. Publikace přijaté na QMB</i>			
Předplatně	Řetězec tématu	Přijatá publikování	Notes
SPORTS	Sports/#	Sports/Rugby/ Leeds	Všechny publikace v podstromu Fotbal jsou blokovány WILDCARD (BLOCK) on Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/ Football zabraňuje odběru se zástupnými znaky na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/ Leeds	Výchozí hodnota WILDCARD na Sports/Rugby nezabrání odběru se zástupnými znaky na Leeds.
FARSENAL	Sports/Football/ Arsenal	Sports/Football/ Arsenal	Produkt Arsenal obdrží publikování, protože odběr neobsahuje zástupný znak.

Tabulka 5. Publikace přijaté na QMB (pokračování)

Předplatně	Řetězec tématu	Přijatá publikování	Notes
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikaci v každém případě.

Ve většině aplikací je nežádoucí, aby jeden odběr ovlivňoval chování jiného odběru. Jedním z důležitých použití vlastnosti ZÁSTUPNÝ znak s hodnotou BLOCK je zajistit, aby se odběry pro stejný řetězec tématu obsahující zástupné znaky chovaly jednotně. Ať už je odběr ve stejném správci front jako vydavatel, nebo v jiném, výsledky odběru jsou stejné.

Zástupné znaky a proudy

Pro novou aplikaci zapsanou do rozhraní API publikování/odběru platí, že odběr produktu * nepřijímá žádná publikování. Chcete-li přijímat všechny sportovní publikace, musíte se přihlásit k odběru publikací Sports/*nebo Sports/#a podobně pro publikace Business .

Chování existující aplikace publikování/odběru ve frontě se nezmění, když je zprostředkovatel publikování/odběru migrován na novější verzi produktu IBM MQ. Vlastnost **StreamName** v příkazech **Publish**, **Register Publisher**nebo **Subscriber** je mapována na název tématu, do kterého byl proud migrován.

Zástupné znaky a body odběru

V případě nové aplikace zapsané do rozhraní API publikování/odběru je výsledkem migrace skutečnost, že odběr produktu * nepřijímá žádná publikování. Chcete-li přijímat všechny sportovní publikace, musíte se přihlásit k odběru publikací Sports/*nebo Sports/#a podobně pro publikace Business .

Chování existující aplikace publikování/odběru ve frontě se nezmění, když je zprostředkovatel publikování/odběru migrován na novější verzi produktu IBM MQ. Vlastnost **SubPoint** v příkazech **Publish**, **Register Publisher**nebo **Subscriber** je mapována na název tématu, na které byl odběr migrován.

Příklad: Vytvoření klastru publikování/odběru Sport

Následující kroky vytvoří klastr CL1se čtyřmi správci front: dvěma úplnými úložišti CL1A a CL1Ba dvěma dílčími úložišti QMA a QMB. Úplná úložiště se používají k uchování pouze definic klastrů. QMA je označen jako hostitel tématu klastru. Trvalé odběry jsou definovány v systémech QMA i QMB.

Poznámka: Příklad je kódován pro Windows. Chcete-li nakonfigurovat a testovat příklad na jiných platformách, musíte znovu kódovat [Vytvořit qmgrs.bat](#) a [vytvořit pub.bat](#) .

1. Vytvořte skriptové soubory.
 - a. [Vytvořit topics.tst](#)
 - b. [Vytvořit wildsubs.tst](#)
 - c. [Vytvořit fullsubs.tst](#)
 - d. [Vytvořit qmgrs.bat](#)
 - e. [vytvořit pub.bat](#)
2. Spuštěním příkazu [Create qmgrs.bat](#) vytvořte konfiguraci.

```
qmgrs
```

Vytvořte témata v adresáři Obrázek 23 na stránce 76. Skript na obrázku 5 vytvoří témata klastru Sports/ Football a Sports/Rugby.

Poznámka: Volba REPLACE nenahrazuje vlastnosti TOPICSTR tématu. TOPICSTR je vlastnost, která se v příkladu používá k testování různých stromů témat. Chcete-li změnit témata, nejprve je odstraňte.

```

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')

```

Obrázek 25. Odstranit a vytvořit témata: topics.tst

Poznámka: Odstraňte témata, protože REPLACE nenahrazuje řetězce témat.

Vytvořte odběry se zástupnými znaky. Zástupné znaky odpovídající tématům s objekty témat v souboru [Obrázek 23](#) na stránce 76. Vytvořte frontu pro každý odběr. Fronty se vymažou a odběry se odstraní při spuštění nebo opětovném spuštění skriptu.

Poznámka: Volba REPLACE nenahrazuje vlastnosti TOPICOBJ nebo TOPICSTR odběru. TOPICOBJ nebo TOPICSTR jsou vlastnosti, které jsou v příkladu užitečné pro testování různých odběrů. Chcete-li je změnit, nejprve odběr odstraňte.

```

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)

```

Obrázek 26. Vytvořit odběry se zástupnými znaky: wildsubs.tst

Vytvořte odběry, které odkazují na objekty témat klastru.

Poznámka:

Oddělovač /je automaticky vložen mezi řetězec tématu, na který odkazuje TOPICOBJ, a řetězec tématu definovaný pomocí TOPICSTR.

Definice DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) vytvoří stejný odběr. TOPICOBJ se používá jako rychlý způsob odkazování na řetězec tématu, který jste již definovali. Vytvořený odběr již neodkazuje na objekt tématu.


```

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)

```

Obrázek 27. Odstranit a vytvořit odběry: *fullsubs.tst*

Vytvořte klastr se dvěma úložišti. Vytvořte dvě dílčí úložiště pro publikování a odběr. Znovu spusťte skript, abyste vše odstranili, a začněte znovu. Skript také vytvoří hierarchii témat a počáteční odběry se zástupnými znaky.

Poznámka:

Na jiných platformách napište podobný skript nebo zadejte všechny příkazy. Použití skriptu umožňuje rychle odstranit vše a začít znovu s identickou konfigurací.

```

@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

```

Obrázek 28. Vytvořit správce front: *qmgrs.bat*

Aktualizujte konfiguraci přidáním odběrů do témat klastru.

```

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

```

Obrázek 29. Odběry aktualizací: *upsubs.bat*

Spuštěním příkazu `pub.bat` parametrem správce front publikujte zprávy obsahující řetězec tématu publikování. Produkt `pub.bat` používá ukázkový program **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
```

Obrázek 30. Publikovat: pub.bat

Proudy a témata

Publikování/odběr ve frontě má koncept proudu publikování, který neexistuje v integrovaném modelu publikování/odběru. V publikování/odběru ve frontě poskytují proudy způsob, jak oddělit tok informací pro různá témata. Proud je implementován jako téma nejvyšší úrovně, které lze administrativně mapovat na jiný identifikátor tématu.

Výchozí proud SYSTEM.BROKER.DEFAULT.STREAM je nastaven automaticky pro všechny zprostředkovatele a správce front v síti a pro použití výchozího proudu není vyžadována žádná další konfigurace. Výchozí proud považujte za nepojmenovaný výchozí prostor tématu. Témata publikovaná do výchozího proudu jsou okamžitě k dispozici všem připojeným správcům front s povoleným publikováním/přihlášením k odběru ve frontě. Pojmenované proudy jsou jako oddělené, pojmenované, prostory tématu. Pojmenovaný proud musí být definován na každém zprostředkovateli, kde se používá.

Pokud se vydavatelé a odběratelé nacházejí v různých správcích front, není po připojení zprostředkovatelů ve stejné hierarchii zprostředkovatelů vyžadována žádná další konfigurace pro publikování a odběry pro jejich vzájemné propojení. Stejná interoperabilita funguje i opačně.

Pojmenované proudy

Návrhář řešení, který pracuje s programovacím modelem publikování/odběru zařazeným ve frontě, se může rozhodnout umístit všechny sportovní publikace do pojmenovaného proudu s názvem Sport. Má-li být proud dostupný pro správce front, který je spuštěn v systému IBM MQ s povoleným publikováním/odběrem ve frontě, musí být proud přidán ručně.

Aplikace publikování/odběru zařazené ve frontě, které se přihlašují k odběru Soccer/Results v proudu Sport, pracují beze změny. Integrované aplikace publikování/odběru, které se přihlásí k odběru tématu Sport pomocí MQSUBa dodáním řetězce tématu Soccer/Results, obdrží také stejná publikování.

Úloha přidání proudu je popsána v tématu [Přidání proudu](#). Možná budete muset přidat proudy ručně ze dvou důvodů.

1. Nadále vyvíjíte aplikace publikování/odběru zařazené ve frontě, které jsou spuštěny ve správcích front novější verze, namísto migrace aplikací do integrovaného rozhraní MQI publikování/odběru.
2. Výchozí mapování proudů na témata vede ke "kolizi" v prostoru tématu a publikování v proudu mají stejný řetězec tématu jako publikování odjinud.

Oprávnění

Ve výchozím nastavení se v kořenovém adresáři stromu témat nachází více objektů tématu: SYSTEM.BASE.TOPIC, SYSTEM.BROKER.DEFAULT.STREAM a SYSTEM.BROKER.DEFAULT.SUBPOINT. Orgány (například pro publikování nebo odběr) jsou určeny orgány na webu SYSTEM.BASE.TOPIC; jakákoli oprávnění v systému SYSTEM.BROKER.DEFAULT.STREAM nebo SYSTEM.BROKER.DEFAULT.SUBPOINT jsou ignorována. Pokud dojde k odstranění objektu SYSTEM.BROKER.DEFAULT.STREAM nebo SYSTEM.BROKER.DEFAULT.SUBPOINT a jeho opětovnému vytvoření s neprázdným řetězcem tématu, budou oprávnění definovaná pro tyto objekty použita stejným způsobem jako běžný objekt tématu.

Mapování mezi proudy a tématy

Proud publikování/odběru zařazený ve frontě se napodobuje v produktu IBM MQ vytvořením fronty a jejím uvedením stejného názvu jako proud. Někdy se fronta nazývá fronta proudu, protože tak se zdá, že

zařazuje do fronty aplikace publikování/odběru. Fronta je identifikována pro stroj publikování/odběru jejím přidáním do speciálního seznamu názvů s názvem SYSTEM.QPUBSUB.QQUEUE.NAMELIST. Přidáním dalších speciálních front do seznamu názvů můžete přidat libovolný počet proudů. Nakonec musíte přidat témata se stejnými názvy jako proudy a se stejnými řetězci témat jako název proudu, abyste mohli publikovat a přihlásit se k odběru témat.

Za výjimečných okolností však můžete dát tématům, která odpovídají proudům, libovolné řetězce témat, které vyberete při definování témat. Účelem řetězce tématu je poskytnout tématu jedinečný název v prostoru tématu. Obvykle název proudu slouží tomuto účelu dokonale. Někdy dochází ke kolizi názvu proudu a existujícího názvu tématu. Chcete-li problém vyřešit, zvolte jiný řetězec tématu pro téma přidružené k proudu. Vyberte libovolný řetězec tématu a ujistěte se, že je jedinečný.

Řetězec tématu definovaný v definici tématu má normální předponu odpovídající řetězci tématu poskytovanému vydavateli a odběrateli pomocí volání rozhraní MQI MQOPEN nebo MQSUB. Aplikace odkazující na témata používající objekty témat nejsou ovlivněny volbou řetězce předpony tématu-proto můžete vybrat libovolný řetězec tématu, který zajistí jedinečnost publikování v prostoru tématu.

Přemapování různých proudů na různá témata závisí na tom, že předpony použité pro řetězce témat jsou jedinečné, aby se jedna sada témat zcela oddělila od druhé. Musíte definovat univerzální konvenci pojmenování témat, která je pevně dodržována, aby mapování fungovalo.

V produktu IBM MQ můžete pomocí mechanismu předpony přemapovat řetězec tématu na jiné místo v prostoru tématu.

Poznámka: Při odstraňování proudu nejprve odstraňte všechny odběry v proudu. Tato akce je nejdůležitější, pokud některý z odběrů pochází od jiných zprostředkovatelů v hierarchii zprostředkovatele.

Body odběru a témata

Pojmenované body odběru jsou emulovány pomocí témat a objektů témat.

Chcete-li přidat body odběru ručně, přečtěte si téma [Přidání bodu odběru](#).

Body odběru v adresáři IBM MQ

Produkt IBM MQ mapuje body odběru na různé prostory tématu ve stromu témat IBM MQ. Témata ve zprávách příkazů bez bodu odběru jsou beze změny mapována na kořen stromu témat IBM MQ a dědí vlastnosti z SYSTEM.BASE.TOPIC.

Zprávy příkazů s bodem odběru se zpracovávají pomocí seznamu objektů tématu v adresáři SYSTEM.QPUBSUB.SUBPOINT.NAMELIST. Název bodu odběru ve zprávě příkazu je porovnán s řetězcem tématu pro každý z objektů tématu v seznamu. Je-li nalezena shoda, bude před název bodu odběru jako uzel tématu uveden řetězec tématu. Téma dědí své vlastnosti z přidruženého objektu tématu, který se nachází v adresáři SYSTEM.QPUBSUB.SUBPOINT.NAMELIST.

Výsledkem použití bodů odběru je vytvoření samostatného prostoru tématu pro každý bod odběru. Prostor tématu má kořen v tématu, které má stejný název jako bod odběru. Témata v jednotlivých prostorech témat dědí své vlastnosti z objektu tématu se stejným názvem jako bod odběru.

Všechny vlastnosti, které nejsou nastaveny v odpovídajícím objektu tématu, jsou normálně zděděny z SYSTEM.BASE.TOPIC.

Existující aplikace publikování/odběru zařazené do fronty s použitím záhlaví zpráv MQRFH2 pokračují v práci nastavením vlastnosti **SubPoint** ve zprávách příkazu Publish nebo Register subscriber. Bod odběru je kombinován s řetězcem tématu ve zprávě příkazu a výsledné téma je zpracováno stejně jako jiné.

Aplikace IBM MQ nejsou body odběru ovlivněny. Pokud aplikace používá téma, které dědí informace z jednoho z odpovídajících objektů tématu, bude tato aplikace spolupracovat s aplikací ve frontě s použitím odpovídajícího bodu odběru.

Příklad

Existující WebSphere Message Broker (nyní označovaný jako IBM Integration Bus) aplikace publikování/odběru, která byla migrována do produktu IBM MQ , vytvořila dva objekty tématu GBP a USDs odpovídajícími řetězci tématu ' GBP ' a ' USD ' .

Existující vydavatelé tématu NYSE/IBM/SPOT, kteří jsou migrovali pro spuštění v systému IBM MQ, kteří používají bod odběru USD k vytvoření publikování v tématu USD/NYSE/IBM/SPOT. Podobně existující odběratelé produktu NYSE/IBM/SPOT pomocí bodu odběru USD vytvořit odběry produktu USD/NYSE/IBM/SPOT.

Přihlaste se k odběru bodové ceny dolaru v programu IBM MQ publikování/odběr voláním funkce MQSUB. Vytvořte odběr pomocí objektu tématu USD a řetězce tématu ' NYSE/IBM/SPOT ' , jak je znázorněno ve fragmentu kódu ' 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. Nastavte atribut CLUSTER objektů tématu USD a GBP na hostiteli tématu klastru.
2. Odstraňte všechny kopie objektů tématu USD a GBP v jiných správčích front v klastru.
3. Ujistěte se, že USD a GBP jsou definovány v SYSTEM.QPUBSUB.SUBPOINT.NAMELIST na každém správci front v klastru.

Příklad konfigurace publikování/odběru jednoho správce front

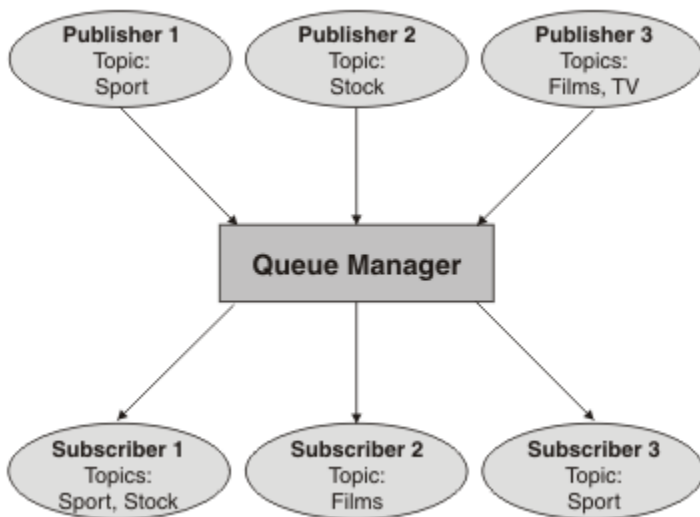
[Obrázek 31](#) na [stránce 85](#) ilustruje základní konfiguraci jednoho správce front publikování/odběru. Příklad zobrazuje konfiguraci pro službu novinek, kde jsou od vydavatelů k dispozici informace o několika tématech:

- Vydavatel 1 publikuje informace o sportovních výsledcích pomocí tématu Sport
- Vydavatel 2 zveřejňuje informace o cenách akcií pomocí tématu Stock
- Vydavatel 3 zveřejňuje informace o filmových recenzích na téma Filmy a o televizních výpisech na téma TV

Tři odběratelé zaregistrovali zájem o různá témata, takže jim správce front odešle informace, o které mají zájem:

- Účastník 1 obdrží sportovní výsledky a ceny akcií
- Předplatitel 2 obdrží filmové recenze
- Účastník 3 obdrží sportovní výsledky

Žádný z účastníků zaregistrovali zájem o televizní výpisy, takže tyto nejsou distribuovány.



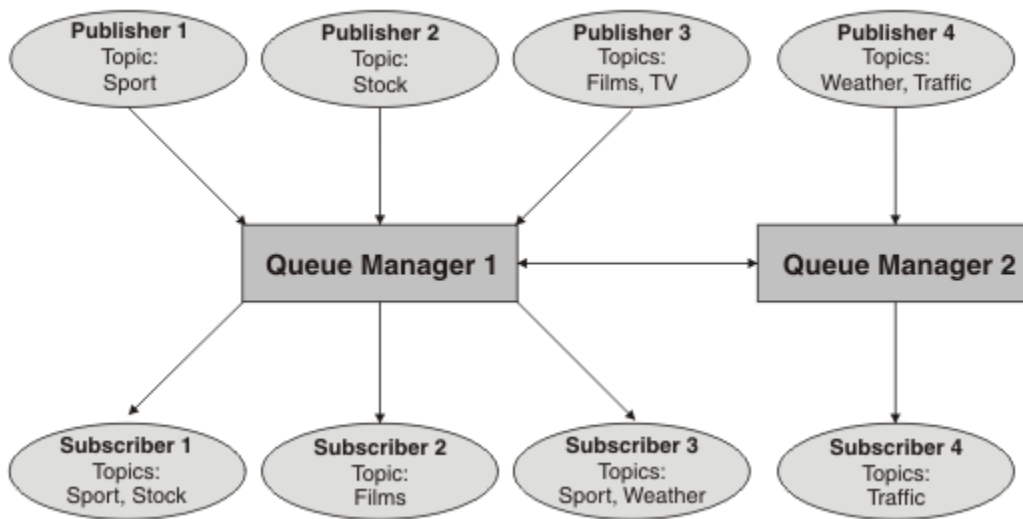
Obrázek 31. Příklad publikování/odběru jednoho správce front

Distribuované sítě publikování/odběru

Každý správce front porovnává zprávy publikované v tématu s lokálně vytvořenými odběry, které se k danému tématu přihlásili. Můžete konfigurovat síť správců front tak, aby zprávy publikované aplikací připojenou k jednomu správci front byly doručovány odpovídajícím odběrům vytvořeným v jiných správcích front v síti. To vyžaduje další konfiguraci prostřednictvím jednoduchých kanálů mezi správcí front.

Distribuovaná konfigurace publikování/odběru je sada vzájemně propojených správců front. Všichni správci front mohou být na stejném fyzickém systému nebo mohou být distribuováni na více fyzických systémech. Při společném připojení správců front se mohou odběratelé přihlásit k odběru jednoho správce front a přijímat zprávy, které byly původně publikovány do jiného správce front. Pro ilustraci je na následujícím obrázku přidán druhý správce front do konfigurace popsané v části [“Příklad konfigurace publikování/odběru jednoho správce front”](#) na stránce 84.

- Správce front 2 používá vydavatel 4 k publikování informací o předpovědích počasí s použitím tématu Počasí a informací o dopravních podmínkách na hlavních silnicích s použitím tématu Provoz.
- Odběratel 4 také používá tohoto správce front a odebírá informace o podmínkách provozu pomocí tématu Přenos.
- Odběratel 3 se také přihlásí k odběru informací o povětrnostních podmínkách, i když používá jiného správce front než vydavatel. To je možné, protože správci front jsou vzájemně propojeni.

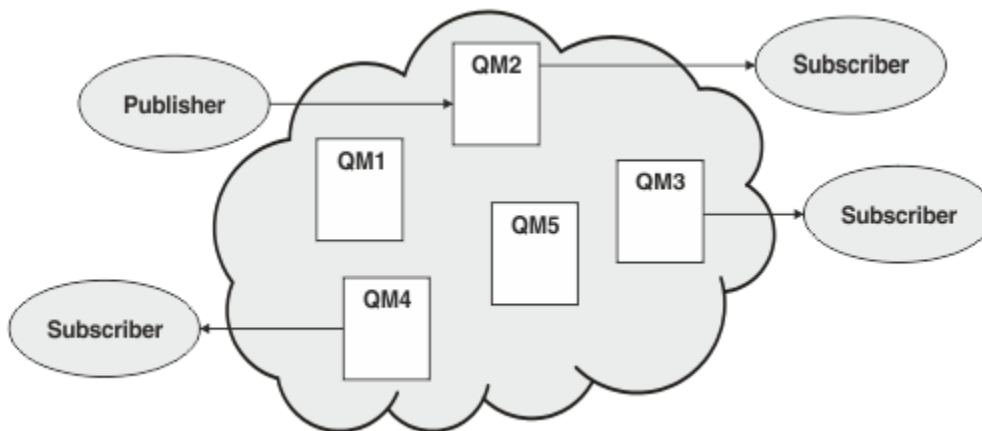


Obrázek 32. Příklad publikování/odběru se dvěma správci front

Můžete ručně připojit správce front v hierarchii nadřízených a podřízených objektů nebo můžete vytvořit klastr publikování/odběru a nechat produkt IBM MQ definovat velkou část podrobností o připojení. Můžete také použít obě topologie v kombinaci, například spojením několika klastrů v hierarchii.

Přehled klastrů publikování/odběru

Klastr publikování/odběru je standardní klastr s jedním nebo více objekty tématu přidanými do klastru. Definujete-li objekt administrativního tématu v libovolném správci front v klastru a určíte-li tento objekt tématu jako klastrovaný zadáním názvu klastru, mohou se vydavatelé a odběratelé tématu připojit k libovolnému správci front v klastru a publikované zprávy jsou směrovány k odběratelům prostřednictvím kanálů klastru mezi správci front.



Obrázek 33. klastr systému publikování/odběr

Existují dva způsoby konfigurace směrování zpráv publikování/odběru v klastru:

- přímé směrování
- směrování hostitele témat

Při konfiguraci přímo směrovaného klastrovaného tématu jsou zprávy publikované v jednom správci front odesílány přímo z tohoto správce front do každého odběru v jiném správci front v klastru. To může poskytnout nejpřímější cestu pro publikování, ale povede k tomu, že všichni správci front v klastru budou informováni o všech ostatních správcích front, přičemž každý z nich může mít mezi sebou zřízeny kanály klastru.

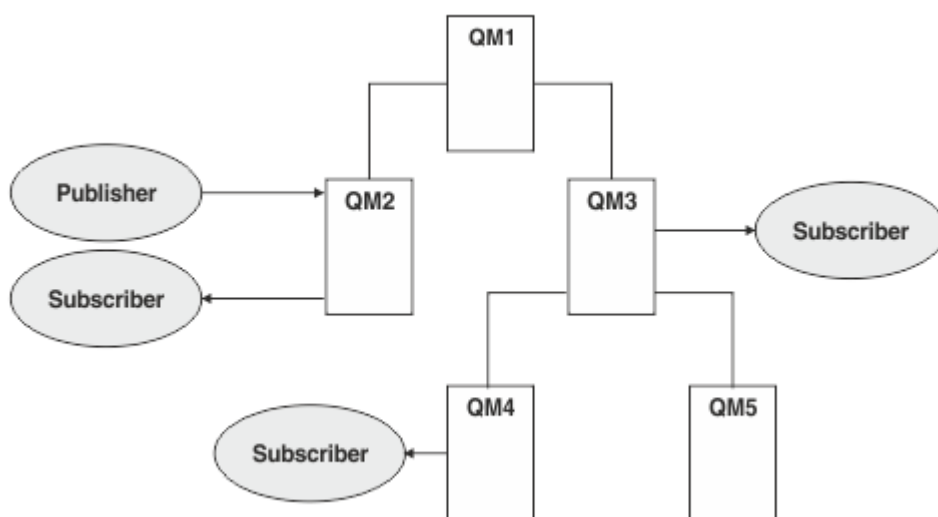
Při použití směrování hostitelů témat jsou zprávy publikované v jednom správci front odesílány z tohoto místa do správce front, který je hostitelem definice spravovaného objektu tématu. Tento *správce front hostitele tématu* směruje zprávu do každého odběru v každém dalším správci front v klastru. Nejsou-li vydavatelé nebo odběratelé umístěni ve správcih front hostitele tématu, bude výsledkem delší trasa pro publikování. Výhodou však je, že se o všech ostatních správcih front v klastru dozvědí pouze správci front hostitele tématu a že s nimi mohou být zřízeny kanály klastru.

Další informace viz [“Klastry publikování/odběru”](#) na stránce 88.

Přehled hierarchií publikování/odběru

Hierarchie publikování/odběru je sada správců front propojených kanály do hierarchické struktury. Každý správce front identifikuje svého *nadřízeného* správce front, jak je popsáno v tématu [Připojení správce front k hierarchii publikování/odběru](#).

Vydavatelé a odběratelé tématu se mohou připojit k libovolnému správci front v hierarchii a zprávy se mezi nimi pohybují pomocí hierarchické konektivity správce front.



Obrázek 34. Hierarchie publikování/odběru

Na předchozím obrázku byly publikace doručené odběratelům v systémech QM3 a QM4 směrovány z QM2 do QM1 a poté do QM3 a nakonec do QM4.

Hierarchie vám poskytují přímou kontrolu nad vztahy mezi jednotlivými správci front v hierarchii. To umožňuje jemnou kontrolu nad směrováním zpráv od vydavatelů k odběratelům a je užitečné zejména při směrování mezi sítěmi správců front s omezenou konektivitou. Měli byste pečlivě zvážit dostupnost a schopnosti každého správce front, jehož prostřednictvím je zpráva směrována na cestě od vydavatele k odběratelům.

Další informace viz [“Hierarchie publikování/odběru”](#) na stránce 90.

Distribuce publikování mezi správci front

Kromě voleb směrování existují dva přístupy k distribuci publikací v síti správců front:

- Publikování z jednoho správce front odesíláte pouze správcům front, kteří jsou aktuálně hostiteli odběru pro dané publikování.
- Odešlete každé publikování všem správcům front a nechte je porovnat s jejich odběry.

První z nich vede k tomu, že se zprávy publikování odesílají pouze v případě potřeby, ale vyžaduje sdílení úrovně znalostí odběru mezi správci front. Druhý z nich nevyžaduje sdílení znalostí odběru, ale může vést k odesílání nepotřebných zpráv publikování mezi správci front.

Ve výchozím nastavení produkt IBM MQ používá dřívější metodu, v níž jsou publikování odesílána pouze správcům front, kteří pro ně mají odběry. Znalosti odběru jsou šířeny mezi správci front ve formě *proxy odběrů*. Závisí na distribuci a životnosti odběrů a na frekvenci publikování, která je neefektivnější pro použití v distribuované topologii publikování/odběru. Viz [Výkon odběru v sítích publikování/odběru](#).

Související pojmy

[“Stromy témat” na stránce 72](#)

Každé téma, které definujete, je prvkem, neboli uzlem, stromu témat. Strom témat může být prázdný, aby mohl začínat nebo obsahovat témata, která byla dříve definována pomocí příkazů MQSC nebo PCF. Nové téma můžete definovat buď pomocí příkazů pro vytvoření tématu, nebo zadáním tématu poprvé v publikování či odběru.

[Scénáře hierarchie publikování/odběru](#)

Související úlohy

[Návrh klastrů publikování/odběru](#)

Klastry publikování/odběru

Klaster publikování/odběru je standardní klaster propojených správců front, v němž jsou publikace automaticky přesouvány z publikačních aplikací do odběrů, které existují ve všech správcích front v klastru. Pro směrování publikací ve klastru publikování/odběru jsou k dispozici dvě volby: *přímé směrování* a *směrování hostitelů témat*. Směrování, které zvolíte, závisí na velikosti a očekávaných vzorcích aktivity pro váš klaster.

Klaster používaný pro systém zpráv publikování/odběru se neliší od standardního klastru IBM MQ. V takovém případě mohou správci front v rámci klastru publikování/odběru existovat ve fyzicky samostatných počítačích a každá dvojice správců front je v případě potřeby automaticky propojena kanály klastru. Další informace viz [Klastry](#).

Při konfiguraci standardního klastru správců front pro účely publikování/odběru zpráv definujete nejméně jeden spravovaný objekt tématu ve správci front v klastru. Chcete-li nastavit téma jako téma klastru, konfiguruje vlastnost **CLUSTER** s názvem klastru. Pokud tak učiníte, jakékoli téma používané vydavatelem nebo odběratelem v daném bodě nebo níže v stromu témat bude sdíleno mezi všemi správci front v klastru a zprávy publikované v klastrované větvi stromu témat budou automaticky směrovány na odběry v jiných správcích front v klastru.

Mezi správcem front vydavatele a všemi ostatními správci front je odeslána pouze jedna kopie každé zprávy bez ohledu na počet odběratelů pro zprávu v cílovém správci front. Při příchodu do správce front s jedním nebo více odběry je zpráva duplikována ve všech odběrech.

Každý správce front, který se připojí ke klastru, se automaticky dozví o klastrovaných tématech a vydavatelé a odběratelé v tomto správci front se automaticky účastní klastru.

Neklastrovaná aktivita publikování/odběru může také probíhat v klastru publikování/odběru, a to tak, že bude pracovat s řetězcí témat, které nespádají do klastrovaného objektu tématu.

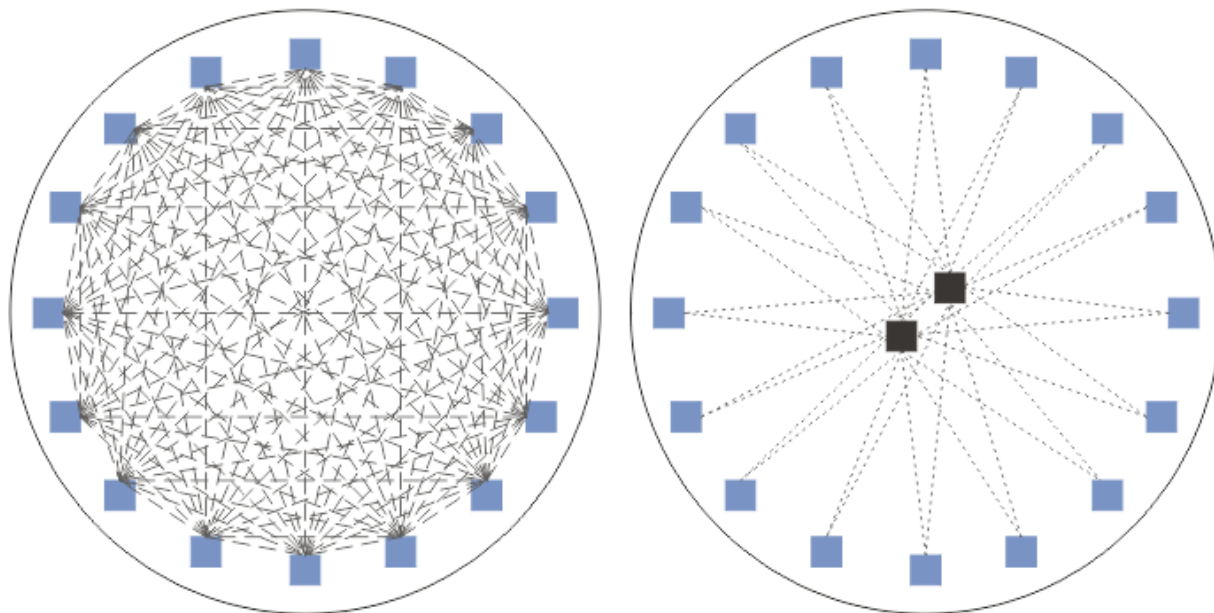
Pro směrování publikací ve klastru publikování/odběru jsou k dispozici dvě volby: *přímé směrování* a *směrování hostitelů témat*. Chcete-li zvolit směrování zpráv, které má být použito uvnitř klastru, nastavte vlastnost **CLROUTE** na administrovaném objektu tématu na jednu z následujících hodnot:

- **DIRECT**
- **TOPICHOST**

Standardně je směrování tématu **DIRECT**. Před produktem IBM MQ 8.0 bylo k dispozici pouze toto směrování. Pokud nakonfigurujete přímo směrované klastrované téma ve správci front, všichni správci front ve klastru budou mít informace o všech ostatních správcích front ve klastru. Při provádění operací publikování a odběru se každý správce fronty může připojit přímo k jinému správci fronty v klastru.

Od IBM MQ 8.0 můžete místo toho konfigurovat směrování témat jako **TOPICHOST**. Při použití směrování hostitelů témat budou mít všichni správci front v klastru informace o správcích front klastru, kteří jsou hostiteli směrované definice tématu (tj. správcích front, na kterých jste definovali objekt tématu). Při provádění operací publikování a odběru se správci front v klastru připojí pouze ke správcům front hostitele

tématu a nikoli přímo každý s každým. Správci front hostitele tématu odpovídají za směrování publikování ze správců front, na nichž dochází k publikování publikací, na správce front s odpovídajícími odběry.



Obrázek 35. Přímé směrování a směrování hostitelů témat

Přehled přímého směrování

Je-li administrovaný objekt tématu konfigurován pro přímé směrování, je třeba objekt tématu definovat pouze v jednom ze správců front v klastru, aby se o něm všichni správci front dozvěděli. Volba správce front, pro kterého je téma definováno, nemá vliv na chování systému zpráv publikování/odběru pro dané téma.

Každá zpráva směřuje přímo ze správce front vydavatele ke každému odběru v ostatních správcích front v klastru, nikoli k průchodu prostředními správci front.

Standardně jsou zprávy odesílány pouze jiným správcům front v klastru, kteří jsou hostiteli jednoho nebo více odběrů.

- To závisí na tom, aby každý správce front přímo informoval všechny ostatní správce front v klastru o všech tématech, pro která je aktuálně k dispozici jeden či více odběrů. To má za následek, že všichni správci front v klastru budou mít přehled o všech tématech, která jsou přihlášena k odběru, a o všech správcích front, kteří jsou hostiteli odběru, který vytváří kanál pro všechny ostatní správce front. Tato volba je nezávislá na tom, zda má každý správce front vydavatele.
- Znalost jednotlivých odebíraných témat ve všech správcích front lze odebrat změnou na model odesílání všech publikování všem správcům front v klastru bez ohledu na to, zda mají odběry. Tím se snižuje přenos znalostí odběru, ale je pravděpodobné, že se zvýší provoz publikování a počet kanálů, které každý správce front zavede. Viz [Výkon odběru v sítích publikování/odběru](#).

Toky zpráv publikování/odběru používající přímo směrovaná klastrovaná témata mohou zahrnovat více klastrů publikování/odběru přidáním jednoho správce front z každého klastru do hierarchie publikování/odběru. Viz [Kombinace prostorů témat více klastrů](#).

Podrobnější prozkoumání přímého směrování naleznete v tématu [Přímé směrování v klastrech publikování/odběru](#).

Přehled směrování hostitelů témat

Je-li administrovaný objekt tématu konfigurován pro směrování hostitele tématu, jsou publikováni ze správce front v klastru směrována prostřednictvím správce front, v němž je objekt tématu konfigurován ("hostitel tématu"), a odtud do správců front, v nichž existují odběry.

- To závisí na tom, aby každý správce front informoval všechny hostitele témat o každém tématu, pro které je aktuálně k dispozici jeden nebo více odběrů. Každý správce front, který je hostitelem odběru, vytvoří kanál pro každého hostitele tématu, k němuž se odběr vztahuje.
- Správci front, kteří nejsou hostiteli témat, nejsou upozorněni na jiné správce front, kteří nejsou hostiteli témat, v klastru pro účely publikování/odběru, a kanály mezi nimi nejsou pro tento účel zřízeny.
- Je-li publikační aplikace připojena ke správci front, který je hostitelem daného tématu, budou publikované zprávy směrovány přímo ke správcům front, pro které byly vytvořeny odpovídající odběry, aniž by byl vyžadován další přechod. Podobně, pokud jsou odpovídající odběry vytvořeny v jediném správci front, který je hostitelem tématu, jsou zprávy publikované do tohoto tématu směrovány přímo do tohoto správce front, aniž by byl vyžadován další přechod.
- Odběry ve stejném správci front jako vydavatel jsou splněny bez předchozího směrování publikací na hostitele objektu tématu.

Stejně jako v případě klastrovaných front může více správců front konfigurovat stejný objekt administrativního tématu. To poskytuje vyšší dostupnost směrování zpráv a horizontální škálování prostřednictvím vyrovnávání pracovní zátěže. V případě objektů témat směrovaných hostitelem témat platí, že pokud více správců front konfiguruje stejné pojmenované téma pro stejnou větev stromu témat, každý hostitel témat bude informován o odebíraných tématech každým správcem front, který je hostitelem odběru.

- Když je zpráva publikována, je odeslána jednomu ze správců front hostitele tématu, aby ji předal správcům front, kteří jsou hostiteli odběru. Volba správce front hostitele tématu se řídí stejnými výchozími pravidly pro vyrovnávání pracovní zátěže jako pro klastrované fronty dvoubodového spojení.
- Pokud správce front publikování nemůže kontaktovat jednoho nebo více správců front hostitele tématu, budou zprávy směrovány na zbývající dostupné téma, které je hostitelem správců front.

Každé publikování tématu ve směrované větvi stromu témat je předáno jednomu z hostitelů témat, a to i v případě, že neexistují žádné odběry tohoto tématu kdekoli v klastru. Při výchozím nastavení jsou zprávy odesílány z tohoto místa pouze jiným správcům front v klastru, kteří jsou hostiteli jednoho či více odběrů.

- To závisí na tom, aby byl každý správce front hostitele tématu informován o všech odebíraných řetězcích témat v každém správci front v klastru.
- Znalost jednotlivých odebíraných témat lze odebrat změnou na model odesílání všech publikací směrovaných na hostitele tématu všem správcům front v klastru bez ohledu na to, zda mají odběry. Tím se snižuje provoz na základě informací o odběrech, ale je pravděpodobné, že dojde ke zvýšení provozu publikování a potenciálně i počtu kanálů zavedených s jednotlivými tématy, která jsou hostitelem správce front. Viz [Výkon odběru v sítích publikování/odběru](#).

Toky zpráv publikování/odběru používající klastrovaná témata směrovaná hostitelem tématu **nemohou** zasahovat do více klastrů publikování/odběru prostřednictvím použití hierarchie publikování/odběru.

Podrobnější informace o směrování hostitelů témat naleznete v tématu [Směrování hostitelů témat v klastrech publikování/odběru](#).

Hierarchie publikování/odběru

Hierarchii publikování/odběru sestavíte propojením správců front pomocí kanálů a následným definováním vztahu podřízeného a nadřízeného prvku mezi dvojicí správců front. Zpráva směřuje od vydavatele k odběrům prostřednictvím přímých vztahů v hierarchii. Všimněte si, že to může znamenat více "přechodů", abyste se tam dostali.

Mezi libovolnou dvojicí správců front je odeslána pouze jedna kopie zprávy bez ohledu na počet odběratelů pro zprávu v cílovém správci front. Při příchodu do správce front s jedním nebo více odběry je zpráva duplikována ve všech odběrech.

Standardně jsou zprávy odesílány pouze jiným správcům front v hierarchii, kteří jsou na trase k odběru v jiném správci front:

- To závisí na každém správci front, který informuje každý přímý vztah o všech tématech, která k němu aktuálně mají jeden nebo více odběrů, a to buď na tomto správci front, nebo na jednom z jeho dalších vztahů. To má za následek, že všichni správci front v hierarchii budou informováni o všech tématech, která jsou přihlášena k odběru.
- Toto chování lze změnit tak, aby vždy odesílala publikování všem správcům front v hierarchii bez ohledu na existující odběry. To odstraňuje potřebu šíření informací o odběru v rámci hierarchie, ale může zvýšit provoz publikování.

Při vytváření klastru je třeba dbát na to, abyste nevytvořili smyčku, která způsobí, že zprávy budou v síti cyklovat navždy. V hierarchii nelze vytvořit žádné takové smyčky.

Každý správce front musí mít jedinečný název správce front.

Toky zpráv publikování/odběru mohou zahrnovat více klastrů publikování/odběru. Chcete-li tak učinit, přidejte jednoho správce front z každého klastru do hierarchie publikování/odběru.

Podrobnější prozkoumání naleznete v tématu [Směrování v hierarchiích publikování/odběru](#).

Proxy odběry v síti publikování/odběru

Proxy odběr je odběr vytvořený jedním správcem front pro témata publikován na jiném správci front. Proxy odběr probíhá mezi správci front pro každý jednotlivý řetězec tématu, k jehož odběru je přihlášený nějaký odběr. Proxy odběry nevytváříte explicitně, správce front je vytváří vaším jménem.

Správce front můžete spojit dohromady do klastru publikování/odběru nebo do hierarchie publikování/odběru. Tok proxy odběrů mezi připojenými správci front. Proxy odběry způsobují, že publikování v tématu vytvořeném vydavatelem připojeným k jednomu správci front jsou přijímána odběrateli v daném tématu připojeném k jiným správcům front. Viz [“Distribuované síť publikování/odběru”](#) na stránce 85.

V topologiích publikování/odběru s mnoha tisíci odběrů pro jednotlivé řetězce témat nebo v případech, kdy se existence těchto odběrů může rychle měnit, je třeba vzít v úvahu režii šíření proxy odběrů. Kromě automatické agregace popsané ve zbytku tohoto tématu můžete provádět ruční změny konfigurace, které dále omezují tok proxy odběrů a publikování mezi připojenými správci front a snižují latenci čekání na předání proxy odběru všem připojeným správcům front. Viz [Výkon odběru v sítích publikování/odběru](#).

Proxy odběry neobsahují žádné selektory používané lokálními odběry a řetězce témat odběrů, které obsahují zástupné znaky, mohou být zjednodušeny. To může vést k publikování, která se shodují s proxy odběry, kde skutečné odběry nejsou, což může vést k dalšímu toku publikování mezi správci front. Správce front, který je hostitelem odběrů, tyto nesrovnalosti odfiltruje, aby se do odběrů nevracela další publikování.

Agregace odběrů proxy

Proxy odběry jsou agregovány pomocí duplicitního systému eliminace. Pro konkrétní vyřešený řetězec tématu je při prvním lokálním odběru nebo při přijatém proxy odběru odeslán proxy odběr. Následné odběry téhož řetězce tématu využívají tento existující proxy odběr.

Proxy odběr je zrušen po zrušení posledního lokálního odběru nebo přijatého proxy odběru.

Agregace publikování

Pokud ve správci front existuje více než jeden odběr stejného řetězce tématu, bude z jiných správců front v topologii publikování/odběru odeslána pouze jedna kopie každého publikování, které odpovídá danému řetězci tématu. Při přijetí zprávy lokální správce front doručí kopii zprávy každému odpovídajícímu odběru.

Je možné, aby více než jeden proxy odběr odpovídal řetězci tématu jednoho publikování, pokud proxy odběry obsahují zástupné znaky. Je-li zpráva publikována ve správci front, který odpovídá dvěma nebo více proxy odběrům vytvořeným jedním připojeným správcem front, bude do vzdáleného správce front předána pouze jedna kopie publikování, aby bylo možné uspokojit více proxy odběrů.

Související pojmy

[Detekce smyček v distribuované síti publikování/odběru](#)

Zástupné znaky v proxy odběrech

Odběry mohou v řetězcích témat používat zástupné znaky pro porovnání s více řetězci témat v publikacích.

Existují dvě schémata se zástupnými znaky, která může odběr použít: *na základě tématu* a *na základě znaků*. Viz [“Schémata zástupných znaků”](#) na stránce 66.

V produktu IBM MQ jsou všechny proxy odběry pro zástupné znaky převedeny na použití zástupných znaků založených na tématu. Je-li nalezen zástupný znak založený na znacích, je nahrazen znakem # , zpět na nejbližší / . Například /aaa/bbb/c*d se převede na /aaa/bbb/#. Výsledkem převodu je, že vzdálení správci front odesílají o něco více publikování, než bylo explicitně přihlášeno k odběru. Další publikování jsou odfiltrována lokálním správcem front při jejich doručení lokálními odběrateli.

Řízení použití zástupných znaků pomocí vlastnosti ZÁSTUPNÝ znak

Pomocí vlastnosti MQSC **Topic ZÁSTUPNÝ** znak nebo ekvivalentní vlastnosti PCF **Téma WildcardOperation** můžete řídit doručování publikací do aplikací odběratele, které používají názvy řetězců témat se zástupnými znaky. Vlastnost ZÁSTUPNÝ znak může mít jednu ze dvou možných hodnot:

WILDCARD

Chování odběrů používajících zástupné znaky s ohledem na toto téma.

PASSTHRU

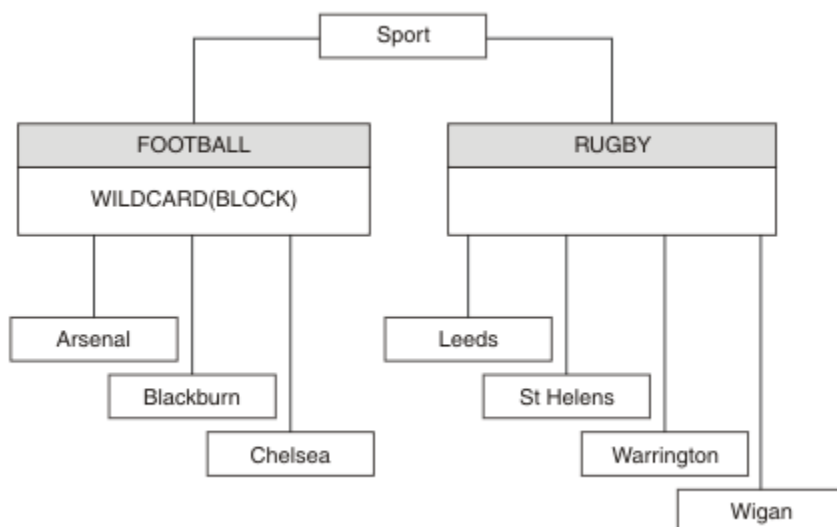
U odběrů registrovaných pro téma, které používá zástupné znaky a které je méně specifické než řetězec tématu v tomto objektu tématu, jsou poskytnuty publikace k tomuto tématu a k řetězcům tématu, které jsou specifičtější než toto téma.

BLOCK

U odběrů registrovaných pro téma, které používá zástupné znaky a které je méně specifické než řetězec tématu v tomto objektu tématu, nejsou poskytnuty publikace k tomuto tématu ani k řetězcům tématu, které jsou specifičtější než toto téma.

Hodnota tohoto atributu se použije při definici odběrů. Když tento atribut změníte, sada témat pokrytých existujícími odběry nebude touto změnou ovlivněna. Tento scénář platí i v případě, že se změnila topologie, tj. když byly objekty tématu vytvořeny nebo odstraněny. Sada témat odpovídajících odběrům, které byly vytvořeny po této změně atributu WILDCARD, se vytvoří s použitím této nové, upravené topologie. Pokud chcete vynutit opětovné vyhodnocení odpovídající sady témat pro existující odběry, musíte restartovat správce front.

V příkladu [“Příklad: Vytvoření klastru publikování/odběru Sport”](#) na stránce 79 můžete postupovat podle kroků a vytvořit stromovou strukturu témat zobrazenou v souboru [Obrázek 23](#) na stránce 76.



Obrázek 36. Strom témat, který používá vlastnost ZÁSTUPNÝ znak BLOCK .

Odběratel používající řetězec tématu se zástupnými znaky # obdrží všechna publikování do tématu Sport a podstromu Sport /Rugby . Odběratel neobdrží žádná publikování do podstromu Sport/Football , protože hodnota vlastnosti ZÁSTUPNÝ znak tématu Sport/Football je BLOCK.

PASSTHRU je výchozí nastavení. Hodnotu vlastnosti ZÁSTUPNÝ znak PASSTHRU můžete nastavit na uzly ve stromu Sport . Pokud uzly nemají hodnotu vlastnosti ZÁSTUPNÝ znak BLOCK, nastavení PASSTHRU nezmění chování pozorované odběrateli pro uzly ve stromu Sports .

V tomto příkladu vytvořte odběry, abyste viděli, jak nastavení zástupných znaků ovlivňuje publikování, která jsou dodána; viz Obrázek 27 na stránce 81. Chcete-li vytvořit některá publikování, spusťte příkaz publish v adresáři Obrázek 30 na stránce 82 .

pub QMA

Obrázek 37. Publikovat do QMA

Výsledky jsou uvedeny v části Tabulka 3 na stránce 76. Všimněte si, jak nastavení hodnoty vlastnosti ZÁSTUPNÝ znak BLOCKzabrání odběru se zástupnými znaky v přijímání publikování do témat v rámci rozsahu zástupného znaku.

Tabulka 6. Publikace přijaté na QMA			
Předplatn é	Řetězec tématu	Přijatá publikování	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace na fotbalový podstrom blokován WILDCARD (BLOCK) na Sports/ Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/ Football zabraňuje odběru se zástupnými znaky na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Výchozí nastavení WILDCARD na Sports/Rugby nezabrání odběru se zástupnými znaky na Leeds.

Poznámka:

Předpokládejme, že odběr obsahuje zástupný znak, který odpovídá objektu tématu s hodnotou vlastnosti ZÁSTUPNÝ BLOCK. Pokud má odběr také řetězec tématu napravo od odpovídajícího zástupného znaku, odběr nikdy neobdrží publikování. Sada publikací, které nejsou blokovány, jsou publikace pro témata, která jsou nadřazená blokovánému zástupnému znaku. Publikace k tématům, která jsou podřizenými tématy tématu s hodnotou vlastnosti BLOCK, jsou blokovány zástupným znakem. Proto řetězce tématu odběru, které obsahují téma napravo od zástupného znaku, nikdy neobdrží žádná publikování, která by se shodovala.

Nastavení vlastnosti WILDCARD na hodnotu BLOCK neznámá, že se nemůžete přihlásit k odběru pomocí řetězce tématu, který obsahuje zástupné znaky. Takové předplatné je normální. Odběr má explicitní téma, které odpovídá tématu s objektem tématu, který má hodnotu vlastnosti ZÁSTUPNÝ znak BLOCK. Používá zástupné znaky pro témata, která jsou nadřizenými nebo podřizenými položkami tématu s hodnotou vlastnosti ZÁSTUPNÝ znak BLOCK. V příkladu v souboru Obrázek 23 na stránce 76 může odběr, jako např. Sports/Football/#, přijímat publikování.

Zástupné znaky a témata klastru

Definice témat klastru jsou šířeny do všech správců front v klastru. Odběr tématu klastru v jednom správci front v klastru má za následek vytvoření proxy odběrů správce front. Proxy odběr je vytvořen v každém jiném správci front v klastru. Odběry používající řetězce témat obsahující zástupné znaky v kombinaci s tématy klastru mohou obtížně předpovídat chování. Chování je vysvětleno v následujícím příkladu.

V klastru nastaveném pro tento příklad, "Příklad: Vytvoření klastru publikování/odběru Sport" na stránce 79, má QMB stejnou sadu odběrů jako QMA, ale QMB neobdržel žádná publikování po publikování vydavatele v produktu QMA, viz Obrázek 24 na stránce 76. Ačkoli témata Sports/Football a Sports/Rugby jsou témata klastru, odběry definované v souboru `fullsubs.tst` neodkazují na téma klastru. Z adresáře QMB do adresáře QMA nejsou šířeny žádné proxy odběry. Bez proxy odběrů nejsou do adresáře QMA předávány žádné publikace do adresáře QMB.

Některé odběry, například Sports/#/Leeds, se mohou jevit jako odkaz na téma klastru, v tomto případě Sports/Rugby. Odběr Sports/#/Leeds se skutečně interpretuje na objekt tématu SYSTEM.BASE.TOPIC.

Pravidlo pro vyřešení objektu tématu, na který odkazuje odběr, například Sports/#/Leeds, je následující. Oříznete řetězec tématu na první zástupný znak. Projděte řetězec tématu a vyhledejte první téma, které má přidružený objekt administrativního tématu. Objekt tématu může určovat název klastru nebo definovat lokální objekt tématu. V příkladu Sports/#/Leeds je řetězec tématu po oříznutí Sports, který nemá žádný objekt tématu, a proto Sports/#/Leeds dědí od SYSTEM.BASE.TOPIC, což je lokální objekt tématu.

Chcete-li zjistit, jak může přihlášení k odběru klastrovaných témat změnit způsob fungování šíření zástupných znaků, spusťte dávkový skript `upsubs.bat`. Skript vymaže fronty odběrů a přidá odběry témat klastru do souboru `fullsubs.tst`. Spusťte znovu `puba.bat`, abyste vytvořili dávku publikací; viz Obrázek 24 na stránce 76.

Tabulka 4 na stránce 78 zobrazuje výsledek přidání dvou nových odběrů do stejného správce front, v němž byla publikování publikována. Výsledek je takový, jak se očekávalo, nové odběry obdrží vždy jedno publikování a počet publikování přijatých ostatními odběry se nezmění. Neočekávané výsledky se vyskytují v jiném správci front klastru; viz Tabulka 5 na stránce 78.

Tabulka 7. Publikace přijaté na QMA			
Předplatné	Řetězec tématu	Přijatá publikování	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace na fotbalový podstrom blokován WILDCARD (BLOCK) na Sports/Football

<i>Tabulka 7. Publikace přijaté na QMA (pokračování)</i>			
Předplatně	Řetězec tématu	Přijatá publikování	Notes
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football zabraňuje odběru se zástupnými znaky na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Výchozí nastavení WILDCARD na Sports/Rugby nezabrání odběru se zástupnými znaky na Leeds.
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Produkt Arsenal obdrží publikování, protože odběr neobsahuje zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikaci v každém případě.

Tabulka 5 na stránce 78 zobrazuje výsledky přidání dvou nových odběrů v systému QMB a publikování v systému QMA. Připomeňme, že produkt QMB neobdržel žádná publikování bez těchto dvou nových odběrů. Podle očekávání obdrží oba nové odběry publikování, protože témata Sports/FootBall a Sports/Rugby jsou témata klastru. QMB předal proxy odběry pro Sports/Football/Arsenal a Sports/Rugby/Leeds na QMA, které poté odeslaly publikování na adresu QMB.

Neočekávaným výsledkem je, že dva odběry Sports/# a Sports/#/Leeds, které dříve neobdržely žádná publikování, nyní přijímají publikování. Důvodem je skutečnost, že publikace Sports/Football/Arsenal a Sports/Rugby/Leeds předané produktu QMB pro ostatní odběry jsou nyní k dispozici pro všechny odběratele připojené k produktu QMB. Následně odběry lokálních témat Sports/# a Sports/#/Leeds obdrží publikaci Sports/Rugby/Leeds. Produkt Sports/#/Arsenal nadále nepřijímá publikování, protože sport/fotbal má hodnotu vlastnosti ZÁSTUPNÝ znak nastavenou na BLOCK.

<i>Tabulka 8. Publikace přijaté na QMB</i>			
Předplatně	Řetězec tématu	Přijatá publikování	Notes
<i>SPORTS</i>	<i>Sports/#</i>	<i>Sports/Rugby/Leeds</i>	<i>Všechny publikace v podstromu Fotbal jsou blokovány WILDCARD (BLOCK) on Sports/Football</i>
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) on Sports/Football zabraňuje odběru se zástupnými znaky na Arsenal
<i>SLEEDS</i>	<i>Sports/#/Leeds</i>	<i>Sports/Rugby/Leeds</i>	<i>Výchozí hodnota WILDCARD na Sports/Rugby nezabrání odběru se zástupnými znaky na Leeds.</i>
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Produkt Arsenal obdrží publikování, protože odběr neobsahuje zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikaci v každém případě.

Ve většině aplikací je nežádoucí, aby jeden odběr ovlivňoval chování jiného odběru. Jedním z důležitých použití vlastnosti ZÁSTUPNÝ znak s hodnotou BLOCK je zajistit, aby se odběry pro stejný řetězec tématu

obsahující zástupné znaky chovaly jednotně. Ať už je odběr ve stejném správci front jako vydavatel, nebo v jiném, výsledky odběru jsou stejné.

Zástupné znaky a proudy

Pro novou aplikaci zapsanou do rozhraní API publikování/odběru platí, že odběr produktu * nepřijímá žádná publikování. Chcete-li přijímat všechny sportovní publikace, musíte se přihlásit k odběru publikací `Sports/*` nebo `Sports/#a` podobně pro publikace `Business`.

Chování existující aplikace publikování/odběru ve frontě se nezmění, když je zprostředkovatel publikování/odběru migrován na novější verzi produktu IBM MQ. Vlastnost **StreamName** v příkazech **Publish**, **Register Publisher** nebo **Subscriber** je mapována na název tématu, do kterého byl proud migrován.

Zástupné znaky a body odběru

V případě nové aplikace zapsané do rozhraní API publikování/odběru je výsledkem migrace skutečnost, že odběr produktu * nepřijímá žádná publikování. Chcete-li přijímat všechny sportovní publikace, musíte se přihlásit k odběru publikací `Sports/*` nebo `Sports/#a` podobně pro publikace `Business`.

Chování existující aplikace publikování/odběru ve frontě se nezmění, když je zprostředkovatel publikování/odběru migrován na novější verzi produktu IBM MQ. Vlastnost **SubPoint** v příkazech **Publish**, **Register Publisher** nebo **Subscriber** je mapována na název tématu, na které byl odběr migrován.

Příklad: Vytvoření klastru publikování/odběru Sport

Následující kroky vytvoří klastr CL1 se čtyřmi správci front: dvěma úplnými úložišti CL1A a CL1Ba dvěma dílčími úložišti QMA a QMB. Úplná úložiště se používají k uchování pouze definic klastrů. QMA je označen jako hostitel tématu klastru. Trvalé odběry jsou definovány v systémech QMA i QMB.

Poznámka: Příklad je kódován pro Windows. Chcete-li nakonfigurovat a testovat příklad na jiných platformách, musíte znovu kódovat [Vytvořit qmgrs.bat](#) a [vytvořit pub.bat](#).

1. Vytvořte skriptové soubory.
 - a. [Vytvořit topics.tst](#)
 - b. [Vytvořit wildsubs.tst](#)
 - c. [Vytvořit fullsubs.tst](#)
 - d. [Vytvořit qmgrs.bat](#)
 - e. [vytvořit pub.bat](#)
2. Spuštěním příkazu [Create qmgrs.bat](#) vytvořte konfiguraci.

```
qmgrs
```

Vytvořte témata v adresáři [Obrázek 23 na stránce 76](#). Skript na [obrázku 5](#) vytvoří témata klastru `Sports/Football` a `Sports/Rugby`.

Poznámka: Volba `REPLACE` nenahrazuje vlastnosti `TOPICSTR` tématu. `TOPICSTR` je vlastnost, která se v příkladu používá k testování různých stromů témat. Chcete-li změnit témata, nejprve je odstraňte.


```

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')

```

Obrázek 38. Odstranit a vytvořit témata: *topics.tst*

Poznámka: Odstraňte témata, protože REPLACE nenahrazuje řetězce témat.

Vytvořte odběry se zástupnými znaky. Zástupné znaky odpovídající tématům s objekty témat v souboru [Obrázek 23](#) na stránce 76. Vytvořte frontu pro každý odběr. Fronty se vymažou a odběry se odstraní při spuštění nebo opětovném spuštění skriptu.

Poznámka: Volba REPLACE nenahrazuje vlastnosti TOPICOBJ nebo TOPICSTR odběru. TOPICOBJ nebo TOPICSTR jsou vlastnosti, které jsou v příkladu užitečné pro testování různých odběrů. Chcete-li je změnit, nejprve odběr odstraňte.

```

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)

```

Obrázek 39. Vytvořit odběry se zástupnými znaky: *wildsubs.tst*

Vytvořte odběry, které odkazují na objekty témat klastru.

Poznámka:

Oddělovač /je automaticky vložen mezi řetězec tématu, na který odkazuje TOPICOBJ, a řetězec tématu definovaný pomocí TOPICSTR.

Definice DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) vytvoří stejný odběr. TOPICOBJ se používá jako rychlý způsob odkazování na řetězec tématu, který jste již definovali. Vytvořený odběr již neodkazuje na objekt tématu.

```

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)

```

Obrázek 40. Odstranit a vytvořit odběry: fullsubs.tst

Vytvořte klastr se dvěma úložišti. Vytvořte dvě dílčí úložiště pro publikování a odběr. Znovu spusťte skript, abyste vše odstranili, a začněte znovu. Skript také vytvoří hierarchii témat a počáteční odběry se zástupnými znaky.

Poznámka:

Na jiných platformách napište podobný skript nebo zadejte všechny příkazy. Použití skriptu umožňuje rychle odstranit vše a začít znovu s identickou konfigurací.

```

@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

```

Obrázek 41. Vytvořit správce front: qmgrs.bat

Aktualizujte konfiguraci přidáním odběrů do témat klastru.

```

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

```

Obrázek 42. Odběry aktualizací: upsubs.bat

Spuštěním příkazu pub.bat parametrem správce front publikujte zprávy obsahující řetězec tématu publikování. Produkt Pub.bat používá ukázkový program **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
```

Obrázek 43. Publikovat: pub.bat

Související pojmy

[Odběry se zástupnými znaky a zachovaná publikování](#)

Obor publikování

Při konfiguraci klastru nebo hierarchie publikování/odběru obor publikování dále řídí, zda mají správci front předávat publikování vzdáleným správcům front. Pomocí atributu tématu **PUBSCOPE** můžete spravovat rozsah publikování.

Není-li publikování předáno vzdáleným správcům front, obdrží publikování pouze lokální odběratelé.

Při použití klastru publikování/odběru je rozsah publikování primárně řízen definicí klastrovaných objektů tématu v určitých bodech stromu témat. Obor publikování musí být nastaven tak, aby umožňoval tok publikování do jiných správců front v klastru. Rozsah publikování pro klastrované téma byste měli omezit pouze v případě, že potřebujete podrobné řízení specifických témat v určitých správcích front.

Při použití hierarchie publikování/odběru je rozsah publikování primárně řízen tímto atributem v kombinaci s atributem [rozsah odběru](#).

Atribut **PUBSCOPE** se používá k určení rozsahu publikování pro specifické téma. Atribut můžete nastavit na jednu z následujících hodnot:

QMGR

Publikace je doručena pouze lokálním odběratelům. Tyto publikace se nazývají *lokální publikace*. Lokální publikace nejsou předávány vzdáleným správcům front, a proto nejsou přijímány odběrateli připojenými ke vzdáleným správcům front.

ALL

Publikování je doručeno lokálním odběratelům a odběratelům připojeným ke vzdáleným správcům front v klastru nebo hierarchii publikování/odběru. Tyto publikace se nazývají *globální publikace*.

AsParent

Použijte nastavení **PUBSCOPE** nadřazeného tématu ve stromu témat.

Vydavatelé mohou také určit, zda je publikování lokální nebo globální, pomocí volby MQPMO_SCOPE_QMGR vložení zprávy. Je-li použita tato volba, potlačí jakékoli chování, které bylo nastaveno pomocí atributu tématu **PUBSCOPE**.

Související pojmy

[“Objekty administrativních témat” na stránce 73](#)

Pomocí objektu administrativního tématu můžete k tématům přiřadit specifické, jiné než výchozí atributy.

Související úlohy

[Konfigurace distribuovaných sítí publikování/odběru](#)

Obor odběru

Obor odběru řídí, zda odběr v jednom správci front přijímá publikování publikovaná v jiném správci front v klastru či hierarchii publikování/odběru, nebo pouze publikování od lokálních vydavatelů.

Omezení rozsahu odběru na správce front zastaví předávání proxy odběrů jiným správcům front v topologii publikování/odběru. Tím se snižuje provoz systému zpráv publikování/odběru mezi správci front.

Při použití klastru publikování/odběru je rozsah odběrů primárně řízen definicí klastrovaných objektů tématu v určitých bodech stromu témat. Obor odběru musí být nastaven tak, aby umožňoval tok proxy

odběrů pro ostatní správce front v klastru. Rozsah odběru pro klastrované téma byste měli omezit pouze v případě, že potřebujete podrobné řízení specifických témat v určitých správcích front.

Při použití hierarchie publikování/odběru je rozsah odběrů primárně řízen tímto atributem v kombinaci s atributem rozsah publikování.

Atribut tématu **SUBSCOPE** se používá k určení rozsahu odběrů provedených pro specifické téma. Atribut můžete nastavit na jednu z následujících hodnot:

QMGR

Odběr přijímá pouze lokální publikování a proxy odběry nejsou šířeny do vzdálených správců front.

ALL

Proxy odběr je šířen do vzdálených správců front v klastru nebo hierarchii publikování/odběru a odběratel přijímá lokální a vzdálená publikování.

AsParent

Použijte nastavení **SUBSCOPE** nadřazeného tématu ve stromu témat.

Je-li rozsah odběru pro téma nastaven na hodnotu ALL, a to buď přímo, nebo prostřednictvím volby ASPARENT, mohou jednotlivé odběry pro dané téma omezit rozsah na QMGR zadáním hodnoty MQSO_SCOPE_QMGR při vytváření odběru. Odběr tématu, které má rozsah QMGR, nemůže rozšířit rozsah na ALL.

Související pojmy

[“Objekty administrativních témat” na stránce 73](#)

Pomocí objektu administrativního tématu můžete k tématům přiřadit specifické, jiné než výchozí atributy.

Související úlohy

[Konfigurace distribuovaných sítí publikování/odběru](#)

Prostory témat

Prostor tématu je sada témat, pro která se můžete přihlásit k odběru a publikovat. Správce front v distribuované topologii publikování/odběru má prostor tématu, který potenciálně zahrnuje témata, která byla přihlášená k odběru a publikována v připojených správcích front v této topologii.

Poznámka: Přehled témat v rámci správce front, jako jsou objekty administrativních témat, řetězce témat a stromy témat, viz [“Témata” na stránce 64](#). Další odkazy na *témata* v aktuálním článku viz [řetězce témat](#), není-li uvedeno jinak.

Témata jsou nejprve vytvořena jedním z následujících způsobů:

- administrativně, když definujete objekt tématu nebo trvalý odběr.
- dynamicky, když aplikace dynamicky vytvoří publikování nebo odběr nového tématu.

Témata jsou šířena do jiných správců front prostřednictvím proxy odběrů i vytvářením objektů témat administrativního klastru. Proxy odběry vedou k předání publikování ze správce front, ke kterému je připojen vydavatel, správcům front odběratelů.

Proxy odběry jsou šířeny mezi všemi správci front, kteří jsou navzájem propojeni vztahy nadřazený-podřazený v hierarchii správců front. Výsledkem je, že se můžete přihlásit k odběru jednoho správce front pro téma definované v libovolném jiném správci front v hierarchii. Dokud existuje připojená cesta mezi správci front, nezáleží na tom, jak jsou tyto správci front připojeni.

Proxy odběry jsou také šířeny pro odběry do témat klastru v klastru publikování/odběru. Téma klastru je téma, které je připojeno k objektu tématu, který má atribut **CLUSTER**, nebo zdědí atribut od nadřazeného objektu. Témata, která nejsou tématy klastru, jsou označována jako lokální témata a nejsou replikována do klastru. Do klastru nejsou šířeny žádné proxy odběry z odběrů do lokálních témat.

Chcete-li to shrnout, proxy odběry jsou vytvořeny pro odběratele za dvou okolností.

1. Správce front je členem hierarchie a proxy odběr je předán nadřazenému a podřazenému správci front.
2. Správce front je členem klastru a řetězec tématu odběru se interpretuje jako téma přidružené k objektu tématu klastru. Je-li téma *přímo směřované* téma klastru, jsou proxy odběry postoupeny

všem členům klastru. Jedná-li se o téma *směřované hostitelem tématu*, jsou proxy odběry předávány pouze správčům front v klastru, kteří definovali objekt klastrovaného tématu. Další informace viz [“Klastry publikování/odběru”](#) na stránce 88.

Je-li správce front členem klastru a hierarchie, jsou proxy odběry šířeny oběma mechanismy bez doručování duplicitních publikování odběrateli.

Prostory témat tří topologií publikování/odběru jsou popsány v následujícím seznamu:

- [“Případ 1. Klastry publikování/odběru”](#) na stránce 101.
- [“Případ 2. Hierarchie publikování/odběru”](#) na stránce 101.

Následující konfigurační úlohy v samostatných tématech popisují, jak kombinovat prostory tématu.

- [Vytvoření jednoho prostoru tématu v klastru publikování/odběru.](#)
- [Kombinace prostorů témat více klastrů.](#)
- [Kombinace a izolace prostorů témat ve více klastrech.](#)
- [Publikování a přihlášení k odběru prostorů témat ve více klastrech.](#)

Případ 1. Klastry publikování/odběru

V tomto příkladu předpokládejme, že správce front není připojen k hierarchii publikování/odběru.

Pokud je správce front členem klastru publikování/odběru, jeho prostor tématu je tvořen lokálními tématy a tématy klastru. Lokální témata jsou přidružena k objektům témat bez atributu **CLUSTER**. Pokud má správce front lokální definice objektů témat, jeho prostor tématu se liší od jiného správce front v klastru, který má také vlastní lokálně definované objekty témat.

V klastru publikování/odběru se nemůžete přihlásit k odběru tématu definovaného v jiném správci front, pokud se téma, k jehož odběru se přihlásíte, nevyřeší na objekt tématu klastru.

Jsou-li ve více správcích front vyžadovány stejné pojmenované definice objektu tématu klastru, například při použití *směrování hostitele tématu*, je důležité, aby se všechny definice v případě potřeby shodovaly. Další informace naleznete v tématu [Vytvoření jednoho prostoru tématu v klastru publikování/odběru](#).

Lokální definice objektu tématu, bez ohledu na to, zda je definice určena pro téma klastru nebo pro lokální téma, má přednost před stejným objektem tématu definovaným jinde v klastru. Použije se lokálně definované téma, i když je objekt definovaný na jiném místě novější.

Je důležité, aby byl objekt tématu klastru přidružen ke stejnému řetězci tématu všude v klastru. Nelze upravit řetězec tématu, ke kterému je objekt tématu přidružen. Chcete-li přidružit stejný objekt tématu k jinému řetězci tématu, musíte odstranit objekt tématu a znovu jej vytvořit s novým řetězcem tématu. Pokud je téma klastrované, efekt je odstranit kopie objektu tématu uložené v ostatních členech klastru a poté vytvořit kopie nového objektu tématu všude v klastru. Všechny kopie objektu tématu odkazují na stejný řetězec tématu.

Je možné náhodně vytvořit dvě definice stejného pojmenovaného objektu tématu v různých správcích front v klastru s různými řetězci témat. To může mít za následek matoucí chování, protože více definic stejného objektu tématu s různými řetězci tématu může vést k různým výsledkům v závislosti na tom, jak a kde je téma odkazováno. Další informace o tomto důležitém bodu viz [Vícenásobné definice témat klastru se stejným názvem](#).

Případ 2. Hierarchie publikování/odběru

V tomto příkladu předpokládejme, že správce front není členem klastru publikování/odběru.

Pokud je v produktu IBM MQ správce front členem hierarchie publikování/odběru, jeho prostor tématu se skládá ze všech témat definovaných lokálně a v připojených správcích front. Prostor tématu všech správců front v hierarchii je stejný. Není rozdělení témat na lokální témata a globální témata.

Nastavte jednu z voleb **PUBSCOPE** a **SUBSCOPE** na QMGR, abyste zabránili publikování v tématu, které proudí z vydavatele na odběratele připojeného k různým správčům front v hierarchii.

Předpokládejme, že definujete objekt tématu Alabama s řetězcem tématu USA/Alabama ve správci front QMA. Výsledek je následující:

1. Prostor tématu na adrese QMA nyní obsahuje objekt tématu Alabama a řetězec tématu USA/Alabama.
2. Aplikace nebo administrátor může vytvořit odběr na adrese QMA s použitím názvu objektu tématu Alabama.
3. Aplikace může vytvořit odběr libovolného tématu, včetně tématu USA/Alabama, v libovolném správci front v hierarchii. Pokud QMA nebylo definováno lokálně, téma USA/Alabama se interpretuje jako objekt tématu SYSTEM.BASE.TOPIC.

Související pojmy

[“Obor publikování” na stránce 99](#)

Při konfiguraci klastru nebo hierarchie publikování/odběru obor publikování dále řídí, zda mají správci front předávat publikování vzdáleným správcům front. Pomocí atributu tématu **PUBSCOPE** můžete spravovat rozsah publikování.

[“Obor odběru” na stránce 99](#)

Obor odběru řídí, zda odběr v jednom správci front přijímá publikování publikovaná v jiném správci front v klastru či hierarchii publikování/odběru, nebo pouze publikování od lokálních vydavatelů.

Související úlohy

[Konfigurace distribuovaných sítí publikování/odběru](#)

IBM MQ Výběrové vysílání

Výběrové vysílání produktu IBM MQ nabízí spolehlivé výběrové vysílání zpráv s nízkou latencí a vysokým stupněm větvení.

Výběrové vysílání je efektivní forma systému zpráv publikování/odběru, protože jej lze škálovat na vysoký počet odběratelů bez nepříznivých dopadů na výkon. Produkt IBM MQ umožňuje spolehlivý výběrový přenos zpráv s použitím potvrzení, negativních potvrzení a pořadových čísel, který dosahuje nízké latence přenosu zpráv a vysokého stupně větvení.

Rovnoměrné doručování výběrového vysílání IBM MQ umožňuje dosáhnout prakticky současného doručení, takže žádný příjemce nezíská výhodu před ostatními. Jelikož výběrové vysílání produktu IBM MQ využívá k doručování zpráv síť, k rozdělování dat do větví není nutné použít stroj pro publikování/odběr. Po namapování tématu na adresu skupiny není třeba správce front používat, protože vydavatelé a odběratelé mohou pracovat v režimu peer-to-peer. Tento přístup umožňuje snížit zatížení serverů správců front, takže servery správců front přestanou tvořit potenciální bod selhání.

Počáteční koncepce výběrového vysílání

Výběrové vysílání IBM MQ lze snadno integrovat do existujících systémů a aplikací pomocí objektu Informace o komunikaci (COMMINFO). Dvě pole objektů TOPIC umožňují rychlou konfiguraci existujících objektů TOPIC pro podporu nebo ignorování přenosů výběrového vysílání.

Objekty potřebné pro výběrové vysílání

Následující informace obsahují stručný přehled dvou objektů potřebných pro výběrové vysílání produktu IBM MQ :

Objekt COMMINFO

Objekt COMMINFO obsahuje atributy přidružený k přenosu výběrového vysílání. Další informace o parametrech objektu COMMINFO viz [DEFINE COMMINFO](#).

Jediné pole COMMINFO, které MUSÍ být nastaveno, je název objektu COMMINFO. Tento název se pak používá k identifikaci objektu COMMINFO pro téma. Pole **GRPADDR** objektu COMMINFO musí být zkontrolováno, aby se zajistilo, že hodnota je platná adresa skupiny výběrového vysílání.

Objekt tématu

Téma je předmětem informací publikovaných ve zprávě publikování/odběru a téma je definováno vytvořením objektu TOPIC. Další informace o parametrech objektu TOPIC viz [DEFINE TOPIC](#).

Existující témata lze použít s výběrovým vysíláním změnou hodnot následujících parametrů objektu TOPIC: **COMMINFO** a **MCAST**.

- **COMMINFO** Tento parametr určuje název objektu informací o komunikaci výběrového vysílání.
- **MCAST** Tento parametr určuje, zda je výběrové vysílání povoleno na této pozici ve stromu témat. Standardně je parametr **MCAST** nastaven na hodnotu **ASPARENT**, což znamená, že atribut výběrového vysílání tématu je zděděn od nadřazeného prvku. Nastavení parametru **MCAST** na hodnotu **ENABLED** povolí provoz výběrového vysílání v tomto uzlu.

Sítě a témata výběrového vysílání

Následující informace jsou přehledem toho, co se stane s odběry s různými typy odběrů a definicí tématu. Všechny tyto příklady předpokládají, že parametr **COMMINFO** objektu TOPIC je nastaven na název platného objektu **COMMINFO**:

Téma nastaveno na výběrové vysílání povoleno

Je-li parametr **MCAST** řetězce tématu nastaven na hodnotu **ENABLED**, jsou povoleny odběry z klientů s možností výběrového vysílání a je proveden odběr výběrového vysílání, pokud:

- Jedná se o trvalý odběr z klienta s možností výběrového vysílání.
- Jedná se o nespravovaný odběr od klienta s možností výběrového vysílání.
- Jedná se o odběr od klienta, který nepodporuje výběrové vysílání.

V těchto případech je proveden odběr bez výběrového vysílání a odběry jsou sníženy na normální publikování/odběr.

Téma nastaveno na výběrové vysílání zakázáno

Pokud je parametr **MCAST** řetězce tématu nastaven na hodnotu **DISABLED**, vždy se provede odběr bez výběrového vysílání a odběry jsou sníženy na normální publikování/odběr.

Téma je nastaveno pouze na výběrové vysílání

Je-li parametr **MCAST** řetězce tématu nastaven na hodnotu **ONLY**, jsou povoleny odběry z klientů s možností výběrového vysílání a je proveden odběr výběrového vysílání, pokud:

- Jedná se o trvalý odběr: Trvalé odběry jsou odmítnuty s kódem příčiny [2436 \(0984\) \(RC2436\): MQRC_DURABILITY_NOT_ALLOWED](#)
- Jedná se o nespravovaný odběr: Spravované odběry jsou odmítnuty s kódem příčiny [2046 \(07FE\) \(RC2046\): MQRC_OPTIONS_ERROR](#)
- Jedná se o odběr od klienta, který nepodporuje výběrové vysílání: Tyto odběry jsou odmítnuty s kódem příčiny [2560 \(0A00\) \(RC2560\): MQRC_MULTICAST_ONLY](#)
- Jedná se o odběr z lokálně vázané aplikace: Tyto odběry jsou odmítnuty s kódem příčiny [2560 \(0A00\) \(RC2560\): MQRC_MULTICAST_ONLY](#)

Windows

Linux

AIX

Přehled produktu MQ Telemetry

Produkt MQ Telemetry obsahuje službu telemetrie (MQXR), která je součástí správce front, klienty telemetrie, které si můžete sami zapsat nebo stáhnout zdarma, a administrativní rozhraní příkazového řádku a průzkumníku. Telemetrie se týká shromažďování dat z a správy širokého spektra vzdálených zařízení. Pomocí produktu MQ Telemetry můžete integrovat shromažďování dat a řízení zařízení s webovými aplikacemi.

MQ Telemetry je komponentou produktu IBM MQ. Upgrade pro tyto verze v podstatě instaluje novější verzi produktu IBM MQ.

Ukázkové aplikace jsou i nadále volně dostupné z prostředí Eclipse Paho a [MQTT.org](#). Viz [IBM MQ Telemetry Transport ukázkové programy](#).

Protože MQ Telemetry je komponentou produktu IBM MQ, lze produkt MQ Telemetry nainstalovat spolu s hlavním produktem nebo po instalaci hlavního produktu. Informace o migraci viz [Migrace MQ Telemetry na Windows](#) a [Migrace MQ Telemetry na Linux](#).

V produktu MQ Telemetry jsou zahrnuty následující komponenty:

Kanály telemetrie

Pomocí kanálů telemetrie můžete spravovat připojení klientů MQTT k produktu IBM MQ . Kanály telemetrie používají k interakci s produktem IBM MQ nové objekty IBM MQ , například objekty SYSTEM . MQTT . TRANSMIT . QUEUE .

Služba (MQXR) telemetrie

Klienti MQTT používají službu telemetrie SYSTEM . MQXR . SERVICE pro připojení ke kanálům telemetrie.

IBM MQ Explorer podpora MQ Telemetry

MQ Telemetry lze spravovat pomocí IBM MQ Explorer.

Dokumentace

Dokumentace k produktu MQ Telemetry je součástí standardní dokumentace k produktu IBM MQ . Dokumentace SDK pro klienty Java a C je poskytována v dokumentaci k produktu a ve formátu Javadoc a HTML.

Koncepce telemetrie

Shromažďujete informace z prostředí všude kolem sebe, abyste se rozhodli, co dělat. Jako spotřebitel zkontrolujte, co máte v obchodě, než se rozhodnete, co jídlo koupit. Chcete vědět, jak dlouho bude cesta trvat, pokud odejdete nyní, před rezervací připojení. Než se rozhodnete, zda navštívit lékaře, zkontrolujte své příznaky. Můžete zkontrolovat, kdy autobus se chystá přijet, před rozhodnutím, zda čekat. Informace o těchto rozhodnutích pocházejí přímo z měřidel a zařízení, z psaného slova na papíře nebo z obrazovky a od vás. Všude, kde jste a kdy potřebujete, shromažďujete informace, shromažďujete je, analyzujete a jednáte na nich.

Pokud jsou zdroje informací široce rozptýlené nebo nepřístupné, je shromažďování nejpřesnějších informací obtížné a nákladné. Pokud existuje mnoho změn, které chcete provést, nebo je obtížné provést změny, pak se změny neprovedou nebo se provedou, když jsou méně účinné.

Co v případě, že náklady na shromažďování informací z, a řízení, široce rozptýlené zařízení je výrazně snížena tím, že připojení zařízení s digitální technologií na internetu? Informace lze analyzovat pomocí prostředků Internetu a podniku. Máte více příležitostí činit informovaná rozhodnutí a jednat podle nich.

Technologické trendy a environmentální a ekonomické tlaky jsou hybnou silou těchto změn:

1. Náklady na připojení a řízení senzorů a akčních členů se snižují díky standardizaci a připojení k nízkonákladovým digitálním procesorům.
2. Internet a internetové technologie se stále více používají pro připojení zařízení. V některých zemích mobilní telefony překračují počet osobních počítačů v počtu připojení k internetovým aplikacím. Ostatní zařízení jsou jistě následovat.
3. Internet a internetové technologie usnadňují aplikaci získat data. Snadný přístup k datům je hnací silou využití analýzy dat k přeměně dat ze senzorů na informace, které jsou užitečné v mnoha dalších řešeních.
4. Inteligentní využívání zdrojů je často rychlejší a levnější způsob snižování emisí uhlíku a nákladů. Alternativy: nalezení nových zdrojů nebo vývoj nových technologií pro využití stávajících zdrojů by mohly být dlouhodobým řešením. V krátkodobém horizontu je vývoj nových technologií nebo nalezení nových zdrojů často riskantnější, pomalejší a nákladnější než zlepšení stávajících řešení.

Příklad

Příklad ukazuje, jak tyto trendy vytvářejí nové příležitosti k inteligentní interakci s prostředím.

Mezinárodní úmluva o bezpečnosti lidského života na moři (SOLAS) vyžaduje, aby byl systém automatické identifikace (AIS) nasazen na mnoha lodích. Je vyžadováno na obchodních lodích nad 300 tun a osobních

lodích. AIS je především systém předcházení srážkám pro pobřežní plavbu. Používá se mořskými orgány ke sledování a kontrole pobřežních vod.

Nadšenci po celém světě nasazují nízkonákladové sledovací stanice AIS a umísťují informace o pobřežní lodní dopravě na internet. Další nadšenci píší aplikace, které kombinují informace z AIS s dalšími informacemi z internetu. Výsledky jsou umístěny na webových stránkách a publikovány pomocí Twitteru a SMS.

V jedné aplikaci jsou informace ze stanic AIS poblíž Southamptonu kombinovány s vlastnictvím lodí a geografickými informacemi. Aplikace poskytuje živé informace o příjezdech a odjezdech trajektů na Twitter. Pravidelné dojíždějící pomocí trajekty mezi Southampton a Isle of Wight přihlásit do zpravodajského kanálu pomocí Twitter nebo SMS. Pokud krmivo ukazuje, že jejich trajekt je pozdě, dojíždějící mohou odložit svůj odlet a chytit trajekt, když doky později, než je plánovaný čas příjezdu.

Další příklady viz [“Případy použití telemetrie”](#) na stránce 106.

Související úlohy

[Instalace produktu MQ Telemetry](#)

[Správa serveru MQ Telemetry](#)

[Migrace MQ Telemetry na Windows](#)

[Migrace MQ Telemetry na Linux](#)

[Vyvíjení aplikací pro MQ Telemetry](#)

[Odstraňování problémů s produktem MQ Telemetry](#)

Související odkazy

[Odkaz na MQ Telemetry](#)

Windows

Linux

AIX

Úvod do produktu MQ Telemetry

Lidé, podniky a vlády stále více chtějí používat produkt MQ Telemetry k chytnějšímu interakci s prostředím, ve kterém žijeme a ve kterém pracujeme. Produkt MQ Telemetry připojuje všechny druhy zařízení k Internetu a k podniku a snižuje náklady na sestavení aplikací pro inteligentní zařízení.

Co je to MQ Telemetry?

- Jedná se o funkci produktu IBM MQ, která rozšiřuje univerzální páteř systému zpráv poskytovanou produktem IBM MQ na širokou škálu vzdálených senzorů, akčních členů a telemetrických zařízení. Produkt MQ Telemetry rozšiřuje produkt IBM MQ tak, aby mohl propojit inteligentní podnikové aplikace, služby a osoby s rozhodovacími pravomocemi se sítěmi instrumentovaných zařízení.
- Základní části produktu MQ Telemetry jsou:

Služba MQ Telemetry (MQXR).

Tato služba je spuštěna na serveru IBM MQ a používá protokol IBM MQ Telemetry Transport (MQTT) ke komunikaci s telemetrickými zařízeními.

MQTT aplikací, které píšete.

Tyto aplikace řídí informace přenášené mezi telemetrickými zařízeními a správcem front IBM MQ a všechny akce, které jsou prováděny v reakci na tyto informace. Chcete-li pomoci vytvořit tyto aplikace, použijte knihovny klienta MQTT.

Co to pro mě může udělat?

- MQTT je otevřený přenos systému zpráv, který umožňuje vytváření implementací produktu MQTT pro širokou škálu zařízení.
- Klienti MQTT mohou pracovat na malých zařízeních obsazených obsazením prostoru, která mají omezené prostředky.
- Produkt MQTT pracuje efektivně na sítích, kde je šířka pásma nízká, kde jsou náklady na odesílání dat drahé nebo které mohou být křehké.
- Doručení zpráv je zajištěno a odpojeno od aplikace.

- Programátoři aplikací nemusí mít znalosti programování komunikací.
- Zprávy lze vyměňovat s jinými aplikacemi systému zpráv. Může se jednat o jinou aplikaci telemetrie nebo o aplikaci MQI, JMS nebo podnikovou aplikaci systému zpráv.

Jak ji mám používat?

- K dispozici jsou ukázkové skripty, které pracují s ukázkovou aplikací klienta IBM MQ Telemetry Transport v3 (mqttv3app.jar). Viz [IBM MQ Telemetry Transport ukázkové programy](#).
- Pomocí produktu IBM MQ Explorer a jeho přidružených nástrojů spravujte funkci telemetrie produktu IBM MQ.
- Knihovny klienta vám pomohou vytvořit aplikace MQTT, které se připojují ke správci front a používají systém zpráv publikování/odběru.
- Distribuujte svou knihovnu aplikací a klientů na zařízení, kde má být aplikace spuštěna.

Jak to funguje?

- MQTT je protokol publikování a odběru. Klientská aplikace MQTT může publikovat zprávy na server MQTT nebo se přihlásit k odběru zpráv odeslaných aplikacemi, které se připojují k serveru MQTT.
- Klientské aplikace MQTT používají knihovny klienta, které implementují přenos zpráv MQTT.
- Základní klientská aplikace MQTT funguje jako standardní klient produktu MQ, ale může být spuštěna na mnohem širším spektru platforem a sítí.
- Služba MQ Telemetry (MQXR) změní správce front IBM MQ na server MQTT.
- Pokud se správce front IBM MQ chová jako server MQTT, mohou další aplikace, které se připojují ke správci front, odebírat zprávy a přijímat je od klienta MQTT.
- Správce front se chová jako směrovač, který distribuuje zprávy z publikačních aplikací do odebíraných aplikací.
- Zprávy lze distribuovat mezi různé typy klientských aplikací. Například mezi klienty telemetrie a klienty JMS.

Poznámka: Produkt MQ Telemetry nahrazuje uzly SCADA, které byly staženy ve verzi 7 produktu WebSphere Message Broker (nyní známé jako IBM Integration Bus). a běží na systémech Windows, Linuxu a AIX.

Windows

Linux

AIX

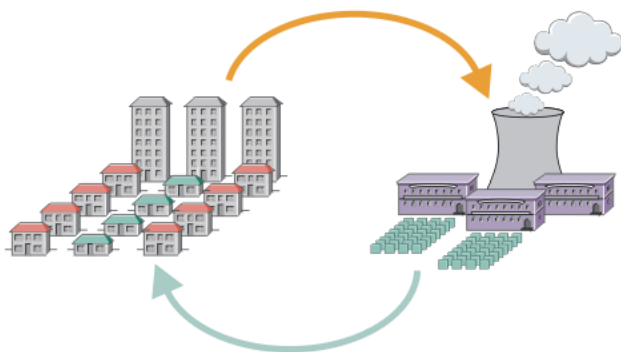
Případy použití telemetrie

Telemetrie je automatické snímání, měření dat a ovládání vzdálených zařízení. Důraz je kladen na přenos dat ze zařízení do centrálního řídicího bodu. Telemetrie také zahrnuje odesílání informací o konfiguraci a řízení na zařízení.

Produkt MQ Telemetry připojuje malá zařízení pomocí produktu MQTT protokola připojuje zařízení k jiným aplikacím pomocí produktu IBM MQ. MQ Telemetry překlenuje mezeru mezi zařízeními a internetem, což usnadňuje vytváření "chytrých řešení". Inteligentní řešení odemykají množství informací dostupných na internetu a v podnikových aplikacích pro aplikace, které monitorují a řídí zařízení.

Následující diagramy ukazují některá typická použití produktu MQ Telemetry:

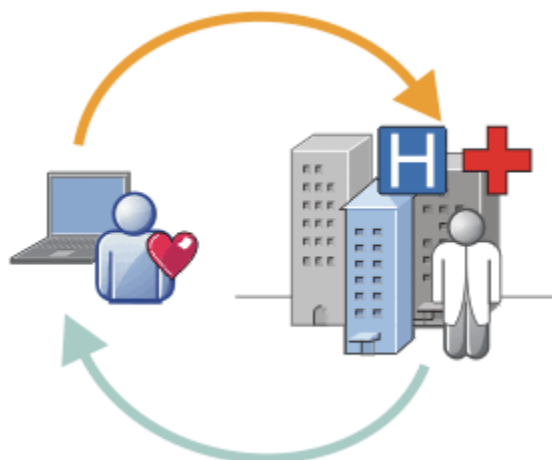
Telemetrie: Inteligentní elektřina



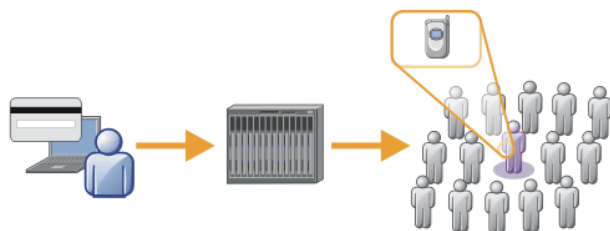
- Zpráva MQTT obsahující data o využití energie odeslaná poskytovateli služeb.
- Produkt MQ Telemetry odesílá příkazy CONTROL COMMANDS na základě analýzy dat o využití energie.
- Další informace naleznete v následujícím příkladu použití: [“Případ použití telemetrie: Monitorování a řízení domácí energie”](#) na stránce 109

Telemetrie: Inteligentní zdravotní služby

- MQ Telemetry odesílá zdravotní údaje do nemocnice a lékaře.
- Výstrahy zpráv MQTT nebo zpětnou vazbu lze odeslat na základě analýzy zdravotních údajů.
- Další informace naleznete v následujícím příkladu použití: [“Případ použití telemetrie: Monitorování domácího pacienta”](#) na stránce 107



telemetrie: jeden v davu



- Na server banky je odeslána jednoduchá karetní transakce.
- Produkt MQ Telemetry identifikuje jednu osobu z tisíců a upozorní zákazníka, že byla použita jejich karta.
- Produkt MQ Telemetry může použít nejjednodušší vstup informací a vyhledat tohoto jednotlivce.

Příklady použití popsané v dílčích tématech jsou čerpány ze skutečných příkladů. Ilustrují některé způsoby využití telemetrie a některé běžné problémy, které musí technologie telemetrie vyřešit.

Windows

Linux

AIX

Případ použití telemetrie: Monitorování domácího pacienta

Ve spolupráci mezi společností IBM a poskytovatelem zdravotní péče na systému péče o srdeční pacienty komunikuje implantovaný kardioverter defibrilátor s nemocnicí. Data o pacientovi a implantovaném zařízení jsou přenášena pomocí RF telemetrie do zařízení MQTT v domácnosti pacienta.

Obvykle se přenos odehrává v noci na vysílaci umístěném na nočním stolku. Vysílač přenáší data bezpečně přes telefonní systém do nemocnice, kde jsou data analyzována.

System snižuje počet návštěv, které musí pacient provést u lékaře. Zjistí, když pacient nebo zařízení potřebuje pozornost, a v případě nouze upozorní lékaře na pohotovost.

Spolupráce mezi produktem IBM a poskytovatelem zdravotní péče má charakteristiky, které jsou společné pro řadu případů použití telemetrie:

Neviditelnost

Zařízení nevyžaduje žádný jiný zásah uživatele než napájení, telefonní linku a být v blízkosti zařízení pro část dne. Jeho provoz je spolehlivý a snadno se používá.

Chcete-li odstranit potřebu pacienta nastavit zařízení, dodavatel zařízení předkonfiguruje zařízení. Pacient ji musí pouze zapojit. Eliminace konfigurace pacientem zjednodušuje provoz zařízení a snižuje pravděpodobnost, že je zařízení nakonfigurováno nesprávně.

Klient MQTT je vestavěn jako součást zařízení. Vývojář zařízení vloží implementaci klienta MQTT do zařízení a vývojář nebo dodavatel nakonfiguruje klienta MQTT jako součást předkonfigurace.

Klient MQTT je dodáván jako soubor Java SE JAR, který vývojář zahrne do své aplikace Java. V jiných prostředích než Java, jako je toto, může vývojář zařízení implementovat klienta v jiném jazyce pomocí publikovaných formátů a protokolu MQTT. Alternativně může vývojář použít jednoho z klientů jazyka C dodávaných jako sdílené knihovny pro platformy Windows, Linux a ARM.

Nerovnoměrná konektivita

Komunikace mezi defibrilátorem a nemocnicí má nerovnoměrné síťové charakteristiky. Dvě různé sítě se používají k řešení různých problémů sběru dat od pacienta a odesílání dat do nemocnice. Mezi pacientem a zařízením MQTT se používá nízkoenergetická RF síť krátkého dosahu. Vysílač se připojí k nemocnici pomocí připojení VPN TCP/IP přes telefonní linku s nízkou šířkou pásma.

Často je nepraktické najít způsob, jak připojit každé zařízení přímo k síti Internet Protocol. Použití dvou sítí, propojených rozbočovačem, je běžným řešením. Zařízení MQTT je jednoduchý rozbočovač, který ukládá informace od pacienta a předává je do nemocnice.

Zabezpečení

Lékař musí být schopen důvěřovat autenticitě údajů o pacientech a pacient chce, aby bylo respektováno soukromí jejich údajů.

V některých situacích stačí šifrovat připojení pomocí VPN nebo TLS. V jiných situacích je žádoucí uchovat data v bezpečí i po jejich uložení.

Někdy není telemetrické zařízení bezpečné. Může být například ve sdíleném obydlí. Uživatel zařízení musí být ověřen, aby se ujistil, že data jsou od správného pacienta. Samotné zařízení lze ověřit na serveru pomocí TLS a server ověřený na zařízení.

Kanál telemetrie mezi zařízením a správcem front podporuje službu JAAS pro ověřování uživatelů a protokol TLS pro šifrování komunikace a ověřování zařízení. Přístup k publikování je řízen správcem oprávnění k objektu v adresáři IBM MQ.

Identifikátor použitý k ověření uživatele může být mapován na jiný identifikátor, například na společnou identitu pacienta. Společný identifikátor zjednodušuje konfiguraci autorizace k tématům publikování v produktu IBM MQ.

Konektivita

Připojení mezi zařízením MQTT a nemocnicí používá telefonické připojení a pracuje s šířkou pásma až 300 baudů.

Aby systém MQTT protocol fungoval efektivně při 300 baud, přidá ke zprávě kromě záhlaví TCP/IP pouze několik dalších bajtů.

System zpráv MQTT protocol poskytuje jeden přenos, *spustí se a zapomene*, což udržuje nízké latence. Může také použít více přenosů k zajištění *alespoň jednou a přesně jednou* doručení, pokud je zaručené doručení důležitější než doba odezvy. Pro zaručené doručení jsou zprávy uloženy v zařízení, dokud nejsou úspěšně doručeny. Pokud je zařízení připojeno bezdrátově, zaručené doručení je obzvláště užitečné.

Rozšiřitelnost

Telemetrická zařízení jsou obvykle nasazena ve velkém počtu, od desítek tisíc až po miliony.

Připojení mnoha zařízení k systému klade velké nároky na řešení. Existují obchodní požadavky, jako jsou náklady na zařízení a jejich software a administrativní požadavky na správu licencí, zařízení a uživatelů. Technické požadavky zahrnují zatížení sítě a serverů.

Při otevírání připojení je použito více prostředků serveru než při udržování otevřených připojení. Ale v případě použití, jako je tento, který používá telefonní linky, náklady na připojení znamenají, že připojení jsou ponechána otevřená déle, než je požadováno. Přenos dat je z velké části dávkového charakteru. Připojení lze naplánovat po celou noc, aby se zabránilo náhlému maximu připojení před spaním.

Na straně klienta je rozšiřitelnost klientů nápomocna minimální požadovanou konfigurací klienta. Klient MQTT je vložen do zařízení. Neexistuje žádný požadavek na konfiguraci nebo krok přijetí licence klienta MQTT, který by byl integrován do implementace zařízení pro pacienty.

Na serveru má produkt MQ Telemetry počáteční cíl 50 000 otevřených připojení na jednoho správce front.

Připojení jsou spravována pomocí konzoly IBM MQ Explorer. Produkt IBM MQ Explorer filtruje připojení, která se mají zobrazit, na spravovatelné číslo. S vhodně zvoleným schématem přidělování identifikátorů klientům můžete filtrovat připojení na základě zeměpisné oblasti nebo abecedně podle jména pacienta.

Windows Linux AIX **Případ použití telemetrie: Monitorování a řízení domácí energie**

Chytré měřiče shromažďují více podrobností o spotřebě energie než tradiční měřiče.

Chytré měřiče jsou často spojeny s lokální telemetrickou sítí pro sledování a ovládání jednotlivých spotřebičů v domácnosti. Některé jsou také vzdáleně připojeny pro monitorování a ovládání na dálku.

Vzdálené připojení může být nastaveno jednotlivcem, napájecí sítí nebo centrálním řídicím bodem. Vzdálený řídicí bod může číst spotřebu energie a poskytovat data o využití. Může poskytovat data, která mají vliv na využití, jako jsou průběžné ceny a informace o počasí. Může omezit zatížení, aby se zlepšila celková účinnost výroby energie.

Inteligentní měřidla začínají být široce nasazena. Britská vláda například konzultuje zavedení inteligentních měřidel do každého britského domu do roku 2020.

Domácí měření případy použití mají řadu společných charakteristik:

Neviditelnost

Pokud uživatel nechce být zapojen do úspory energie pomocí měřidla, měřidlo nesmí vyžadovat zásah uživatele. Nesmí snižovat spolehlivost dodávek energie jednotlivým spotřebičům.

Klient MQTT může být vložen do softwaru nasazeného s měřidlem a nevyžaduje oddělenou instalaci nebo konfiguraci.

Nerovnoměrná konektivita

Komunikace mezi přístroji a inteligentním měřidlem vyžaduje jiné standardy konektivity než mezi měřidlem a vzdáleným přípojným bodem.

Připojení z inteligentního měřiče k přístrojům musí být vysoce dostupné a musí odpovídat síťovým standardům pro domácí počítačovou síť.

Vzdálená síť pravděpodobně používá různá fyzická připojení. Některé z nich, například buněčné, mají vysoké náklady na přenos a mohou být přerušované. Specifikace MQTT v3 je zaměřena na vzdálená připojení a připojení mezi lokálními adaptéry a inteligentním měřidlem.

Spojení mezi napájecími zásuvkami a aplikacemi a měřidlem používá domácí síť, jako je například Zigbee. Produkt MQTT pro senzorové sítě (MQTT-S) je navržen pro práci se systémem Zigbee a dalšími

síťovými protokoly s nízkou šířkou pásma. MQ Telemetry nepodporuje přímo MQTT-S. Vyžaduje bránu pro připojení MQTT-S k MQTT v3.

Stejně jako monitorování domácích pacientů, řešení pro monitorování a kontrolu domácí energie vyžadují více sítí, připojených pomocí inteligentního měřidla jako rozbočovače.

Zabezpečení

K inteligentním měřidlem je přidružena řada bezpečnostních problémů. Mezi tyto problémy patří neodmítání transakcí, autorizace všech iniciovaných kontrolních akcí a soukromí dat o spotřebě energie.

Pro zajištění soukromí lze data přenášená mezi měřidlem a vzdáleným řídicím bodem produktem MQTT šifrovat pomocí TLS. Chcete-li zajistit autorizaci řídicích akcí, lze připojení MQTT mezi měřidlem a vzdáleným řídicím bodem vzájemně ověřovat pomocí TLS.

Konektivita

Fyzická povaha vzdálené sítě se může značně lišit. Může používat existující širokopásmové připojení nebo mobilní síť s vysokými náklady na volání a občasnou dostupnost. Pro vysoké náklady, občasně, připojení MQTT je efektivní a spolehlivý protokol; viz [“Případ použití telemetrie: Monitorování domácího pacienta”](#) na stránce 107.

Rozšiřitelnost

Nakonec energetické společnosti nebo centrální kontrolní body plánují rozmístit desítky milionů inteligentních měřidel. Zpočátku je počet metrů na nasazení v desítkách až stovkách tisíc. Tento počet je srovnatelný s počátečním cílem MQTT 50 000 otevřených připojení klienta na jednoho správce front.

Kritickým aspektem architektury pro monitorování a řízení domácí energie je použití inteligentního měřidla jako síťového koncentrátoru. Každý adaptér zařízení je samostatným senzorem. Připojením k lokálnímu rozbočovači pomocí produktu MQTT může rozbočovač koncentrovat datové toky na jednu relaci TCP/IP s centrálním řídicím bodem a také ukládat zprávy na krátkou dobu, aby překonal výpadky relace.

Vzdálená připojení musí být ponechána otevřená v domácích případech použití energie ze dvou důvodů. Za prvé, protože otevírání připojení trvá dlouhou dobu vzhledem k odesílání požadavků. Doba pro otevření mnoha připojení k odeslání požadavků "load-limit" v krátkém intervalu je příliš dlouhá. Za druhé, chcete-li přijímat požadavky na omezení zátěže od energetické společnosti, musí být připojení nejprve otevřeno klientem. S produktem MQTT jsou připojení vždy iniciována klientem a pro příjem požadavků na omezení zátěže od energetické společnosti musí být připojení ponecháno otevřené.

Pokud je rychlost otevírání připojení kritická nebo server iniciuje časově kritické požadavky, řešením je obvykle udržovat mnoho otevřených připojení.

Případy použití telemetrie: identifikace rádiové frekvence (RFID)

RFID je použití vložené značky RFID k bezdrátové identifikaci a sledování objektu. Štítky RFID lze číst až do rozsahu několika metrů a mimo dosah čtečky RFID. Pasivní štítky jsou aktivovány RFID čtečkou. Aktivní značky vysílají bez externí aktivace. Aktivní značky musí mít zdroj napájení. Pasivní značky mohou obsahovat zdroj napájení pro zvýšení jejich rozsahu.

RFID se používá v mnoha aplikacích a typy případů použití se značně liší. Případy použití RFID a monitorování domácích pacientů a monitorování domácí energie a kontrola případů použití mají některé podobnosti a rozdíly.

Neviditelnost

V mnoha případech použití je čtečka RFID nasazena ve velkém počtu a musí pracovat bez zásahu uživatele. Čtečka obsahuje vestavěného klienta MQTT pro komunikaci s centrálním řídicím bodem.

Například v distribučním skladu používá čtečka snímač pohybu k detekci palety. Aktivuje RFID tagy položek na paletě a posílá data a požadavky centrálním aplikacím. Data se používají k aktualizaci

umístění zásob. Požadavky kontrolují, co se stane s paletou, jako je přesun do konkrétního zálivu. Letecké společnosti a systémy letištních zavazadel používají RFID tímto způsobem.

V některých případech použití RFID má čtečka standardní výpočetní prostředí, například Java Platform, Micro Edition (Java ME). V těchto případech může být klient MQTT implementován v odlišném kroku konfigurace po výrobě.

Nerovnoměrná konektivita

Čtečky RFID mohou být odděleny od lokálního řídicího zařízení, které obsahuje klienta MQTT, nebo může každá čtečka vložit klienta MQTT. Obvykle geografické nebo komunikační faktory označují volbu topologie.

Zabezpečení

Soukromí a autenticita jsou bezpečnostní obavy v příloze RFID tagů. RFID štítky jsou nenápadné a mohou být skrytě monitorovány, falšovány nebo manipulovány.

Řešení bezpečnostních problémů RFID zvyšuje možnost nasazení nových řešení RFID. Ačkoli bezpečnostní riziko je ve značce RFID a místní čtenář, použití centrálního zpracování informací navrhuje přístupy pro boj s různými hrozbami. Manipulace se značkami může být například zjištěna dynamickou korelací úrovní zásob vůči dodávkám a odbavením.

Konektivita

Aplikace RFID obvykle zahrnovaly jak dávkové ukládání, tak předávání informací shromážděných ze čteček RFID a okamžitých dotazů. V případě použití distribučního skladu je čtečka RFID neustále připojena. Když je značka přečtena, je publikována spolu s informacemi o čtečce. Aplikace datového skladu publikuje odezvu zpět čtenáři.

V aplikaci datového skladu je síť obvykle spolehlivá a okamžité požadavky mohou používat zprávy *spustit a zapomenout* pro nízký výkon latence. Dávková data úložiště a data předání mohou používat systém zpráv *právě jednou*, aby se minimalizovaly náklady na administraci spojené se ztracenými daty.

Rozšiřitelnost

Pokud aplikace RFID vyžaduje okamžité odpovědi, v řádu sekund nebo dvou, pak musí čtečky RFID zůstat připojené.

Případy použití telemetrie: snímání prostředí

Průzkum životního prostředí využívá telemetrii ke shromažďování informací o hladinách a kvalitě říčních vod, atmosférických znečišťujících látkách a dalších údajích o životním prostředí.

Senzory jsou často umístěny na vzdálených místech, bez přístupu k kabelové komunikaci. Bezdrátová šířka pásma je drahá a spolehlivost může být nízká. Obvykle je řada senzorů prostředí v malé zeměpisné oblasti připojena k lokálnímu monitorovacímu zařízení na bezpečném místě. Lokální připojení mohou být drátová nebo bezdrátová.

Neviditelnost

Senzorová zařízení jsou pravděpodobně méně přístupná, méně napájená a nasazená ve větším počtu než centrální monitorovací zařízení. Senzory jsou někdy "hloupé" a lokální monitorovací zařízení zahrnuje adaptéry pro transformaci a ukládání dat senzoru. Monitorovací zařízení bude pravděpodobně obsahovat počítač s obecným účelem, který podporuje Java Platform, Standard Edition (Java SE) nebo Java Platform, Micro Edition (Java ME). Neviditelnost pravděpodobně nebude hlavním požadavkem při konfiguraci klienta MQTT.

Nerovnoměrná konektivita

Schopnosti senzorů a náklady a šířka pásma vzdáleného připojení obvykle vedou k lokálnímu monitorovacímu rozbočovači připojenému k centrálnímu serveru.

Zabezpečení

Pokud není řešení používáno v případě vojenského nebo obranného použití, není bezpečnost hlavním požadavkem.

Konektivita

Mnoho použití nevyžaduje průběžné monitorování nebo okamžitou dostupnost dat. Data výjimky, jako např. výstraha úrovně zahlcení, musí být okamžitě postoupena. Data senzoru jsou agregována na lokálním monitoru, aby se snížily náklady na připojení a komunikaci, a poté jsou přenášena pomocí plánovaných připojení. Data výjimky jsou postoupena, jakmile jsou zjištěna na monitoru.

Rozšiřitelnost

Senzory jsou soustředěny kolem lokálních rozbočovačů a data senzorů jsou agregována do paketů, které jsou přenášeny podle plánu. Oba tyto faktory snižují zatížení centrálního serveru, které by bylo vynucováno pomocí přímo připojených senzorů.

Windows

Linux

AIX

Případy použití telemetrie: Mobilní aplikace

Mobilní aplikace jsou aplikace, které běží na bezdrátových zařízeních. Zařízení jsou buď generické aplikační platformy, nebo vlastní zařízení.

Obecné platformy zahrnují kapesní zařízení, jako jsou telefony a asistenti osobních údajů, a přenosná zařízení, jako jsou notebooky. Vlastní zařízení používají hardware pro speciální účely přizpůsobené specifickým aplikacím. Zařízení pro záznam doručení balíku "signed-for" je příkladem vlastního mobilního zařízení. Aplikace na vlastních mobilních zařízeních jsou často postaveny na generické softwarové platformě.

Neviditelnost

Implementace vlastních mobilních aplikací je spravována a může zahrnovat konfiguraci klientské aplikace MQTT. Neviditelnost pravděpodobně nebude hlavním požadavkem při konfiguraci klienta MQTT.

Nerovnoměrná konektivita

Na rozdíl od lokální topologie centrálního serveru předchozích případů použití se mobilní klienti připojují vzdáleně. Vrstva klientské aplikace se připojuje přímo k aplikaci v centrálním rozbočovači.

Zabezpečení

S malým fyzickým zabezpečením musí být mobilní zařízení a mobilní uživatel ověřeni. TLS se používá k potvrzení identity zařízení a JAAS k ověření uživatele.

Konektivita

Pokud mobilní aplikace závisí na bezdrátovém pokrytí, musí být schopen pracovat v režimu offline a efektivně se vypořádat s přerušným připojením. V tomto prostředí je cílem zůstat ve spojení, ale aplikace musí být schopna ukládat a předávat zprávy. Zprávy jsou často objednávkami nebo potvrzeními o doručení a mají důležitou obchodní hodnotu. Musí být uloženy a předány spolehlivě.

Rozšiřitelnost

Rozšiřitelnost není hlavním problémem. Počet aplikačních klientů pravděpodobně nepřekročí tisíce, nebo desítky tisíc, v případech použití vlastních mobilních aplikací.

Windows

Linux

AIX

Připojení telemetrických zařízení ke správci

front

Telemetrická zařízení se připojují ke správci front pomocí klienta MQTT v3. Klient MQTT v3 používá protokol TCP/IP pro připojení k modulu listener protokolu TCP/IP, který se nazývá služba telemetrie (MQXR).

Když připojíte zařízení telemetrie ke správci front, klient MQTT zahájí připojení TCP/IP pomocí metody `MqttClient.connect`. Stejně jako klienti IBM MQ musí být klient MQTT připojen ke správci front, aby mohl odesílat a přijímat zprávy. Připojení se provádí na serveru pomocí modulu listener protokolu TCP/IP nainstalovaného s produktem MQ Telemetry, který se nazývá služba telemetrie (MQXR). Každý správce front spouští maximálně jednu službu telemetrie (MQXR).

Služba telemetrie (MQXR) používá adresu vzdáleného socketu nastavenou každým klientem v metodě `MqttClient.connect` k přidělení připojení ke kanálu telemetrie. Adresa socketu je kombinací názvu hostitele TCP/IP a čísla portu. Více klientů používajících stejnou adresu vzdáleného socketu je připojeno ke stejnému kanálu telemetrie pomocí služby telemetrie (MQXR).

Pokud je na serveru více správců front, rozdělte kanály telemetrie mezi správce front. Přidělte adresy vzdálených socketů mezi správci front. Definujte každý kanál telemetrie s jedinečnou adresou vzdáleného socketu. Dva kanály telemetrie nesmí používat stejnou adresu socketu.

Pokud je pro kanály telemetrie ve více správcích front konfigurována stejná adresa vzdáleného socketu, první kanál telemetrie, který se má připojit, vyhrává. Následné kanály, které se připojují na stejnou adresu, selhávají.

Pokud je na serveru více síťových adaptérů, rozdělte adresy vzdálených socketů mezi kanály telemetrie. Přidělování adres socketů je zcela libovolné, pokud je konkrétní adresa socketu konfigurována pouze na jednom kanálu telemetrie.

Nakonfigurujte IBM MQ pro připojení klientů MQTT pomocí průvodců uvedených v MQ Telemetry doplňku pro IBM MQ Explorer. Případně postupujte podle pokynů v části [Konfigurace správce front pro telemetrii v systémech Linux a AIX](#) a [Konfigurace správce front pro telemetrii v systému Windows](#) pro ruční konfiguraci telemetrie.

Související odkazy

[Vlastnosti MQXR](#)

Windows

Linux

AIX

Protokoly připojení telemetrie

Produkt MQ Telemetry podporuje protokoly TCP/IP IPv4 a IPv6a TLS.

Windows

Linux

AIX

Služba (MQXR) telemetrie

Služba telemetrie (MQXR) je modul listener TCP/IP, který je spravován jako služba IBM MQ . Vytvořte službu pomocí průvodce IBM MQ Explorer nebo pomocí příkazu `runmqsc` .

Služba MQ Telemetry (MQXR) se nazývá `SYSTEM.MQXR.SERVICE` .

Průvodce konfigurací ukázek telemetrie, který je k dispozici ve funkci MQ Telemetry for IBM MQ Explorer, vytvoří službu telemetrie a ukázkový kanál telemetrie; viz [Ověření instalace produktu MQ Telemetry pomocí IBM MQ Explorer](#) .

Vytvořte ukázkovou konfiguraci z příkazového řádku; viz [Ověření instalace produktu MQ Telemetry pomocí příkazového řádku](#).

Služba telemetrie (MQXR) se spouští a zastavuje automaticky se správcem front. Řízení služby pomocí složky služeb v souboru IBM MQ Explorer. Chcete-li zobrazit službu, musíte klepnutím na ikonu zastavit IBM MQ Explorer odfiltrování objektů `SYSTEM` ze zobrazení.

Příklad, jak vytvořit službu ručně, viz

- [Linux](#) [AIX](#) [Vytvoření SYSTEM.MQXR.SERVICE na Linux.](#)
- [Windows](#) [Vytvoření SYSTEM.MQXR.SERVICE na Windows.](#)

V 9.3.0 Od IBM MQ 9.3.0 dále, [Vytvoření SYSTEM.MQXR.SERVICE v systému Linuxa](#) [Vytvoření SYSTEM.MQXR.SERVICE v systému Windows](#) se aktualizují, aby uváděli výchozí klíč, který vyžaduje šifrování přístupových frází pro kanály MQTT TLS. Další informace naleznete v tématu [Šifrování přístupových frází pro MQTT kanály TLS](#).

Windows

Linux

AIX

Kanály telemetrie

Vytvořte kanály telemetrie pro vytvoření připojení s různými vlastnostmi, jako je například služba Java Authentication and Authorization Service (JAAS) nebo ověřování TLS, nebo pro správu skupin klientů.

Vytvořte kanály telemetrie pomocí průvodce **New Telemetry Channel** , který je součástí funkce MQ Telemetry pro IBM MQ Explorer. Pomocí průvodce nakonfigurujte kanál tak, aby přijímal připojení od klientů MQTT na konkrétním portu TCP/IP. Od IBM WebSphere MQ 7.1 můžete konfigurovat produkt MQ Telemetry pomocí programu příkazového řádku **runmqsc**.

Vytvořte více kanálů telemetrie na různých portech, abyste usnadnili správu velkého počtu připojení klienta rozdělením klientů do skupin. Každý kanál telemetrie má jiný název.

Můžete konfigurovat kanály telemetrie s různými atributy zabezpečení pro vytvoření různých typů připojení. Vytvořte více kanálů pro přijetí připojení klienta na různých adresách TCP/IP. Pomocí protokolu TLS můžete šifrovat zprávy a ověřit kanál a klienta telemetrie. Viz téma [Konfigurace protokolu TLS pro MQTT klienty a kanály telemetrie](#). Zadejte ID uživatele pro zjednodušení autorizace přístupu k objektům IBM MQ . Zadejte konfiguraci JAAS pro ověření uživatele MQTT pomocí služby JAAS; viz [MQTT Identifikace, autorizace a ověření klienta](#).

Windows

Linux

AIX

IBM MQ Telemetry Transport protokol

Protokol IBM MQ Telemetry Transport (MQTT) v3 je navržen pro výměnu zpráv mezi malými zařízeními s nízkou šířkou pásma nebo drahými připojeními a pro spolehlivé odesílání zpráv. Používá protokol TCP/IP.

Soubor MQTT protocol je publikován; viz [IBM MQ Telemetry Transport formát a protokol](#). Verze 3 protokolu používá publikování/odběr a podporuje tři kvality služby: *spustit a zapomenout*, *alespoň jednou* *právě jednou*.

Malá velikost záhlaví protokolu a informační obsah zprávy bajtového pole udržují zprávy malé. Záhlaví se skládají z pevné hlavičky o délce 2 bajty a až 12 bajtů z dalších proměnných hlaviček. Protokol používá 12bajtová záhlaví proměnných k odběru a připojení a pouze 2bajtová záhlaví proměnných pro většinu publikací.

Díky třem kvalitám služeb se můžete vyrovnat mezi nízkou latencí a spolehlivostí; viz [Kvality služeb poskytovaných MQTT klientem](#). Volba *Spustit a zapomenout* nepoužívá žádné trvalé úložiště zařízení a pouze jeden přenos k odeslání nebo přijetí publikace. *Alespoň jednou přesně jednou* vyžadují trvalé úložiště na zařízení, aby udrželo stav protokolu a uložilo zprávu, dokud nebude potvrzena.

Windows

Linux

AIX

MQTT klienti

Klientská aplikace MQTT je zodpovědná za shromažďování informací ze zařízení telemetrie, připojení k serveru a publikování informací na serveru. Může se také přihlásit k odběru témat, přijímat publikace a řídit zařízení telemetrie.

Na rozdíl od aplikací klienta IBM MQ nejsou aplikace klienta MQTT aplikacemi IBM MQ . Neurčují správce front, ke kterému se má připojit. Nejsou omezeny na použití specifických programovacích rozhraní IBM MQ . Namísto toho klienti MQTT implementují protokol MQTT 3 . Můžete napsat vlastní klientskou knihovnu do rozhraní MQTT protocol v programovacím jazyce a na zvolené platformě. Viz [IBM MQ Telemetry Transport formát a protokol](#).

Chcete-li zjednodušit zápis aplikací klienta MQTT , použijte knihovny klienta C, Javaa JavaScript , které zapouzdřují MQTT protocol pro řadu platform. Pokud tyto knihovny začleníte do svých aplikací MQTT , může být plně funkční klient produktu MQTT kratší než 15 řádků kódu. Knihovny klienta MQTT jsou volně dostupné z prostředí Eclipse Paho a MQTT.org. Viz [IBM MQ Telemetry Transport ukázkové programy](#).

Aplikace klienta MQTT je vždy zodpovědná za zahájení připojení pomocí kanálu telemetrie. Po připojení může aplikace klienta MQTT nebo aplikace IBM MQ zahájit výměnu zpráv.

Aplikace MQTT klienta a aplikace IBM MQ publikují a odebírají stejnou sadu témat. Aplikace IBM MQ může také odeslat zprávu přímo aplikaci klienta MQTT , aniž by aplikace klienta nejprve vytvořila odběr. Informace naleznete v tématu [Konfigurace distribuovaného řazení do front pro odesílání zpráv MQTT klientům](#).

Aplikace klienta MQTT jsou připojeny k produktu IBM MQ pomocí kanálu telemetrie. Kanál telemetrie funguje jako most mezi různými typy zpráv používaných produktem MQTT a produktem IBM MQ . Vytváří publikování a odběry ve správci front jménem aplikace klienta MQTT . Kanál telemetrie odesílá publikování, která odpovídají odběrům aplikace klienta MQTT ze správce front do aplikace klienta MQTT .

Aplikace IBM MQ mohou odesílat zprávy klientů MQTT v3 publikováním do odběrů vytvořených klienty nebo přímým odesláním zpráv. Klienti produktu MQTT mohou vzájemně odesílat zprávy publikováním témat, která odebírají jiní klienti.

Klient MQTT odebírá publikování, které obdrží od uživatele IBM MQ .

Provedte úlohu [“Publikování zprávy do obslužného programu klienta MQTT z adresáře IBM MQ Explorer”](#) na stránce 117 , chcete-li odeslat publikaci z produktu IBM MQ klientovi MQTT .

Standardním způsobem, jak může klient MQTT v3 přijímat zprávy, je vytvořit odběr tématu nebo sadu témat. V ukázkovém úseku kódu [Obrázek 44](#) na stránce 115 se klient MQTT přihlásí k odběru pomocí řetězce tématu "MQTT Examples". IBM MQ Aplikace v jazyce C [Obrázek 45](#) na stránce 116 publikuje do tématu pomocí řetězce tématu "MQTT Examples". V úseku kódu [Obrázek 46](#) na stránce 116 klient MQTT obdrží publikování v metodě zpětného volání `messageArrived`.

Další informace o tom, jak nakonfigurovat produkt IBM MQ tak, aby odesílal publikace v reakci na odběry z klientů systému MQTT , naleznete v tématu [Publikování zprávy v reakci na MQTT odběr klienta](#).

Aplikace IBM MQ odešle zprávu přímo klientovi MQTT .

Provedte úlohu [“Odeslání zprávy klientovi MQTT pomocí IBM MQ Explorer”](#) na stránce 121 , chcete-li odeslat zprávu přímo z produktu IBM MQ na klienta MQTT .

Zpráva odeslaná tímto způsobem klientovi MQTT se nazývá nevyžádaná zpráva. Klienti MQTT v3 přijímají nevyžádané zprávy jako publikace s nastaveným názvem tématu. Služba telemetrie (MQXR) nastaví název tématu na název vzdálené fronty.

Další informace o tom, jak nakonfigurovat produkt IBM MQ tak, aby odesílal zprávy přímo klientům MQTT , naleznete v tématu [Odeslání zprávy přímo klientovi](#).

Klient MQTT publikuje zprávu

Klient MQTT v3 může publikovat zprávu, která je přijata jiným klientem MQTT v3 , ale nemůže odeslat nevyžádanou zprávu. Úsek kódu [Obrázek 47](#) na stránce 116 ukazuje, jak klient MQTT v3 napsaný v souboru Javapublikuje zprávu.

Typickým vzorem pro odeslání zprávy jednomu konkrétnímu klientovi MQTT v3 je vytvoření odběru vlastního `ClientIdentifier` pro každého klienta. Provedte úlohu [“Publikování zprávy do specifického klienta MQTT v3”](#) na stránce 122 , abyste publikovali zprávu z jednoho klienta MQTT na jiného klienta MQTT pomocí `ClientIdentifier` jako řetězec tématu.

Příklady úseků kódu

Úsek kódu v souboru [Obrázek 44](#) na stránce 115 ukazuje, jak klient MQTT napsaný v produktu Java vytvoří odběr. Také potřebuje metodu zpětného volání `messageArrived` pro příjem publikování pro odběr.

```
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);
```

Obrázek 44. MQTT v3 odběratel klienta

Úsek kódu v souboru [Obrázek 45](#) na stránce 116 ukazuje, jak aplikace IBM MQ napsaná v jazyce C odešle publikaci. Úsek kódu je extrahován z úlohy [Vytvořit vydavatele do tématu proměnné](#).

```
/* 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);
```

Obrázek 45. IBM MQ vydavatel

Když publikování dorazí, klient MQTT zavolá metodu `messageArrived` třídy MQTT aplikačního klienta `MqttCallback`.

```
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
```

Obrázek 46. Metoda `messageArrived`

Obrázek 47 na stránce 116 zobrazuje MQTT v3 publikování zprávy v odběru vytvořeném v produktu [Obrázek 44](#) na stránce 115.

```
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);
```

Obrázek 47. MQTT v3 vydavatel klienta

klienta MQTT z adresáře IBM MQ Explorer

Chcete-li publikovat zprávu pomocí produktu IBM MQ Explorera přihlásit se k jejímu odběru pomocí obslužného programu klienta MQTT , postupujte podle kroků v této úloze. V další úloze je uveden způsob konfigurace aliasu správce front namísto nastavení výchozí přenosové fronty na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Než začnete

Úloha předpokládá, že jste obeznámeni s produktem IBM MQ a produktem IBM MQ Explorera že jsou nainstalovány funkce IBM MQ a MQ Telemetry .

Uživatel, který vytváří prostředky správce front pro tuto úlohu, musí mít k tomu dostatečná oprávnění. Pro demonstrační účely se předpokládá, že ID uživatele IBM MQ Explorer je členem skupiny `mqm` .

Informace o této úloze

V úloze vytvoříte téma v souboru IBM MQ a přihlásíte se k odběru tématu pomocí obslužného programu klienta MQTT . Když publikujete do tématu pomocí IBM MQ Explorer, klient MQTT obdrží publikování.

Postup

Proveďte jednu z následujících úloh:

- Nainstalovali jste produkt MQ Telemetry, ale ještě jste jej nespustili. Proveďte úlohu: [“Spustit úlohu bez dosud definované služby telemetrie \(MQXR\)”](#) na stránce 117.
- Dříve jste spustili telemetrii IBM MQ , ale chcete k demonstraci použít nového správce front. Proveďte úlohu: [“Spustit úlohu bez dosud definované služby telemetrie \(MQXR\)”](#) na stránce 117.
- Chcete provést úlohu pomocí existujícího správce front, který nemá definovány žádné prostředky telemetrie. Nechcete spustit průvodce **Definovat ukázkovou konfiguraci** .
 - a. Chcete-li nastavit telemetrii, proveďte jednu z následujících úloh:
 - [Konfigurace správce front pro telemetrii v systémech Linux a AIX](#)
 - [Konfigurace správce front pro telemetrii na systému Windows](#)
 - b. Proveďte úlohu: [“Spustit úlohu se spuštěnou službou telemetrie \(MQXR\)”](#) na stránce 118
- Chcete-li provést úlohu pomocí existujícího správce front, který již má definovány prostředky telemetrie, postupujte takto: [“Spustit úlohu se spuštěnou službou telemetrie \(MQXR\)”](#) na stránce 118.

Jak pokračovat dále

Chcete-li odeslat zprávu přímo obslužnému programu klienta, proveďte příkaz [“Odeslání zprávy klientovi MQTT pomocí IBM MQ Explorer”](#) na stránce 121 .

Spustit úlohu bez dosud definované služby telemetrie (MQXR)

Vytvořte správce front a spuštěním příkazu **Definovat ukázkovou konfiguraci** definujte ukázkové prostředky telemetrie pro správce front. Publikujte zprávu pomocí produktu IBM MQ Explorera přihlaste se k jejímu odběru pomocí obslužného programu klienta MQTT .

Informace o této úloze

Když nastavíte ukázkové prostředky telemetrie pomocí volby **Definovat ukázkovou konfiguraci**, průvodce nastaví oprávnění ID uživatele hosta. Pečlivě zvažte, zda chcete, aby bylo ID uživatele hosta tímto způsobem autorizováno. `guest` v systému Windowsa `nobody` v systému Linuxmají oprávnění publikovat a odebírat kořen stromu témat a vkládat zprávy do adresáře `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Průvodce také nastaví výchozí přenosovou frontu na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE`, což může kolidovat s aplikacemi spuštěnými v existujícím správci front. Je možné, ale pracné, konfigurovat

telemetrii a nepoužívat výchozí přenosovou frontu; proveďte následující úlohu: “Použití aliasu správce front” na stránce 119. V této úloze vytvoříte správce front, abyste se vyvarovali možnosti kolize s existujícími výchozími přenosovými frontami.

Postup

1. Pomocí produktu IBM MQ Explorer vytvořte a spusťte nového správce front.
 - a) Klepněte pravým tlačítkem myši na složku Queue Managers > **Nová** > **Správce front ...**. Zadejte název správce front > **Dokončit**.
Vytvořte název správce front, například MQTTQMGR.
2. Vytvořte a spusťte službu telemetrie (MQXR) a vytvořte ukázkový kanál telemetrie.
 - a) Otevřete složku Queue Managers\QmgrName\Telemetry .
 - b) Klepněte na volbu **Definovat ukázkovou konfiguraci ...** > **Dokončit**
Ponechte zaškrťovací políčko **Spustit MQTT Obslužný program klienta** zaškrtnuté.
3. Vytvořte odběr pro produkt MQTT Example pomocí obslužného programu klienta MQTT .
 - a) Klepněte na tlačítko **Připojit**.
Historie klienta zaznamenává událost Connected .
 - b) Zadejte MQTT Example do pole **Odběr \ téma** > **Odebírat**.
Historie klienta zaznamenává událost Subscribed .
4. Vytvořte soubor MQTTExampleTopic v adresáři IBM MQ.
 - a) Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Topics v Průzkumníku **MQ** > **Nový** > **Téma**.
 - b) Zadejte MQTTExampleTopic jako **Název** > **Další**.
 - c) Zadejte MQTT Example jako **Řetězec tématu** > **Dokončit**.
 - d) Klepnutím na tlačítko **OK** zavřete okno potvrzení.
5. Publikujte Hello World! do tématu MQTT Example pomocí IBM MQ Explorer.
 - a) Klepněte na složku Queue Managers\QmgrName\Topics v adresáři IBM MQ Explorer.
 - b) Klepněte pravým tlačítkem myši na volbu MQTTExampleTopic > **Testovat publikování ...**
 - c) Zadejte Hello World! do pole **Data zprávy** > **Publikovat zprávu** > Přepněte do okna obslužného programu klienta MQTT .

Historie klienta zaznamenává událost Received .

Spustit úlohu se spuštěnou službou telemetrie (MQXR)

Vytvořte kanál telemetrie a téma. Autorizujte uživatele k použití tématu a přenosové fronty telemetrie. Publikujte zprávu pomocí produktu IBM MQ Explorera přihlaste se k jejímu odběru pomocí obslužného programu klienta MQTT .

Než začnete

V této verzi úlohy je definován a spuštěn správce front *QmgrName*. Služba telemetrie (MQXR) je definována a spuštěna. Služba telemetrie (MQXR) mohla být vytvořena ručně nebo spuštěním průvodce **Definovat ukázkovou konfiguraci** .

Informace o této úloze

V této úloze nakonfigurujete existujícího správce front tak, aby odeslal publikování obslužnému programu klienta MQTT .

Krok “1” na stránce 119 úlohy nastaví výchozí přenosovou frontu na SYSTEM.MQTT.TRANSMIT.QUEUE, což může kolidovat s aplikacemi spuštěnými v existujícím správcí front. Je možné, ale pracné,

konfigurovat telemetrii a nepoužívat výchozí přenosovou frontu; proveďte následující úlohu: “Použití aliasu správce front” na stránce 119.

Postup

- Nastavte SYSTEM.MQTT.TRANSMIT.QUEUE jako výchozí přenosovou frontu.
 - Klepněte pravým tlačítkem myši na Queue Managers\QmgrName folder > **Vlastnosti ...**
 - V navigátoru klepněte na volbu **Komunikace**.
 - Klepněte na volbu **Vybrat ...** > Vybrat SYSTEM.MQTT.TRANSMIT.QUEUE > **OK** > **OK**.
- Vytvořte kanál telemetrie MQTTExampleChannel pro připojení obslužného programu klienta MQTT k produktu IBM MQa spusťte obslužný program klienta MQTT.
 - Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Telemetry\Channels v Průzkumníku **MQ** > **Nový** > **Kanál telemetrie ...**
 - Zadejte MQTTExampleChannel do pole **Název kanálu** > **Další** > **Další**.
 - Změňte **Pevné ID uživatele** na panelu autorizace klienta na ID uživatele, které se chystá publikovat a přihlásit se k odběru MQTTExample > **Další**.
 - Nechte políčko **Spustit obslužný program klienta** zaškrtnuté > **Dokončit**.
- Vytvořte odběr pro produkt MQTT Example pomocí obslužného programu klienta MQTT.
 - Klepněte na tlačítko **Připojit**.

Historie klienta zaznamenává událost Connected.
 - Zadejte MQTT Example do pole **Odběr \ téma** > **Odebírat**.

Historie klienta zaznamenává událost Subscribed.
- Vytvořte soubor MQTTExampleTopic v adresáři IBM MQ.
 - Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Topics v Průzkumníku **MQ** > **Nový** > **Téma**.
 - Zadejte MQTTExampleTopic jako **Název** > **Další**.
 - Zadejte MQTT Example jako **Řetězec tématu** > **Dokončit**.
 - Klepnutím na tlačítko **OK** zavřete okno potvrzení.
- Chcete-li, aby uživatel, který není ve skupině mqm, publikoval téma MQTTExample a přihlásili se k jeho odběru, postupujte takto:
 - Autorizujte uživatele k publikování a odběru tématu MQTTExampleTopic:

```
setmqaut -m qMgrName -t topic -n MQTTExampleTopic -p User ID -all +pub +sub
```
 - Autorizujte uživatele, aby vložil zprávu do souboru SYSTEM.MQTT.TRANSMIT.QUEUE:

```
setmqaut -m qMgrName -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p User ID -all +put
```
- Publikujte Hello World! do tématu MQTT Example pomocí IBM MQ Explorer.
 - Klepněte na složku Queue Managers\QmgrName\Topics v adresáři IBM MQ Explorer.
 - Klepněte pravým tlačítkem myši na volbu MQTTExampleTopic > **Testovat publikování ...**
 - Zadejte Hello World! do pole **Data zprávy** > **Publikovat zprávu** > Přepněte do okna obslužného programu klienta MQTT.

Historie klienta zaznamenává událost Received.

Použití aliasu správce front

Publikujte zprávu do obslužného programu klienta MQTT pomocí IBM MQ Explorer bez nastavení výchozí přenosové fronty na SYSTEM.MQTT.TRANSMIT.QUEUE.

Tato úloha je pokračováním předchozí úlohy a používá alias správce front, aby se zabránilo nastavení výchozí přenosové fronty na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Než začnete

Dokončete buď úlohu “Spustit úlohu bez dosud definované služby telemetrie (MQXR)” na stránce 117, nebo úlohu “Spustit úlohu se spuštěnou službou telemetrie (MQXR)” na stránce 118.

Informace o této úloze

Když klient MQTT vytvoří odběr, produkt IBM MQ odešle svou odezvu pomocí `ClientIdentifier` jako název vzdáleného správce front. V této úloze používá `ClientIdentifier`, `MyClient`.

Pokud neexistuje žádná přenosová fronta nebo alias správce front s názvem `MyClient`, odezva se umístí do výchozí přenosové fronty. Nastavením výchozí přenosové fronty na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE` klient MQTT získá odezvu.

Nastavením výchozí přenosové fronty na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE` se můžete vyhnout použitím aliasů správce front. Musíte nastavit alias správce front pro každý `ClientIdentifier`. Obvykle existuje příliš mnoho klientů, aby bylo praktické používat aliasy správce front. Často je `ClientIdentifier` nepředvídatelný, což znemožňuje konfigurovat telemetrii tímto způsobem.

V některých případech však může být nutné nakonfigurovat výchozí přenosovou frontu na jinou hodnotu než `SYSTEM.MQTT.TRANSMIT.QUEUE`. Kroky v části Procedura konfigurují alias správce front namísto nastavení výchozí přenosové fronty na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Postup

- Odeberte `SYSTEM.MQTT.TRANSMIT.QUEUE` jako výchozí přenosovou frontu.
 - Klepněte pravým tlačítkem myši na `Queue Managers\QmgrName` folder > **Vlastnosti ...**
 - V navigátoru klepněte na volbu **Komunikace**.
 - Odeberte soubor `SYSTEM.MQTT.TRANSMIT.QUEUE` z pole **Výchozí přenosová fronta** > **OK**.
- Zkontrolujte, zda již nelze vytvořit odběr pomocí obslužného programu klienta MQTT:
 - Klepněte na tlačítko **Připojit**.
Historie klienta zaznamenává událost `Connected`.
 - Zadejte `MQTT Example` do pole **Odběr \ téma** > **Odebírat**.
Historie klienta zaznamenává událost `Subscribe failed` a událost `Connection lost`.
- Vytvořte alias správce front pro `ClientIdentifier`, `MyClient`.
 - Klepněte pravým tlačítkem myši na složku `Queue Managers\QmgrName\Queues` > **Nová > Definice vzdálené fronty**.
 - Pojmenujte definici `MyClient` > **Další**.
 - Zadejte `MyClient` do pole **Vzdálený správce front**.
 - Zadejte `SYSTEM.MQTT.TRANSMIT.QUEUE` do pole **Přenosová fronta** > **Dokončit**.
- Znovu připojte obslužný program klienta MQTT.
 - Zkontrolujte, zda je **Identifikátor klienta** nastaven na `MyClient`.
 - Připojit**
Historie klienta zaznamenává událost `Connected`.
- Vytvořte odběr pro produkt `MQTT Example` pomocí obslužného programu klienta MQTT.
 - Klepněte na tlačítko **Připojit**.
Historie klienta zaznamenává událost `Connected`.
 - Zadejte `MQTT Example` do pole **Odběr \ téma** > **Odebírat**.

Historie klienta zaznamenává událost `Subscribed` .

6. Publikujte `Hello World!` do tématu `MQTT Example` pomocí `IBM MQ Explorer`.

- a) Klepněte na složku `Queue Managers\QmgrName\Topics` v adresáři `IBM MQ Explorer`.
- b) Klepněte pravým tlačítkem myši na volbu `MQTTExampleTopic > Testovat publikování ...`
- c) Zadejte `Hello World!` do pole **Data zprávy > Publikovat zprávu > Přepněte do okna obslužného programu klienta MQTT** .

Historie klienta zaznamenává událost `Received` .

Windows Linux AIX Odeslání zprávy klientovi MQTT pomocí IBM MQ Explorer

Odešlete zprávu obslužnému programu klienta MQTT vložení zprávy do fronty IBM MQ pomocí `IBM MQ Explorer`. Tato úloha ukazuje, jak nakonfigurovat definici vzdálené fronty tak, aby odesílala zprávu přímo klientovi MQTT .

Než začnete

Provedte úlohu “[Publikování zprávy do obslužného programu klienta MQTT z adresáře IBM MQ Explorer](#)” na stránce 117. Ponechte obslužný program klienta MQTT připojený.

Informace o této úloze

Tato úloha demonstruje odeslání zprávy klientovi MQTT s použitím fronty namísto publikování do tématu. Odběr v klientu nevytváříte. Krok “2” na stránce 121 úlohy ukazuje, že předchozí odběr byl odstraněn.

Postup

1. Zrušte všechny existující odběry odpojením a opětovným připojením obslužného programu klienta MQTT .

Odběr je zrušen, protože pokud nezměníte výchozí nastavení, obslužný program klienta MQTT se připojí k čisté relaci; viz [Vyčistit relace](#).

Chcete-li usnadnit provádění úlohy, zadejte vlastní `ClientIdentifier` nepoužívejte generovaný `ClientIdentifier` vytvořený obslužným programem klienta MQTT .

- a) Klepnutím na volbu **Odpojit** odpojte obslužný program klienta MQTT od kanálu telemetrie.

Historie klienta zaznamenává událost `Disconnected` .

- b) Změňte **Identifikátor klienta** na `MyClient`.

- c) Klepněte na tlačítko **Připojit**.

Historie klienta zaznamenává událost `Connected` .

2. Zkontrolujte, zda obslužný program klienta MQTT již nepřijímá publikování pro produkt `MQTTExampleTopic`.

- a) Klepněte na složku `Queue Managers\QmgrName\Topics` v adresáři `IBM MQ Explorer`.
- b) Klepněte pravým tlačítkem myši na volbu `MQTTExampleTopic > Testovat publikování ...`
- c) Zadejte `Hello World!` do pole **Data zprávy > Publikovat zprávu > Přepněte do okna obslužného programu klienta MQTT** .

Do **Historie klientanení** zaznamenána žádná událost.

3. Vytvořte definici vzdálené fronty pro klienta.

Jako název vzdáleného správce front v definici vzdálené fronty nastavte `ClientIdentifier`, `MyClient`. Jako název vzdálené fronty použijte libovolný název, který se vám líbí. Název vzdálené fronty je předán klientovi MQTT jako název tématu.

- a) Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Queues > **Nová** > **Definice vzdálené fronty**.
 - b) Pojmenujte definici MyClientRemoteQueue > **Další**.
 - c) Zadejte MQTTExampleQueue do pole **Vzdálená fronta**.
 - d) Zadejte MyClient do pole **Vzdálený správce front**.
 - e) Zadejte SYSTEM.MQTT.TRANSMIT.QUEUE do pole **Přenosová fronta** > **Dokončit**.
4. Vložte testovací zprávu do souboru MyClientRemoteQueue.
- a) Klepněte pravým tlačítkem myši na položku **MyClientRemoteQueue** > **Vložit testovací zprávu ...**
 - b) Zadejte Hello queue! do pole Data zprávy > **Vložit zprávu** > **Zavřít**
- Historie klienta** zaznamenává událost Received.
5. Odeberte SYSTEM.MQTT.TRANSMIT.QUEUE jako výchozí přenosovou frontu.
- a) Klepněte pravým tlačítkem myši na Queue Managers\QmgrName folder > **Vlastnosti ...**
 - b) V navigátoru klepněte na volbu **Komunikace**.
 - c) Odeberte soubor SYSTEM.MQTT.TRANSMIT.QUEUE z pole **Výchozí přenosová fronta** > **OK**.
6. Krok pro zopakování “4” na stránce 122.

MyClientRemoteQueue je definice vzdálené fronty, která explicitně pojmenuje přenosovou frontu. Nemusíte definovat výchozí přenosovou frontu pro odeslání zprávy do MyClient.

Jak pokračovat dále

Vzhledem k tomu, že výchozí přenosová fronta již není nastavena na hodnotu SYSTEM.MQTT.TRANSMIT.QUEUE, obslužný program klienta MQTT nemůže vytvořit nový odběr, pokud není definován alias správce front pro ClientIdentifier, MyClient. Obnovte výchozí přenosovou frontu na SYSTEM.MQTT.TRANSMIT.QUEUE.

Windows Linux AIX Publikování zprávy do specifického klienta

MQTT v3

Publikujte zprávu z jednoho klienta MQTT v3 do jiného pomocí ClientIdentifier jako název tématu a IBM MQ jako zprostředkovatele publikování/odběru.

Než začnete

Provedte úlohu “[Publikování zprávy do obslužného programu klienta MQTT z adresáře IBM MQ Explorer](#)” na stránce 117. Ponechte obslužný program klienta MQTT připojený.

Informace o této úloze

Úloha demonstruje dvě věci:

1. Přihlášení k odběru tématu v jednom klientu MQTT a přijetí publikace od jiného klienta MQTT.
2. Nastavení odběrů typu "point-to-point" pomocí ClientIdentifier jako řetězce tématu.

Postup

1. Zrušte všechny existující odběry odpojením a opětovným připojením obslužného programu klienta MQTT.

Odběr je zrušen, protože pokud nezměníte výchozí nastavení, obslužný program klienta MQTT se připojí k čisté relaci; viz [Vyčistit relace](#).

Chcete-li usnadnit provádění úlohy, zadejte vlastní ClientIdentifiera nepoužívejte generovaný ClientIdentifier vytvořený obslužným programem klienta MQTT.

- a) Klepnutím na volbu **Odpojit** odpojte obslužný program klienta MQTT od kanálu telemetrie.

Historie klienta zaznamenává událost `Disconnected` .

- b) Změňte **Identifikátor klienta** na `MyClient`.
- c) Klepněte na tlačítko **Připojit**.

Historie klienta zaznamenává událost `Connected` .

2. Vytvořte odběr tématu `MyClient` .

`MyClient` je `ClientIdentifier` tohoto klienta.

- a) Zadejte `MyClient` do pole **Odběr \ téma > Odebírat**.

Historie klienta zaznamenává událost `Subscribed` .

3. Spusťte jiný obslužný program klienta MQTT .

- a) Otevřete složku `Queue Managers\QmgrName\Telemetry\channels` .
- b) Klepněte pravým tlačítkem myši na kanál **PlainText > Spustit MQTT obslužný program klienta ...**
- c) Klepněte na tlačítko **Připojit**.

Historie klienta zaznamenává událost `Connected` .

4. Publikujte `Hello MyClient!` do tématu `MyClient`.

- a) Zkopírujte téma odběru `MyClient`z obslužného programu klienta MQTT spuštěného s identifikátorem klienta `ClientIdentifier MyClient`.
- b) Vložte soubor `MyClient` do pole **Publikace \ Téma** každé z instancí obslužného programu klienta MQTT .
- c) Zadejte `Hello MyClient!` do pole **Publikování \ zpráva** .
- d) Klepněte na volbu **Publikovat** v obou instancích.

Výsledky

Historie klienta v obslužném programu klienta MQTT s identifikátorem klienta `ClientIdentifier, MyClient`, zaznamenává dvě **přijaté** události a jednu událost **Publikováno** . Druhá instance obslužného programu klienta MQTT zaznamenává jednu událost **Publikováno** .

Pokud vidíte pouze jednu událost **Přijato** , zkontrolujte následující možné příčiny:

1. Je výchozí přenosová fronta pro správce front nastavena na hodnotu `SYSTEM.MQTT.TRANSMIT.QUEUE` ?
2. Vytvořili jste aliasy správce front nebo definice vzdálených front odkazující na produkt `MyClient` při provádění dalších cvičení? V případě problému s konfigurací odstraňte všechny prostředky, které odkazují na produkt `MyClient`, například aliasy správce front nebo přenosové fronty. Odpojte obslužné programy klienta, zastavte a restartujte službu telemetrie (MQXR).

Windows

Linux

AIX

Odeslání zprávy aplikaci IBM MQ z klienta MQTT

Aplikace IBM MQ může přijmout zprávu od klienta MQTT v3 přihlášením k odběru tématu. Klient MQTT se připojí k produktu IBM MQ pomocí kanálu telemetrie a odešle zprávu do aplikace IBM MQ publikováním do stejného tématu.

Proveďte úlohu “[Publikování zprávy do produktu IBM MQ z klienta systému MQTT](#)” na stránce 123, abyste se naučili, jak odeslat publikaci z klienta MQTT na odběr definovaný v produktu IBM MQ.

Je-li téma klastrováno nebo distribuováno pomocí hierarchie publikování/odběru, může se odběr nacházet v jiném správci front, než je správce front, ke kterému je klient MQTT připojen.

Windows

Linux

AIX

Publikování zprávy do produktu IBM MQ z klienta systému MQTT

Vytvořte odběr tématu pomocí produktu IBM MQ Explorer a publikujte do tématu pomocí obslužného programu klienta MQTT .

Než začnete

Provedte úlohu “Publikování zprávy do obslužného programu klienta MQTT z adresáře IBM MQ Explorer” na stránce 117. Ponechte obslužný program klienta MQTT připojený.

Informace o této úloze

Tato úloha demonstruje publikování zprávy s klientem MQTT a příjem publikování s použitím nespravovaného trvalého odběru vytvořeného pomocí produktu IBM MQ Explorer.

Postup

1. Vytvořte trvalý odběr řetězce tématu MQTT Example.

Chcete-li vytvořit frontu a odběr pomocí produktu IBM MQ Explorer, postupujte takto.

- a) Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Queues v nabídce IBM MQ Explorer > **Nový** > **Lokální fronta ...**
- b) Zadejte MQTTExampleQueue jako název fronty > **Dokončit**.
- c) Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Subscriptions v nabídce IBM MQ Explorer > **Nový** > **Odběr ...**
- d) Zadejte MQTTExampleSubscription jako název fronty > **Další**.
- e) Klepněte na volbu **Vybrat ...** > MQTTExampleTopic > **OK**.

Téma MQTTExampleTopic jste již vytvořili v kroku “4” na stránce 118 “Publikování zprávy do obslužného programu klienta MQTT z adresáře IBM MQ Explorer” na stránce 117.

- f) Zadejte MQTTExampleQueue jako název místa určení > **Dokončit**.

2. Jako volitelný krok nastavte frontu pro použití jiným uživatelem bez oprávnění mqm .

Pokud nastavujete konfiguraci pro uživatele s menším oprávněním než mqm, musíte udělit oprávnění put a get pro MQTTExampleQueue. Přístup k tématu a k přenosové frontě byl nakonfigurován v souboru “Publikování zprávy do obslužného programu klienta MQTT z adresáře IBM MQ Explorer” na stránce 117.

- a) Autorizujte uživatele, aby vložil a dostal se do fronty MQTTExampleQueue:

```
setmqaut -m qMgrName -t queue -n MQTTExampleQueue -p User ID -all +put +get
```

3. Publikujte soubor Hello IBM MQ! do tématu MQTT Example pomocí obslužného programu klienta MQTT .

Pokud jste obslužný program klienta MQTT neopustili, klepněte pravým tlačítkem myši na kanál **PlainText** > **Spustit MQTT obslužný program klienta ...** > **Připojit**.

- a) Zadejte MQTT Example do pole **Publikování \ Téma** .
- b) Zadejte Hello IBM MQ! do pole **Publikace \ Zpráva** > **Publikovat**.

4. Otevřete složku Queue Managers\QmgrName\Queues a vyhledejte položku MQTTExampleQueue.

Pole **Aktuální hloubka fronty** je 1 .

5. Klepněte pravým tlačítkem myši na MQTTExampleQueue > **Procházet zprávy ...** a prozkoumat publikaci.

Windows

Linux

AIX

MQTT publikování/odběr aplikací

K zápisu aplikací MQTT použijte publikování/odběr založené na tématu.

Když je klient MQTT připojen, publikace se pohybují v obou směrech mezi klientem a serverem. Publikování jsou odesílána z klienta při publikování informací v klientu. Publikování jsou přijímána v klientu při publikování zprávy do tématu, které odpovídá odběru vytvořenému klientem.

Zprostředkovatel publikování/odběru IBM MQ spravuje témata a odběry vytvořené klienty MQTT . Témata vytvořená klienty MQTT sdílí stejný prostor tématu jako témata vytvořená aplikacemi IBM MQ .

Publikování, která odpovídají řetězci tématu v odběru klienta MQTT, jsou umístěna v systému SYSTEM.MQTT.TRANSMIT.QUEUE s názvem vzdáleného správce front nastaveným na hodnotu ClientIdentifier klienta. Služba telemetrie (MQXR) předává publikace klientovi, který vytvořil odběr. Používá ClientIdentifier, který byl nastaven jako název vzdáleného správce front pro identifikaci klienta.

Obvykle musí být SYSTEM.MQTT.TRANSMIT.QUEUE definováno jako výchozí přenosová fronta. Je možné, ale obtížné, nakonfigurovat produkt MQTT tak, aby nepoužíval výchozí přenosovou frontu. Viz téma [Konfigurovat distribuované fronty pro odesílání zpráv klientům MQTT](#).

Klient MQTT může vytvořit trvalou relaci; viz [“MQTT bezstavové a stavové relace”](#) na stránce 128. Odběry vytvořené v trvalé relaci jsou trvalé. Publikace, které přicházejí pro klienta s trvalou relací, jsou uloženy v adresáři SYSTEM.MQTT.TRANSMIT.QUEUEa předány klientovi při opětovném připojení.

Klient systému MQTT může také publikovat a odebírat zachovaná publikování; viz [Zachovaná publikování a MQTT klienti](#). Odběratel zachovaného tématu publikování obdrží nejnovější publikování do tématu. Odběratel obdrží zachovaná publikování při vytvoření odběru nebo při opětovném připojení k předchozí relaci.

Windows

Linux

AIX

Aplikace telemetrie

Zapíšete aplikace telemetrie pomocí toků zpráv IBM MQ nebo IBM Integration Bus .

Pomocí rozhraní JMS, MQI nebo jiných programovacích rozhraní IBM MQ můžete programovat aplikace telemetrie v produktu IBM MQ.

Služba telemetrie (MQXR) se převádí mezi zprávami MQTT v3 a IBM MQ . Vytváří odběry a publikace jménem klientů systému MQTT a předává publikace klientům systému MQTT . Publikace je informační obsah zprávy MQTT v3 . Informační obsah se skládá ze záhlaví zpráv a bajtového pole ve formátu `jms-bytes` . Server telemetrie mapuje záhlaví mezi zprávou MQTT v3 a zprávou IBM MQ ; viz [“Integrace produktu MQ Telemetry se správcí front”](#) na stránce 125.

Pomocí uzlů Publikace, MQInput a JMSInput můžete odesílat a přijímat publikování mezi klienty IBM Integration Bus a MQTT .

Pomocí toků zpráv můžete telemetrii integrovat s webovými servery pomocí protokolu HTTPa s dalšími aplikacemi používajícími adaptéry IBM MQ a WebSphere .

Windows

Linux

AIX

Integrace produktu MQ Telemetry se správcí front

front

Klient MQTT je integrován s produktem IBM MQ jako aplikace publikování/odběru. Může buď publikovat, nebo se přihlásit k odběru témat v produktu IBM MQ, vytvářet nová témata, nebo používat existující témata. Přijímá publikace z produktu IBM MQ v důsledku toho, že klienti systému MQTT včetně sebe nebo jiných aplikací IBM MQ publikují do témat svých odběrů. Pravidla se používají k rozhodování o atributech publikování.

Mnoho atributů přidružených k tématům, publikacím, odběrům a zprávám poskytovaným produktem IBM MQnení podporováno. [“MQTT klient pro IBM MQ zprostředkovatele publikování/odběru”](#) na stránce 126 a [“IBM MQ do klienta MQTT”](#) na stránce 127 popisují, jak jsou nastaveny atributy publikování. Nastavení závisí na tom, zda publikování probíhá ve zprostředkovateli publikování/odběru produktu IBM MQ nebo od něj.

V produktu IBM MQ jsou témata publikování/odběru přidružena k objektům administrativních témat. Témata vytvořená klienty MQTT se nijak neliší. Když klient MQTT vytvoří řetězec tématu pro publikování, zprostředkovatel publikování/odběru produktu IBM MQ jej přidruží k objektu administrativního tématu. Zprostředkovatel mapuje řetězec tématu v publikování na nejbližší nadřazený objekt administrativního tématu. Mapování je stejné jako pro aplikace IBM MQ . Pokud neexistuje žádné uživatelem vytvořené téma, téma publikování se namapuje na SYSTEM.BASE.TOPIC. Atributy použité na publikování jsou odvozeny z objektu tématu.

Když aplikace IBM MQ nebo administrátor vytvoří odběr, bude tento odběr pojmenován. Vypište odběry pomocí produktu IBM MQ Explorernebo pomocí příkazů **runmqsc** či PCF. Všechny odběry klienta MQTT mají název. Dostanou název formuláře: *ClientIdentifier:Topic name*

MQTT klient pro IBM MQ zprostředkovatele publikování/odběru

Klient MQTT odeslal publikaci na adresu IBM MQ. Služba telemetrie (MQXR) převede publikaci na zprávu IBM MQ . Zpráva IBM MQ obsahuje tři části:

1. MQMD
2. RFH2
3. Zpráva

Vlastnosti MQMD jsou nastaveny na výchozí hodnoty, s výjimkou hodnot uvedených v části [Tabulka 9](#) na stránce 126.

<i>Tabulka 9. Nastavení pro vlastnosti MQMD</i>		
Pole MQMD	Typ	Hodnota
Format	MQCHAR8	MQFMT_RF_HEADER_2
UserIdentifier	MQCHAR12	Nastavit na jednu z následujících možností: MqttClient.ClientIdentifier MqttConnectOptions.UserName ID uživatele nastavené administrátorem produktu IBM MQ pro kanál telemetrie.
Priority	MQLONG	MQPRI_PRIORITY_AS_Q_DEF (Výchozí hodnota pro IBM MQ, která se liší od hodnoty JMS , která má výchozí hodnotu 4.)
Persistence	MQLONG	QoS=0→MQPER_NOT_PERSISTENT QoS=1→MQPER_PERSISTENT QoS=2→MQPER_PERSISTENT

Záhlaví RFH2 neobsahuje složku <msd> pro definování typu zprávy JMS . Služba telemetrie (MQXR) vytvoří zprávu IBM MQ jako výchozí zprávu JMS . Výchozí typ zprávy JMS je zpráva jms-bytes . Aplikace může přistupovat k dalším informacím záhlaví jako k vlastnostem zprávy; viz [Vlastnosti zprávy](#).

RFH2 hodnoty jsou nastaveny tak, jak ukazuje [Tabulka 10](#) na stránce 126. Vlastnost Formát je nastavena v pevném záhlaví RFH2 a ostatní hodnoty jsou nastaveny ve složkách RFH2 .

<i>Tabulka 10. Nastavení pro vlastnosti RFH2</i>		
RFH2 vlastnost	Typ/složka	Header
Formát	MQCHAR8	MQFMT_NONE
ClientIdentifier	mqtt/clientId	Zkopírujte MqttClient.ClientIdentifier s délkou 1...23 bajtů.
QoS	mqtt/qos	Zkopírujte QoS z příchozí MQTT zprávy.
ID zprávy	mqtt/msgid	Zkopírujte ID zprávy z příchozí zprávy MQTT , pokud QoS je 1 nebo 2.

Tabulka 10. Nastavení pro vlastnosti RFH2 (pokračování)

RFH2 vlastnost	Typ/složka	Header
MQIsRetained	mqps/Ret	Nastavte, pokud bylo původní publikování MQTT odesláno se sadou vlastností RETAIN a zpráva je přijata jako zachované publikování.
MQTopicString	mqps/Top	Téma, do kterého byla zpráva MQTT publikována.

Informační obsah v publikaci MQTT je mapován na obsah zprávy IBM MQ :

Tabulka 11. Jak se informační obsah v publikaci MQTT mapuje na obsah zprávy IBM MQ

Obsah zprávy	Typ	Obsah zprávy IBM MQ
Vyrovňovací paměť	MQBYTE <i>n</i>	Kopie bajtů z příchozí zprávy MQTT . Délka může být nula.

IBM MQ do klienta MQTT

Klient se přihlásil k odběru tématu publikování. Do tématu byla publikována aplikace IBM MQ , což má za následek odeslání publikování odběrateli produktu MQTT zprostředkovatelem publikování/odběru produktu IBM MQ . Alternativně aplikace IBM MQ odeslala nevyžádanou zprávu přímo klientovi MQTT . Tabulka 12 na stránce 127 popisuje, jak jsou pevná záhlaví zpráv nastavena ve zprávě, která je odeslána klientovi MQTT . Veškerá další data v záhlaví zprávy IBM MQ nebo v jiných záhlavích jsou vyřazena. Data zprávy ve zprávě IBM MQ se odešlou jako informační obsah zprávy ve zprávě MQTT bez změny. Zpráva MQTT je odeslána klientovi MQTT službou telemetrie (MQXR).

Tabulka 12. Jak jsou nastavena pevná záhlaví zpráv ve zprávě IBM MQ odeslané klientovi MQTT

MQTT pole	Typ	Hodnota
DUP	typ boolean	Nastavte, zda QoS = 1 nebo 2a zpráva byla odeslána tomuto klientovi v předchozím přenosu a zpráva nebyla po čase potvrzena.
QoS	celé číslo	Způsob nastavení hodnoty QoS v odchozím publikování ze zprostředkovatele publikování/odběru v produktu IBM MQ závisí na příchozím publikování. Závisí na tom, zda bylo příchozí publikování odesláno z klienta MQTT nebo z aplikace IBM MQ . MQTT Nižší hodnota QoS v příchozí publikaci a v QoS požadované odběratelem. IBM MQ Nižší hodnota QoS odvozená z příchozího publikování: MQPER_NOT_PERSISTENT→QoS=0 MQPER_PERSISTENT→QoS=2 a QoS požadované odběratelem. Pokud je zpráva odeslána klientovi bez odběru, QoS je standardně nastaven na 2. Klient může tuto hodnotu změnit tak, že se přihlásí k odběru produktu DEFAULT . QoS s jinou QoS.
RETAIN	typ boolean	Nastavte, pokud má příchozí publikování nastavenou zachovanou vlastnost.

Tabulka 13 na stránce 128 popisuje, jak jsou nastavena záhlaví proměnných zpráv ve zprávě MQTT, která je odeslána klientovi MQTT.

Tabulka 13. Jak jsou nastaveny vlastnosti záhlaví proměnné MQTT ve zprávě MQTT odeslané klientovi MQTT		
MQTT pole	Typ	Hodnota
Topic name	Řetězec	Řetězec tématu, se který byla zpráva publikována.
Message ID	Řetězec	Poslední 2 bajty deskriptoru MQMD produktu MQMD. MsgId publikování, když je umístěno do SYSTEM.MQTT.TRANSMIT.QUEUE.
Payload	byte[]	Přímá kopie bajtů z příchozího publikování do zprostředkovatele publikování/odběru. Délka může být nula.

Windows

Linux

AIX

MQTT bezstavové a stavové relace

Klienti MQTT mohou vytvořit stavovou relaci se správcem front. Když se stavový klient MQTT odpojí, správce front udržuje odběry vytvořené klientem a probíhající zprávy. Když se klient znovu připojí, vyřeší probíhající zprávu. Odesílá všechny zprávy, které jsou zařazeny do fronty k doručení, a přijímá všechny zprávy publikované pro své odběry v době, kdy byly odpojeny.

Když se klient MQTT připojí ke kanálu telemetrie, buď spustí novou relaci, nebo obnoví starou relaci. Nová relace nemá žádné nevyřízené zprávy, které nebyly potvrzeny, žádné odběry a žádná publikování čekající na doručení. Když se klient připojí, uvádí, zda se má spustit s čistou relací, nebo zda se má pokračovat v existující relaci; viz [Vyčistit relace](#).

Pokud klient obnoví existující relaci, bude pokračovat, jako by připojení nebylo přerušeno. Publikování čekající na doručení jsou odeslána klientovi a všechny přenosy zpráv, které nebyly potvrzeny, jsou dokončeny. Když se klient v trvalé relaci odpojí od služby telemetrie (MQXR), všechny odběry, které klient vytvořil, zůstanou zachovány. Publikování pro odběry jsou klientovi odesílána při opětovném připojení. Pokud se znovu připojí bez obnovení staré relace, jsou publikování službou telemetrie (MQXR) zrušena.

Informace o stavu relace jsou uloženy správcem front ve frontě SYSTEM.MQTT.PERSISTENT.STATE.

Administrátor produktu IBM MQ může odpojit a vyprázdnit relaci.

Windows

Linux

AIX

Když není klient MQTT připojen

Není-li klient připojen, může správce front nadále přijímat publikování jeho jménem. Jsou postoupeny klientovi při opětovném připojení. Klient může vytvořit "Poslední vůle a závěť", kterou správce front publikuje jménem klienta, pokud se klient neočekávaně odpojí.

Chcete-li být upozorněni, když se klient neočekávaně odpojí, můžete zaregistrovat poslední publikaci závěti a testamentu; viz [Poslední publikace závěti a testamentu](#). Je odeslána službou telemetrie (MQXR), pokud zjistí, že připojení ke klientovi bylo přerušeno, aniž by to klient požadoval.

Klient může kdykoli publikovat zachované publikování; viz [Zachovaná publikování a MQTT klienti](#). Nový odběr tématu může požadovat odeslání jakéhokoli zachovaného publikování přidruženého k tématu. Pokud vytvoříte poslední vůli a závěť jako zachované publikování, můžete ji použít k monitorování stavu klienta.

Klient například publikuje zachovanou publikaci, když se připojí a zveřejní její dostupnost. Zároveň vytváří nezadrženou poslední závěť a testamentovou publikaci, která oznamuje její nedostupnost. Kromě toho těsně před plánovaným odpojením publikuje svou nedostupnost jako zachovanou publikaci. Chcete-li zjistit, zda je klient k dispozici, přihlaste se k odběru tématu zachovaného publikování. Vždy byste obdrželi jednu ze tří publikací.

Pokud má klient přijímat zprávy publikované při odpojení, připojte klienta k předchozí relaci; viz [“MQTT bezstavové a stavové relace”](#) na stránce 128. Jeho odběry jsou aktivní, dokud nejsou odstraněny nebo dokud klient nevytvoří čistou relaci.

Windows

Linux

AIX

Volné spojení mezi klienty MQTT a aplikacemi

IBM MQ

Tok publikací mezi klienty MQTT a aplikacemi IBM MQ je volně spřažen. Publikace mohou pocházet buď z klienta MQTT, nebo z aplikace systému IBM MQ, a to v nenastaveném pořadí. Vydavatelé a odběratelé jsou volně vázáni. Vzájemně nepřímo komunikují prostřednictvím publikování a odběrů. Zprávy můžete také odeslat přímo klientovi MQTT z aplikace IBM MQ.

MQTT klienti a IBM MQ aplikace jsou volně spojeny ve dvou smyslech:

1. Vydavatelé a odběratelé jsou volně vázáni přidružením publikace a odběru k tématu. Vydavatelé a odběratelé obvykle nevědí o adrese nebo identitě jiného zdroje publikování nebo odběru.
2. Klienti MQTT publikují, odebírají, přijímají a zpracovávají potvrzení o doručení na samostatných podprocesech.

Klientská aplikace MQTT nečeká na doručení publikování. Aplikace předá zprávu klientovi MQTT a poté pokračuje ve svém vlastním podprocesu. Token doručení se používá k synchronizaci aplikace s doručením publikace; viz [Tokeny doručení](#).

Po předání zprávy klientovi MQTT má aplikace možnost čekat na token doručení. Klient může namísto čekání poskytnout metodu zpětného volání, která je volána při doručení publikování do produktu IBM MQ. Může také ignorovat token doručení.

V závislosti na kvalitě služby přidružené ke zprávě je token doručení vrácen okamžitě do metody zpětného volání, případně po delší době. Token doručení může být vrácen i po odpojení a opětovném připojení klienta. Je-li kvalita služby *spustit a zapomenout*, token doručení se vrátí okamžitě. V ostatních dvou případech je token doručení vrácen pouze v případě, že klient obdrží potvrzení, že publikování bylo odesláno odběratelům.

Publikace odeslané klientovi MQTT jako výsledek odběru klienta jsou doručeny do metody zpětného volání `messageArrived`. Produkt `messageArrived` se spouští v jiném podprocesu než hlavní aplikace.

Odesílání zpráv přímo klientovi MQTT

Zprávu můžete odeslat konkrétnímu klientovi MQTT jedním ze dvou způsobů.

1. Aplikace IBM MQ může odeslat zprávu přímo klientovi MQTT bez odběru. Viz téma [Odeslání zprávy přímo klientovi](#).
2. Alternativním přístupem je použít konvenci pojmenování `ClientIdentifier`. Nastavte, aby všichni odběratelé produktu MQTT vytvářeli odběry s použitím jejich jedinečného `ClientIdentifier` jako tématu. Publikujte do `ClientIdentifier`. Publikování je odesláno klientovi, který se přihlásil k odběru tématu `ClientIdentifier`. Pomocí této techniky můžete odeslat publikaci konkrétnímu odběrateli produktu MQTT.

Windows

Linux

AIX

Zabezpečení MQ Telemetry

Zabezpečení telemetrických zařízení může být důležité, protože zařízení jsou pravděpodobně přenosná a používají se v místech, která nemohou být pečlivě kontrolována. Pomocí sítě VPN můžete zabezpečit připojení ze zařízení MQTT ke službě telemetrie (MQXR). Produkt MQ Telemetry poskytuje dva další mechanismy zabezpečení, TLS a JAAS.

Protokol TLS se používá hlavně k šifrování komunikace mezi zařízeními a kanálem telemetrie a k ověřování, zda se zařízení připojuje ke správnému serveru. Viz téma [Ověřování kanálu telemetrie pomocí protokolu TLS](#). Pomocí protokolu TLS můžete také zkontrolovat, zda má zařízení klienta povoleno připojení k serveru. Viz téma [MQTT Ověřování klienta pomocí protokolu TLS](#).

Služba JAAS se používá hlavně ke kontrole, zda má uživatel zařízení povoleno používat serverovou aplikaci. Viz téma [MQTT Ověřování klienta pomocí hesla](#). JAAS lze použít s LDAP ke kontrole hesla pomocí adresáře jednotného přihlášení.

Protokoly TLS a JAAS lze použít ve spojení s dvoufaktorovou autentizací. Šifry používané protokolem TLS můžete omezit na šifry, které splňují standardy FIPS.

S nejméně desítkami tisíc uživatelů není vždy praktické poskytovat individuální bezpečnostní profily. Stejně tak není vždy praktické používat profily k autorizaci jednotlivých uživatelů pro přístup k objektům produktu IBM MQ. Místo toho seskupujte uživatele do tříd pro autorizaci publikování a odběru témat a odesílání publikací klientům.

Nakonfigurujte každý kanál telemetrie tak, aby mapoval klienty na společná ID uživatelů klienta. Pro každého klienta, který se připojuje ke specifickému kanálu, použijte společné ID uživatele. Viz téma [MQTT Identita a autorizace klienta](#).

Autorizace skupin uživatelů neohrožuje ověření každého jednotlivce. Každý jednotlivý uživatel může být ověřen na klientovi nebo na serveru pomocí svého jména uživatele a hesla a poté autorizován na serveru pomocí společného ID uživatele.

Windows

Linux

AIX

MQ Telemetry globalizace

Informační obsah zprávy v protokolu MQTT v3 je kódován jako bajtové pole. Obecně platí, že aplikace obsluhující text vytvoří informační obsah zprávy v souboru UTF-8. Kanál telemetrie popisuje informační obsah zprávy jako UTF-8, ale neprovádí žádné převody kódových stránek. Řetězec tématu publikování musí být UTF-8.

Aplikace je zodpovědná za převod abecedních dat na správnou kódovou stránku a numerická data na správné kódování čísel.

Klient MQTT Java má pohodlnou metodu `MqttMessage.toString`. Metoda zachází s informačním obsahem zprávy jako s kódováním ve výchozí znakové sadě lokální platformy, což je obecně UTF-8. Převode informační obsah na řetězec Java. Java má řetězcovou metodu `getBytes`, která převádí řetězec na bajtové pole kódované pomocí výchozí znakové sady lokální platformy. Dva programy MQTT Java, které si v informačním obsahu zprávy vyměňují text mezi platformami se stejnou výchozí znakovou sadou, to v produktu UTF-8 snadno a efektivně dělají.

Pokud výchozí znaková sada jedné z platforem není UTF-8, musí aplikace vytvořit konvenci pro výměnu zpráv. Vydavatel například určuje převod z řetězce na řetězec UTF-8 pomocí metody `getBytes("UTF8")`. Chcete-li přijmout text zprávy, odběratel předpokládá, že zpráva je zakódována ve znakové sadě UTF-8.

Služba telemetrie (MQXR) popisuje kódování všech příchozích publikování ze zpráv klientů MQTT jako UTF-8. Nastavuje `MQMD.CodedCharSetId` to UTF-8 a `RFH2.CodedCharSetId` to `MQCCSI_INHERIT`; viz ["Integrace produktu MQ Telemetry se správci front"](#) na stránce 125. Formát publikování je nastaven na `MQFMT_NONE`, takže žádný převod nelze provést pomocí kanálů nebo pomocí `MQGET`.

Windows

Linux

AIX

Výkon a rozšiřitelnost produktu MQ Telemetry

Při správě velkého počtu klientů a zlepšování rozšiřitelnosti produktu MQ Telemetry zvažte následující faktory.

Plánování kapacity

Informace o sestavách výkonu pro produkt MQ Telemetry naleznete v tématu [MQ Dokumenty o výkonu](#).

Připojení

Náklady spojené s připojením zahrnují

- Náklady na vytvoření samotného připojení z hlediska využití procesoru a času.
- Náklady na síť.

- Paměť, která se používá, když je připojení otevřené, ale nepoužívá se.

Při připojení klientů dochází k dodatečnému zatížení. Je-li připojení zachováno otevřené, toky TCP/IP a zprávy MQTT používají síť ke kontrole, zda je připojení stále k dispozici. Kromě toho se paměť používá na serveru pro každé připojení klienta, které je ponecháno otevřené.

Pokud odesíláte zprávy více než jednu za minutu, udržujte připojení otevřené, abyste se vyhnuli nákladům na zahájení nového připojení. Pokud posíláte zprávy méně než jednu každých 10-15 minut, zvažte zrušení připojení, abyste se vyhnuli nákladům na jeho otevření. Možná budete chtít ponechat připojení TLS otevřené, ale nečinné, po delší dobu, protože jeho nastavení je dražší.

Dále zvažte možnosti klienta. Je-li na klientovi k dispozici prostředek pro ukládání a předávání, můžete zprávy dávkově zpracovat a zrušit připojení mezi odesíláním dávek. Pokud je však klient odpojen, není možné, aby klient obdržel zprávu ze serveru. Proto účel vaší žádosti má vliv na rozhodnutí.

Má-li váš systém jednoho klienta odesílajícího více zpráv, například přenosy souborů, nečekejte na odpověď serveru na jednu zprávu. Místo toho odešlete všechny zprávy a na konci zkontrolujte, zda byly všechny přijaty. Alternativně použijte Kvalita služby (QoS).

Můžete měnit QoS podle zpráv a doručovat nedůležité zprávy pomocí QoS 0 a důležité zprávy pomocí QoS 2. Propustnost zpráv může být dvakrát vyšší s hodnotou QoS 0 než s hodnotou QoS 2.

Konvence pojmenování

Pokud navrhujete aplikaci pro mnoho klientů, implementujte efektivní konvenci pojmenování. Chcete-li mapovat každého klienta na správný `ClientIdentifier`, učiňte `ClientIdentifier` smysluplným. Dobrá konvence pojmenování usnadňuje administrátorovi zjistit, kteří klienti jsou spuštěni. Konvence pojmenování pomáhá administrátorovi filtrovat dlouhý seznam klientů v Průzkumníku IBM MQ a pomáhá při určování problémů; viz Identifikátor klienta.

Propustnost

Délka názvů témat ovlivňuje počet bajtů, které protékají sítí. Při publikování nebo přihlášení k odběru může být důležitý počet bajtů ve zprávě. Proto omezte počet znaků v názvu tématu. Když se MQTT klient přihlásí k odběru tématu IBM MQ, zadá mu název formuláře:

```
ClientIdentifier: TopicName
```

Chcete-li zobrazit všechny odběry pro klienta MQTT, můžete použít příkaz IBM MQ MQSC **DISPLAY** :

```
DISPLAY SUB(' ClientID1:*')
```

Definování prostředků v produktu IBM MQ pro použití klienty MQTT

Klient MQTT se připojuje ke vzdálenému správci front IBM MQ. Existují dvě základní metody, jak může aplikace IBM MQ odesílat zprávy klientovi MQTT: nastavte výchozí přenosovou frontu na `SYSTEM.MQTT.TRANSMIT.QUEUE` nebo použijte aliasy správce front. Definujte výchozí přenosovou frontu správce front, pokud existuje velký počet klientů MQTT. Použití výchozího nastavení přenosové fronty zjednodušuje administrativní práci; viz Konfigurovat distribuované fronty pro odesílání zpráv MQTT klientům.

Zlepšení rozšiřitelnosti tím, že se vyhnete předplatným.

Když se klient MQTT V3 přihlásí k odběru tématu, vytvoří se odběr pomocí služby telemetrie (MQXR) v produktu IBM MQ. Odběr směřuje publikování pro klienta na server `SYSTEM.MQTT.TRANSMIT.QUEUE`. Název vzdáleného správce front v záhlaví přenosu každého publikování je nastaven na hodnotu `ClientIdentifier` klienta MQTT, který provedl odběr. Pokud existuje mnoho klientů, z nichž každý vytváří své vlastní odběry, bude to mít za následek údržbu mnoha proxy odběrů v rámci klastru nebo hierarchie publikování/odběru produktu IBM MQ. Chcete-li získat informace o tom, že nepoužíváte

publikování/odběr, ale místo toho používáte řešení založené na bodu k určitému bodu, prohlédněte si téma [Odeslání zprávy přímo klientovi](#).

Správa velkého počtu klientů

Chcete-li podporovat mnoho souběžně připojených klientů, zvýšte velikost paměti dostupné pro službu telemetrie (MQXR) nastavením parametrů prostředí JVM **-Xms** a **-Xmx**. Postupujte takto:

1. Vyhledejte soubor `java.properties` v konfiguračním adresáři služby telemetrie. Viz [Konfigurační adresář služby telemetrie \(MQXR\) v umístění Windows](#) nebo [Konfigurační adresář služby telemetrie v umístění Linux](#).
2. Postupujte podle pokynů v souboru; halda o velikosti 1 GB je dostatečná pro 50 000 souběžně připojených klientů.

```
# Heap sizing options - uncomment the following lines to set the heap to 1G
#-Xmx1024m
#-Xms1024m
```

3. Přidejte další argumenty příkazového řádku, které mají být předány prostředí JVM spouštějící službu telemetrie (MQXR) v souboru `java.properties`. Viz téma [Předání parametrů prostředí JVM službě telemetrie \(MQXR\)](#).

Chcete-li zvýšit počet otevřených deskriptorů souboru v systému Linux, přidejte do souboru `/etc/security/limits.conf`/následující řádky a znovu se přihlaste.

```
@mqm soft nofile 65000
@mqm hard nofile 65000
```

Každý soket vyžaduje jeden deskriptor souboru. Služba telemetrie vyžaduje některé další deskriptory souboru, takže toto číslo musí být větší než požadovaný počet otevřených soketů.

Správce front používá pro každý přechodný odběr manipulátor objektu. Chcete-li podporovat mnoho aktivních přechodných odběrů, zvýšte maximální počet aktivních manipulátorů ve správci front. Příklad:

```
echo ALTER QMGR MAXHANDS(99999999) | runmqsc qMgrName
```

Obrázek 48. Změnit maximální počet popisovačů na systému Windows

```
echo "ALTER QMGR MAXHANDS(99999999)" | runmqsc qMgrName
```

Obrázek 49. Změnit maximální počet popisovačů na systému Linux

Další aspekty

Při plánování systémových požadavků zvažte dobu potřebnou k restartování systému. Plánovaný výpadek může mít důsledky pro počet zpráv, které jsou zařazeny do fronty a čekají na zpracování. Nakonfigurujte systém tak, aby zprávy mohly být úspěšně zpracovány v přijatelném čase. Zkontrolujte diskové úložiště, paměť a schopnost zpracování. U některých klientských aplikací může být možné vyřadit zprávy při opětovném připojení klienta. Chcete-li vyřadit zprávy, nastavte volbu `CleanSession` v parametrech připojení klienta; viz [Vyčistit relace](#). Alternativně můžete publikovat a odebírat pomocí nejlepší kvality služby `0v` klientu MQTT; viz [Kvalita služby](#). Při odesílání zpráv z IBM MQ použijte dočasné zprávy. Zprávy s těmito vlastnostmi služby nejsou obnoveny, když se systém nebo připojení restartuje.

Windows

Linux

AIX

Zařízení podporovaná produktem MQ Telemetry

Klienti MQTT mohou pracovat na řadě zařízení, od senzorů a akčních členů až po ruční zařízení a systémy vozidel.

Klienti systému MQTT jsou malí a běží na zařízeních omezených malou pamětí a nízkým výkonem zpracování. Server MQTT protocol je spolehlivý a má malá záhlaví, která vyhovují sítím s nízkou šířkou pásma, vysokými náklady a občasnou dostupností.

Produkt MQ Telemetry komunikuje s telemetrickými zařízeními prostřednictvím klientských aplikací MQTT. Tyto aplikace používají následující prostředky, z nichž všechny implementují protokol MQTT v3 :

• Následující knihovny klienta:

- *MQTT client for Java*, který se používá pro sestavení nativních aplikací pro (například) zařízení Android, OS X, Linux nebo Windows. Aplikace, které používají tuto klientskou knihovnu, mohou být spuštěny na všech variantách produktu Java od nejmenšího CLDC (Connected Limited Device Configuration) / MIDP (Mobile Information Device Profile) přes CDC (Connected Device Configuration) / Foundation, J2SE (Java Platform, Standard Edition) a J2EE (Java Platform, Enterprise Edition). Podporována je také přizpůsobená knihovna tříd IBM jclRM. Platforma Java ME se obecně používá na malých zařízeních, jako jsou aktuátory, senzory, mobilní telefony a další vestavěná zařízení. Platforma Java SE je obecně nainstalována na vyšších koncových vestavěných zařízeních, jako jsou stolní počítače a servery.
- *MQTT client for C*, který se používá pro sestavení nativních aplikací pro (například) zařízení iOS, OS X, Linux nebo Windows. Tato klientská knihovna poskytuje implementaci odkazů C společně s předem sestaveným nativním klientem pro systémy Windows a Linux. Referenční implementace jazyka C umožňuje portovat produkt MQTT na širokou škálu zařízení a platform. Některé systémy Windows v systému Intel, včetně systémů Windows 7, RedHat, Ubuntu a některých systémů Linux na platformách ARM, jako je Eurotech Viper, implementují verze produktu Linux, které spouští klienta C, ale produkt IBM neposkytuje servisní podporu pro platformy. Chcete-li volat centrum podpory IBM, musíte reprodukovat problémy s klientem na podporované platformě.
- *MQTT client for Java*, který se používá pro sestavení webových aplikací založených na prohlížeči.

Knihovny klienta MQTT jsou volně dostupné z prostředí Eclipse Paho a MQTT.org. Viz [IBM MQ Telemetry Transport ukázkové programy](#).

Zabezpečení v adresáři IBM MQ

V produktu IBM MQ existuje několik metod zajištění zabezpečení: rozhraní služby autorizace; uživatelské procedury nebo uživatelské procedury kanálu třetí strany; zabezpečení kanálu pomocí protokolu TLS (Transport Layer Security), záznamy ověřování kanálu a zabezpečení zpráv.

Rozhraní autorizační služby

Autorizaci pro použití volání MQI, příkazů a přístupu k objektům poskytuje **správce oprávnění k objektům** (OAM), který je standardně povolen. Přístup k entitám IBM MQ je řízen prostřednictvím skupin uživatelů IBM MQ a OAM. Administrátoři mohou použít rozhraní příkazového řádku k udělení nebo zrušení oprávnění podle potřeby.

Další informace o vytváření komponent služby autorizace naleznete v tématu [Nastavení zabezpečení na systémech AIX, Linux, and Windows](#).

Uživatelské procedury kanálu napsané uživatelem nebo třetí stranou

Kanály mohou používat uživatelské procedury nebo uživatelské procedury kanálu třetí strany. Další informace naleznete v tématu [Kanálové uživatelské programy pro kanály systému zpráv](#).

Zabezpečení kanálu pomocí protokolu TLS

Protokol TLS (Transport Layer Security) poskytuje standardní zabezpečení kanálu s ochranou proti odposlechu, manipulaci a zosobnění.

TLS používá k zajištění důvěrnosti a integrity zpráv a vzájemného ověření veřejný klíč a symetrické techniky.

Chcete-li získat souhrnný přehled o zabezpečení v produktu IBM MQ včetně podrobných informací o protokolu TLS, prohlédněte si téma [Zabezpečení](#). Přehled protokolu TLS, včetně ukazatelů na příkazy popsané v této části, naleznete v tématu [Protokoly šifrování zabezpečení: TLS](#).

Záznamy ověření kanálu

Pomocí záznamů ověřování kanálu můžete provádět přesnou kontrolu nad přístupem uděleným připojujícím se systémům na úrovni kanálu. Další informace naleznete v tématu [Záznamy ověření kanálu](#).

Zabezpečení zpráv

Pomocí produktu Advanced Message Security, který je samostatně instalovanou a licencovanou komponentou produktu IBM MQ, můžete poskytovat kryptografickou ochranu pro zprávy odesílané a přijímané pomocí produktu IBM MQ. Viz [Advanced Message Security](#).

Související úlohy

[Zabezpečení](#)

[Plánování požadavků na zabezpečení](#)

Podpora zabezpečení TLS spravovaného klienta IBM MQ.NET

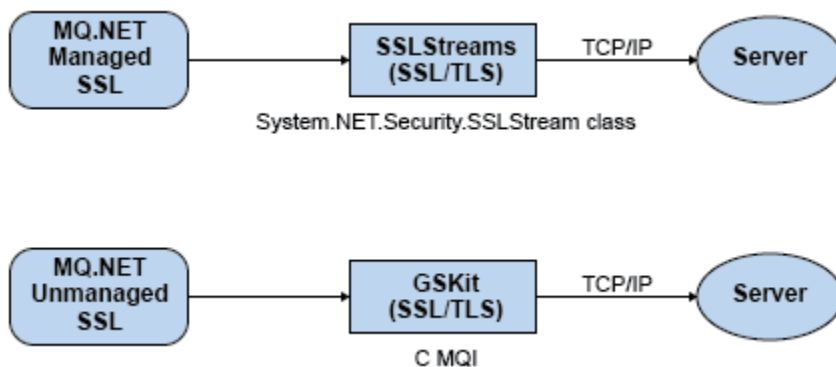
Plně spravovaný klient produktu IBM MQ.NET poskytuje podporu TLS (Transport Layer Security), která je založena na sadě SSLStreams produktu Microsoft.NET. Liší se od ostatních klientů systému IBM MQ, kteří jsou založeni na IBM Global Security Kit (GSKit).

Můžete vyvíjet aplikace produktu IBM MQ.NET pro spuštění ve spravovaném nebo nespravovaném režimu.

- Ve spravovaném režimu pracují aplikace .NET v rámci produktu .NET CLR (Common Language Runtime) bez jakéhokoli vyvolání mezi platformami, jako je například vyvolání rozhraní MQI jazyka C.
- V nespravovaném režimu je pro základní operace MQI vyvolána rozhraní MQI jazyka C. Rozhraní nespravovaného režimu se v podstatě skládá z tříd modulu wrapper .NET nad rozhraním MQI jazyka C.

Spravovaný klient produktu IBM MQ.NET používá knihovny rámce Microsoft.NET k implementaci protokolů zabezpečeného soketu TLS. Zabezpečení systému.NET.Security.SSLStream z Microsoft se používá pro implementaci zabezpečení (TLS) v produktu IBM MQ.NET.

Nespravovaný režim klienta produktu IBM MQ.NET již podporuje funkci TLS, která je založena na rozhraní MQI jazyka C (a GSKit). To znamená, že operace TLS jsou zpracovány rozhraním MQI jazyka C. V tomto případě produkt GSKit implementuje protokoly zabezpečeného soketu TLS.



Obrázek 50. Porovnání spravovaného a nespravovaného TLS produktu IBM MQ.NET

Následující tabulka shrnuje rozdíly mezi spravovanými a nespravovanými implementacemi:

Tabulka 14. Rozdíly mezi spravovanými a nespravovanými implementacemi

Režim	Protokoly	Implementace	Komentář
IBM MQ.NET spravované zabezpečení SSL	TLS	System.NET.Security.SSLStream Třída SSLStream funguje jako proud přes připojený soket TCP	TLS 1.0 TLS 1.2 (pouze s rámcem Microsoft.NET Framework v4.5)
IBM MQ.NET nespravované zabezpečení SSL	TLS	GSKIT a C-MQI	Protokoly zabezpečených soketů TLS

Související pojmy

Podpora zabezpečení SSL (Secure Sockets Layer) a TLS (Transport Layer Security) pro produkt .NET

IBM MQ MQI clients

IBM MQ MQI klient je komponenta produktu IBM MQ, kterou lze nainstalovat v systému, v němž není spuštěn žádný správce front.

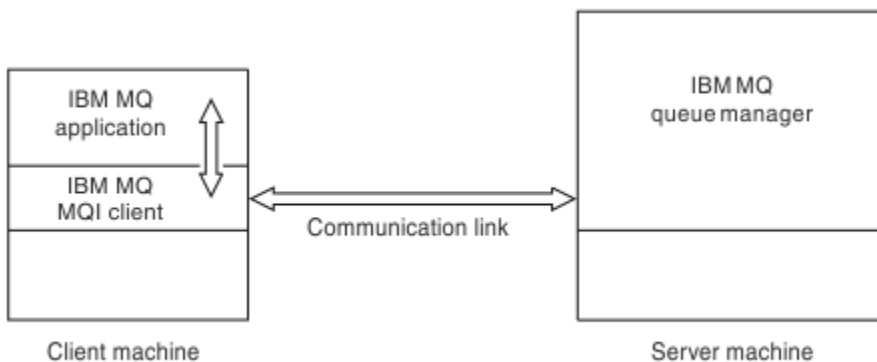
IBM MQ Klient MQI je komponenta, která umožňuje aplikaci spuštěné v systému spouštět volání MQI pro správce front spuštěného v jiném systému. Výstup volání je odeslán zpět klientovi, který jej předá zpět aplikaci.

Pomocí produktu IBM MQ MQI klient se může aplikace spuštěná ve stejném systému jako klient připojit ke správci front, který je spuštěn v jiném systému. Aplikace může pro tohoto správce front zadat volání MQI. Taková aplikace se nazývá aplikace IBM MQ MQI klient a správce front se nazývá *správce front serveru*.

IBM MQ Server je správce front, který poskytuje služby řazení do front jednomu nebo více klientům. Všechny objekty IBM MQ, například fronty, existují pouze v počítači správce front (počítač serveru IBM MQ), nikoli v klientu. Server IBM MQ může také podporovat lokální aplikace IBM MQ.

Rozdíl mezi serverem IBM MQ a běžným správcem front spočívá v tom, že server má vyhrazené komunikační propojení s každým klientem. Další informace o vytváření kanálů pro klienty a servery naleznete v tématu [Konfigurace distribuovaného řazení do front](#).

Aplikace IBM MQ MQI klient a správce front serveru spolu komunikují pomocí *kanálu MQI*. Kanál MQI se spustí, když aplikace klienta vydá volání **MQCONN** nebo **MQCONNX** pro připojení ke správci front, a ukončí se, když aplikace klienta vydá volání **MQDISC** pro odpojení od správce front. Vstupní parametry toku volání MQI v jednom směru kanálu MQI a výstupní parametry v opačném směru.



Obrázek 51. Propojení mezi klientem a serverem

Lze použít následující platformy. Kombinace závisí na tom, který produkt IBM MQ používáte, a jsou popsány v části [“Podpora platformy pro klienty IBM MQ”](#) na stránce 137.

IBM MQ MQI client

AIX and Linux
Windows
IBM i

Server IBM MQ

AIX and Linux
Windows
IBM i
z/OS

Rozhraní MQI je k dispozici pro aplikace spuštěné na platformě klienta. Fronty a další objekty IBM MQ jsou uloženy ve správci front nainstalovaném na serveru.

Aplikace, kterou chcete spustit v prostředí IBM MQ MQI client, musí být nejprve propojena s příslušnou klientskou knihovnou. Když aplikace vydá volání MQI, produkt IBM MQ MQI client přeměruje požadavek na správce front, kde je zpracován a odkud je odeslána odpověď zpět do produktu IBM MQ MQI client.

Odkaz mezi aplikací a produktem IBM MQ MQI client je vytvořen dynamicky za běhu.

Můžete také vyvíjet klientské aplikace pomocí IBM MQ classes for .NET, IBM MQ classes for Java nebo IBM MQ classes for Java Message Service (JMS). Klienty Java a JMS můžete používat na následujících platformách:

-  IBM i
-  AIX
-  Linux
-  Windows

Použití Java a JMS zde není popsáno. Úplné podrobnosti o instalaci, konfiguraci a použití IBM MQ classes for Java a IBM MQ classes for JMS viz [Použití IBM MQ classes for Java](#) a [Použití IBM MQ classes for JMS](#).

Aplikace IBM MQ v prostředí klient-server

Při propojení se serverem mohou klientské aplikace IBM MQ vydávat většinu volání MQI stejným způsobem jako lokální aplikace. Aplikace klienta zadá volání MQCONN pro připojení k určenému správci front. Tento správce front poté zpracuje všechna další volání MQI, která určují manipulátor připojení vrácený z požadavku na připojení.

Aplikace musíte propojit s příslušnými klientskými knihovnami. Viz [Sestavení aplikací pro IBM MQ MQI clients](#).

Související pojmy

[“Proč používat klienty IBM MQ ?” na stránce 137](#)

Použití klientů systému IBM MQ představuje účinný způsob implementace systému zpráv a řazení do front systému IBM MQ.

[“Co je to rozšířený transakční klient?” na stránce 138](#)

Rozšířený transakční klient IBM MQ může aktualizovat prostředky spravované jiným správcem prostředků pod kontrolou externího správce transakcí.

[“Jak se klient připojuje k serveru” na stránce 139](#)

Klient se připojuje k serveru pomocí MQCONN nebo MQCONNX a komunikuje prostřednictvím kanálu.

[“Správa a podpora transakcí” na stránce 141](#)

Úvod do správy transakcí a jak produkt IBM MQ podporuje transakce.

[“Rozšíření prostředků správce front” na stránce 142](#)

Prostředky správce front můžete rozšířit pomocí uživatelských procedur, uživatelských procedur rozhraní API nebo instalovatelných služeb.

Související informace

[Jak nastavit IBM MQ MQI client](#)

Proč používat klienty IBM MQ ?

Použití klientů systému IBM MQ představuje účinný způsob implementace systému zpráv a řazení do front systému IBM MQ .

Můžete mít aplikaci, která používá rozhraní MQI spuštěné na jednom počítači, a správce front spuštěnou na jiném počítači (fyzickém nebo virtuálním). Výhody tohoto jednání jsou:

- V klientském počítači není třeba provádět úplnou implementaci IBM MQ .
- Požadavky na hardware v klientském systému jsou sníženy.
- Požadavky na administraci systému jsou sníženy.
- Aplikace IBM MQ spuštěná v klientu se může připojit k více správcům front v různých systémech.
- Lze použít alternativní kanály používající různé přenosové protokoly.

Podpora platformy pro klienty IBM MQ

Produkt IBM MQ na všech podporovaných platformách serveru přijímá klientská připojení z produktu IBM MQ MQI clients na řadě platform.

Produkt IBM MQ instalovaný jako *Základní produkt a server* na všech podporovaných platformách serveru může přijímat připojení z produktu IBM MQ MQI clients na následujících platformách:

-  IBM i
-  AIX
-  Linux
-  Windows

Klientská připojení se liší v identifikátoru kódované znakové sady (CCSID) a v komunikačním protokolu.

Jaké aplikace se spouštějí na serveru IBM MQ MQI client?

Úplné rozhraní MQI je podporováno v prostředí klienta. To umožňuje konfigurovat téměř všechny aplikace IBM MQ pro spouštění v systému IBM MQ MQI client propojením aplikace v produktu IBM MQ MQI client s knihovnou MQIC a nikoli s knihovnou MQI. Výjimky jsou:

- MQGET se signálem
- Aplikace vyžadující koordinaci synchronizačních bodů s ostatními správci prostředků musí používat rozšířeného transakčního klienta.

Je-li povoleno dopředné čtení, nejsou k dispozici všechny volby MQGET, aby se zlepšil výkon dočasného systému zpráv. V tabulce jsou uvedeny povolené volby a informace o tom, zda je lze mezi voláními MQGET měnit.

Hodnoty	Povoleno, když je povoleno dopředné čtení a lze jej změnit mezi voláními MQGET	Povoleno, když je povoleno dopředné čtení, ale nelze je měnit mezi voláními MQGET ¹	Volby MQGET, které nejsou povoleny, když je povoleno dopředné čtení ²
Hodnoty MQGET MD	MsgId ³ CorrelId ³	Kódování CodedCharSetId	

Tabulka 15. Volby MQGET jsou povoleny, když je povoleno dopředné čtení (pokračování)

Hodnoty	Povoleno, když je povoleno dopředné čtení a lze jej změnit mezi voláními MQGET	Povoleno, když je povoleno dopředné čtení, ale nelze je měnit mezi voláními MQGET ¹	Volby MQGET, které nejsou povoleny, když je povoleno dopředné čtení ²
Volby MQGET MQGMO	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 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 MQGMO_BACKOUT MQGMO_MSG_UNDER_CURSOR ⁴ MQGMO_LOCK MQGMO_UNLOCK
Hodnoty MQGMO		MsgHandle	

1. Pokud se tyto volby změní mezi voláními MQGET, vrátí se kód příčiny MQRC_OPTIONS_CHANGED.
2. Pokud se tyto volby zadaly při prvním volání MQGET, bude dopředné čtení zablokováno. Budou-li tyto volby zadány při následném volání MQGET, vrátí se kód příčiny MQRC_OPTIONS_ERROR.
3. Aplikace klienta si musí být vědomy toho, že pokud se hodnoty MsgId a CorrelId změní mezi voláními MQGET, zprávy s předchozími hodnotami již mohly být odeslány na klienta a zůstávají ve vyrovnávací paměti klienta pro dopředné čtení, dokud nebudou zpracovány (nebo automaticky vyprázdněny).
4. První volání MQGET určuje, zda se mají zprávy procházet nebo získat z fronty, je-li povoleno dopředné čtení. Pokud se aplikace pokusí použít kombinaci procházení a získání, vrátí se kód příčiny MQRC_OPTIONS_CHANGED.
5. MQGMO_MSG_UNDER_CURSOR nelze použít s dopředným čtením. Zprávy lze procházet nebo získat, je-li dopředné čtení povoleno, ale ne kombinaci obojího.

Aplikace spuštěná v systému IBM MQ MQI client se může souběžně připojit k více než jednomu správci front nebo může použít název správce front s hvězdičkou (*) ve volání MQCONN nebo MQCONNEX (viz příklady v části Připojení aplikací IBM MQ MQI client ke správcům front).

Co je to rozšířený transakční klient?

Rozšířený transakční klient IBM MQ může aktualizovat prostředky spravované jiným správcem prostředků pod kontrolou externího správce transakcí.

Pokud nejste obeznámeni s koncepty správy transakcí, viz [“Správa a podpora transakcí”](#) na stránce 141.

Všimněte si, že transakční klient XA je nyní dodáván jako součást produktu IBM MQ.

Klientská aplikace se může podílet na transakci, která je spravována správcem front, k němuž je připojena. V rámci pracovní jednotky může klientská aplikace vkládat zprávy do front vlastních tímto správcem front a získávat z nich zprávy. Klientská aplikace pak může použít volání **MQCMIT** k potvrzení pracovní jednotky nebo volání **MQBACK** k vrácení pracovní jednotky. V rámci stejné pracovní jednotky však aplikace klienta nemůže aktualizovat prostředky jiného správce prostředků, například tabulky databáze Db2. Použití rozšířeného transakčního klienta IBM MQ toto omezení odebere.

IBM MQ Rozšířený transakční klient je IBM MQ MQI client s nějakou další funkcí. Pomocí této funkce může klientská aplikace v rámci stejné pracovní jednotky provádět následující úlohy:

- Vkládání zpráv do front vlastněných správcem front, k němuž je připojen, a získávání zpráv z těchto front.
- Aktualizovat prostředky jiného správce prostředků než správce front IBM MQ

Tuto pracovní jednotku musí spravovat externí správce transakcí, který je spuštěn na stejném systému jako klientská aplikace. Pracovní jednotku nelze spravovat správcem front, ke kterému je klientská aplikace připojena. To znamená, že správce front může fungovat pouze jako správce prostředků, nikoli jako správce transakcí. To také znamená, že klientská aplikace může potvrdit nebo vrátit pracovní jednotku pouze pomocí rozhraní API (Application Programming Interface), které poskytuje externí správce transakcí. Aplikace klienta proto nemůže používat volání MQI **MQBEGIN**, **MQCMIT** a **MQBACK**.

Externí správce transakcí komunikuje se správcem front jako se správcem prostředků pomocí stejného kanálu MQI, který používá klientská aplikace připojená ke správci front. V situaci zotavení po selhání však může správce transakcí v případě, že nejsou spuštěny žádné aplikace, použít vyhrazený kanál MQI k obnovení neúplných pracovních jednotek, v nichž se správce front účastnil v době selhání.

V této sekci se na IBM MQ MQI client, který nemá rozšířenou transakční funkci, odkazuje jako na základního klienta IBM MQ. Proto můžete uvažovat o IBM MQ rozšířeném transakčním klientovi, který se bude skládat ze základního klienta IBM MQ s přidáním rozšířené transakční funkce.





Poznámka:  IBM MQ MQI client on IBM i nepodporuje rozšířenou transakční funkci IBM MQ.


Podpora platformy pro rozšířené transakční klienty

Multi

Rozšíření klienti transakcí jsou k dispozici pro všechny platformy Multiplatforms, které podporují základního klienta. Klienti nejsou k dispozici pro z/OS.

Klientská aplikace, která používá rozšířeného transakčního klienta, se může připojit pouze ke správci front následujících produktů IBM MQ 9.0 nebo novějších produktů:

-  IBM MQ for AIX
-  IBM MQ for IBM i
-  položky IBM MQ pro Linux
-  IBM MQ for Windows

 Ačkoli v systému z/OS nejsou spuštěni žádní přídatní transakční klienti, aplikace klienta, která používá rozšířeného transakčního klienta, se může připojit ke správci front, který je spuštěn v systému z/OS.

Pro každou platformu jsou hardwarové a softwarové požadavky pro rozšířeného transakčního klienta stejné jako požadavky pro základního klienta IBM MQ. Programovací jazyk je podporován rozšířeným transakčním klientem, pokud je podporován základním klientem IBM MQ a správcem transakcí, kterého používáte.

Informace o externích správcích transakcí pro všechny platformy viz [Systémové požadavky pro IBM MQ](#).

Jak se klient připojuje k serveru

Klient se připojuje k serveru pomocí MQCONN nebo MQCONNX a komunikuje prostřednictvím kanálu.

Aplikace spuštěná v prostředí klienta IBM MQ musí udržovat aktivní připojení mezi počítači klienta a serveru.

Připojení je vytvořeno aplikací, která zadává volání MQCONN nebo MQCONNX. Klienti a servery komunikují prostřednictvím *kanálů MQI* nebo v případě sdílení konverzací sdílejí jednotlivé konverzace instanci kanálu

MQI. Po úspěšném volání zůstane instance kanálu MQI nebo konverzace připojena, dokud aplikace nevydá volání MQDISC. Jedná se o případ každého správce front, ke kterému se aplikace potřebuje připojit.

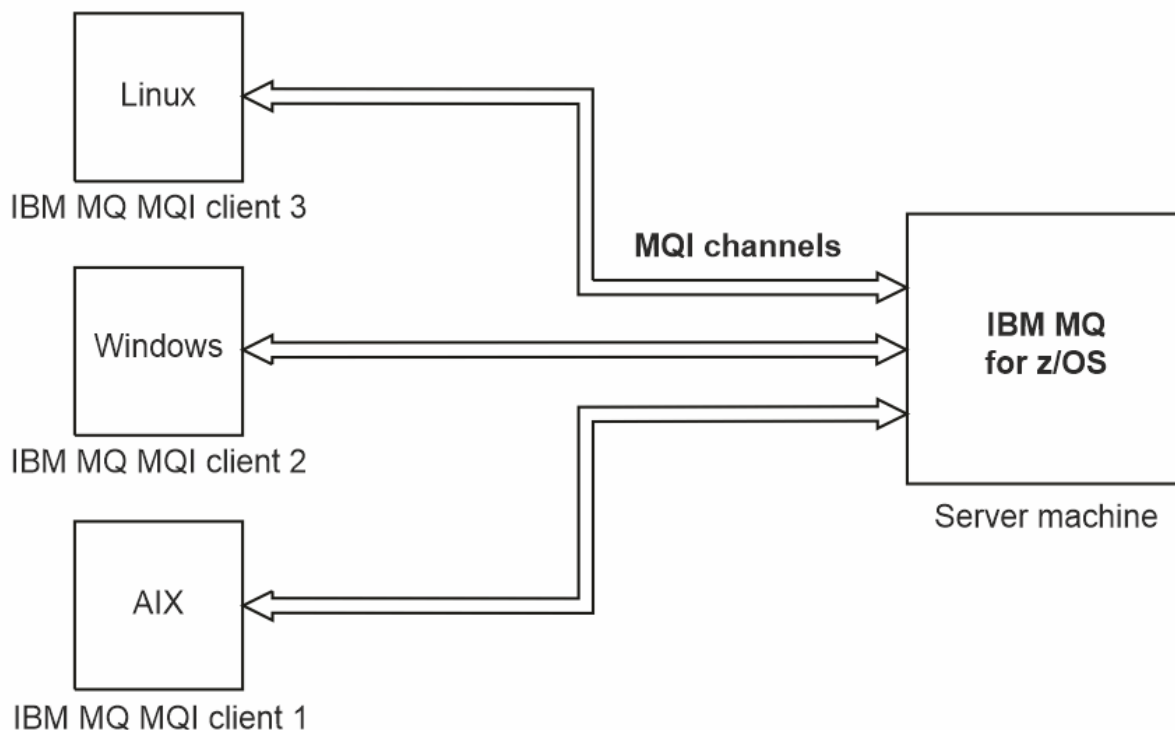
Klient a správce front ve stejném počítači

Můžete také spustit aplikaci v prostředí IBM MQ MQI client , když má váš počítač také nainstalovaného správce front.

V této situaci máte možnost připojit se ke knihovnam správce front nebo ke knihovnam klienta, ale pamatujte, že pokud se připojujete ke knihovnam klienta, musíte stále definovat připojení kanálu. To může být užitečné během fáze vývoje aplikace. Svůj program můžete testovat na svém vlastním počítači, bez závislosti na ostatních, a být si jisti, že bude i nadále fungovat, když jej přesunete do nezávislého prostředí IBM MQ MQI client .

Klienti na různých platformách

V tomto příkladu počítač serveru komunikuje se třemi systémy IBM MQ MQI clients na různých platformách.



Obrázek 52. Server IBM MQ připojený ke klientům na různých platformách

Další složitější prostředí jsou možná. Klient IBM MQ se může například připojit k více než jednomu správci front nebo k libovolnému počtu správců front připojených jako součást skupiny sdílení front.

Použití různých verzí softwaru klienta a serveru

Pokud používáte předchozí verze produktů IBM MQ , ujistěte se, že server podporuje převod kódu z CCSID vašeho klienta.

Klient IBM MQ se může připojit ke všem podporovaným verzím správce front. Pokud se připojujete ke správci front dřívější verze, nemůžete používat funkce a struktury z novější verze produktu v aplikaci IBM MQ v klientu.

Správce front IBM MQ může komunikovat s klienty v různých verzích sám se sebou vyjednáváním na nejvyšší vzájemně podporované úrovni protokolu. To znamená, že starší klienti mohou být použiti s pozdějšími úrovněmi správce front. Doporučuje se, aby klient i server byly ve verzích produktu IBM MQ, které jsou v současné době podporovány, aby usnadnily diagnostiku problémů a povolily podporu produktu IBM.

Další informace viz programovací jazyky podporované v části [Vývoj aplikací](#).

Správa a podpora transakcí

Úvod do správy transakcí a jak produkt IBM MQ podporuje transakce.

Správce prostředků je počítačový subsystém, který vlastní a spravuje prostředky, k nimž mohou přistupovat a které mohou být aktualizovány aplikacemi. Zde jsou příklady správců prostředků:

- Správce front IBM MQ s prostředky, které jsou jeho frontami.
- Databáze Db2 s prostředky, které jsou jejími tabulkami.

Když aplikace aktualizuje prostředky jednoho nebo více správců prostředků, může existovat obchodní požadavek, který zajistí, že určité aktualizace budou všechny úspěšně dokončeny jako skupina, nebo že žádná z nich nebude dokončena. Důvodem tohoto druhu požadavku je, že obchodní data by byla ponechána v nekonzistentním stavu, pokud by některé z těchto aktualizací byly úspěšně dokončeny, ale jiné ne.

Aktualizace prostředků, které jsou spravovány tímto způsobem, se vyskytují v *transakci* nebo *transakci*. Aplikací program může seskupit sadu aktualizací do jednotky práce.

Během pracovní jednotky aplikace požádá správce prostředků o aktualizaci svých prostředků. Jednotka práce skončí, když aplikace vydá požadavek na potvrzení všech aktualizací. Dokud nebudou aktualizace potvrzeny, nebude žádná z nich viditelná pro jiné aplikace, které přistupují ke stejným prostředkům. Alternativně, pokud se aplikace rozhodne, že nemůže dokončit pracovní jednotku z jakéhokoli důvodu, může vydat požadavek na vrácení všech aktualizací, které požadovala až do tohoto bodu. V tomto případě se žádná z aktualizací nikdy nestane viditelnou pro jiné aplikace. Tyto aktualizace jsou obvykle logicky související a musí být úspěšné, aby byla zachována integrita dat. Pokud je jedna aktualizace úspěšná, zatímco jiná selže, integrita dat je ztracena.

Když je jednotka práce úspěšně dokončena, říká se *commit* (potvrdit). Po potvrzení jsou všechny aktualizace provedené v rámci této jednotky práce trvalé a nevratné. Pokud však jednotka práce selže, všechny aktualizace se namísto toho *vycouvají*. Tento proces, kde jsou pracovní jednotky potvrzeny nebo zálohovány s integritou, se nazývá *koordinace synchronizačních bodů*.

Časový okamžik, kdy jsou všechny aktualizace v rámci transakce potvrzeny nebo vráceny zpět, se nazývá *synchronizační bod*. Aktualizace v rámci pracovní jednotky se označuje jako *v rámci řízení synchronizačního bodu*. Pokud aplikace požádá o aktualizaci, která je *mimo řízení synchronizačního bodu*, správce prostředků aktualizaci okamžitě potvrdí, a to i v případě, že probíhá transakce a aktualizaci nelze později vrátit zpět.

Počítačový subsystém, který spravuje jednotky práce, se nazývá *správce transakcí* nebo *koordinátor bodů*.

Lokální jednotka práce je taková, ve které jsou aktualizovány pouze prostředky správce front IBM MQ. Zde je koordinace synchronizačních bodů poskytována samotným správcem front pomocí procesu jednofázového potvrzování.

Globální jednotka práce je jednotka práce, v níž jsou aktualizovány také prostředky patřící jiným správcům prostředků, například databáze kompatibilní s podporou XA. Zde musí být použita procedura dvofázového potvrzování a pracovní jednotka může být koordinována samotným správcem front nebo externě jiným správcem transakcí vyhovujícím standardu XA, například IBM TXSeries, nebo BEA Tuxedo.

Správce transakcí je odpovědný za to, že všechny aktualizace prostředků v rámci jednotky práce byly úspěšně dokončeny, nebo že žádná z nich nebyla dokončena. Je na správci transakcí, aby aplikace zadala požadavek na potvrzení nebo odvolání transakce. Příklady správců transakcí jsou CICS a WebSphere Application Server, ačkoli oba mají také jinou funkci.

Někteří správci prostředků poskytují vlastní funkci správy transakcí. Správce front IBM MQ může například spravovat pracovní jednotky zahrnující aktualizace vlastních prostředků a aktualizace tabulek Db2. Správce front k provedení této funkce nepotřebuje samostatného správce transakcí, ačkoli jej lze použít v případě, že se jedná o požadavek uživatele. Pokud se používá samostatný správce transakcí, nazývá se *externí správce transakcí*.

Aby mohl externí správce transakcí spravovat pracovní jednotku, musí existovat standardní rozhraní mezi správcem transakcí a každým správcem prostředků, který se účastní pracovní jednotky. Toto rozhraní umožňuje správcem transakcí a správcem prostředků vzájemnou komunikaci. Jedním z těchto rozhraní je *Rozhraní XA*, což je standardní rozhraní podporované řadou správců transakcí a správců prostředků. Rozhraní XA publikuje skupina Open Group ve *Distribuované zpracování transakcí: Specifikace XA*.

Pokud se na transakci podílí více než jeden správce prostředků, musí správce transakcí použít protokol *dvoufázového potvrzování*, aby zajistil, že všechny aktualizace v rámci pracovní jednotky budou úspěšně dokončeny, nebo že žádná z nich nebude dokončena, a to ani v případě selhání systému. Když aplikace vydá správcem transakcí požadavek na potvrzení transakce, provede správce transakcí následující:

Fáze 1 (připravit na potvrzení)

Správce transakcí požádá každého správce prostředků, který se účastní pracovní jednotky, aby se ujistil, že všechny informace o zamýšlených aktualizacích jeho prostředků jsou v obnovitelném stavu. Správce prostředků to obvykle provádí zápisem informací do protokolu a zajištěním zápisu informací na pevný disk. Fáze 1 se dokončí, když správce transakcí obdrží od každého správce prostředků oznámení, že informace o zamýšlených aktualizacích jeho prostředků jsou v obnovitelném stavu.

Fáze 2 (potvrdit)

Po dokončení fáze 1 učiní správce transakcí neodvolatelné rozhodnutí o potvrzení pracovní jednotky. Požádá každého správce prostředků, který se účastní pracovní jednotky, aby potvrdil aktualizace svých prostředků. Když správce prostředků obdrží tento požadavek, musí potvrdit aktualizace. Nemá možnost je v této fázi vycouvat. Fáze 2 se dokončí, jakmile správce transakcí obdrží od každého správce prostředků oznámení, že provedl aktualizaci svých prostředků.

Rozhraní XA používá protokol dvoufázového potvrzování.

Další informace viz [Scénáře podpory transakcí](#).

Produkt IBM MQ také poskytuje podporu pro transakční server Microsoft (COM+). [Použití Microsoft Transakčního serveru \(COM+\)](#) poskytuje informace o tom, jak nastavit produkt IBM MQ tak, aby využíval výhod podpory modelu COM+.

Rozšíření prostředků správce front

Prostředky správce front můžete rozšířit pomocí uživatelských procedur, uživatelských procedur rozhraní API nebo instalovatelných služeb.

Uživatelské procedury

Uživatelské procedury poskytují mechanismus pro vložení vlastního kódu do funkce správce front. Podporované uživatelské procedury zahrnují:

Uživatelské procedury kanálu

Tyto východy mění způsob, jakým kanály fungují. Uživatelské procedury kanálu jsou popsány v tématu [Programy uživatelské procedury kanálu pro kanály systému zpráv](#).

Ukončení převodu dat

Tyto uživatelské procedury vytvářejí fragmenty zdrojového kódu, které lze vložit do aplikačních programů za účelem převodu dat z jednoho formátu do jiného. Uživatelské procedury pro převod dat jsou popsány v tématu [Zápis uživatelských procedur pro převod dat](#).

Ukončení pracovní zátěže klastru

Funkce prováděná touto uživatelskou procedurou je definována poskytovatelem uživatelské procedury. Informace o definici volání jsou uvedeny v části [MQ_CLUSTER_WORKLOAD_EXIT-Popis volání](#).

Uživatelské procedury rozhraní API

Uživatelské procedury rozhraní API umožňují psát kód, který mění chování volání rozhraní API produktu IBM MQ, například MQPUT a MQGET, a poté tento kód vložit bezprostředně před nebo bezprostředně po těchto voláních. Vložení je automatické; správce front řídí kód ukončení v registrovaných bodech. Další informace o uživatelských procedurách rozhraní API naleznete v tématu [Použití a zápis uživatelských procedur rozhraní API](#).

Instalovatelné služby

Instalovatelné služby mají formalizovaná rozhraní (rozhraní API) s více vstupními body.

Implementace instalovatelné služby se nazývá *komponenta služby*. Můžete použít komponenty dodávané s produktem IBM MQ nebo můžete napsat vlastní komponentu k provedení požadovaných funkcí.

V současné době jsou poskytovány následující instalovatelné služby:

Autorizační služba

Autorizační služba vám umožňuje vytvořit si vlastní bezpečnostní zařízení.

Výchozí komponentou služby, která implementuje službu, je správce oprávnění k objektu (OAM). Standardně je OAM aktivní a nemusíte dělat nic pro jeho konfiguraci. Rozhraní služby autorizace můžete použít k vytvoření dalších komponent, které nahradí nebo rozšíří OAM. Další informace o modulu OAM naleznete v tématu [Nastavení zabezpečení na systémech AIX, Linux, and Windows](#).

služba názvů

Služba názvů umožňuje aplikacím sdílet fronty identifikací vzdálených front, jako by se jednalo o lokální fronty.

Můžete napsat vlastní komponentu služby názvů. To můžete provést například v případě, že hodláte používat službu názvů s produktem IBM MQ. Chcete-li použít službu názvů, musíte mít buď komponentu, která je napsaná uživatelem, nebo dodávaná jiným dodavatelem softwaru. Standardně je služba názvů neaktivní.



Související pojmy

[Uživatelské procedury](#), [uživatelské procedury rozhraní API](#) a [instalovatelné služby IBM MQ](#)

Jazyková rozhraní produktu IBM MQ Java

Produkt IBM MQ poskytuje tři rozhraní API pro použití v aplikacích Java: IBM MQ classes for Jakarta Messaging, IBM MQ classes for JMS a IBM MQ classes for Java.

Produkt IBM podporuje a je aktivním účastníkem otevřených standardů.

- V produktu IBM MQ 8.0 produkt implementuje standard JMS 2.0, který zavedl nové zjednodušené rozhraní API spolu s funkcemi, jako jsou sdílené odběry.
-   V produktu IBM MQ 9.3.0 je podporován také produkt [Jakarta Messaging 3.0](#).
- Kromě toho WebSphere Liberty podporuje JMS 2.0 a Jakarta Messaging 3.0 s IBM MQ.

V rámci produktu IBM MQ existují tři rozhraní API pro použití v aplikacích Java:

IBM MQ classes for Jakarta Messaging

IBM MQ classes for Jakarta Messaging je poskytovatel systému Jakarta Messaging, který implementuje rozhraní Jakarta Messaging pro systém zpráv IBM MQ. Produkt Jakarta Connectors Architecture poskytuje standardní způsob připojení aplikací spuštěných v prostředí Jakarta EE k podnikovému informačnímu systému (EIS), například IBM MQ nebo Db2.

IBM MQ classes for JMS

IBM MQ classes for JMS je poskytovatel systému JMS, který implementuje rozhraní JMS pro systém zpráv IBM MQ. Java Platform, Enterprise Edition Connector Architecture (JCA) poskytuje standardní

způsob připojení aplikací spuštěných v prostředí Java EE k podnikovému informačnímu systému (EIS), například IBM MQ nebo Db2.

IBM MQ classes for Java

IBM MQ classes for Java vám umožňuje používat produkt IBM MQ v prostředí Java . IBM MQ classes for Java Povolit aplikaci Java připojit se k produktu IBM MQ jako klient IBM MQ nebo se připojit přímo ke správci front IBM MQ .

Poznámka:

- **V 9.3.0** **V 9.3.0** JMS 2.0 byl nahrazen Jakarta Messaging. Produkt IBM MQ classes for JMS nadále podporuje standard JMS 2.0 , ale budoucí vylepšení systému zpráv Java se objeví pouze v souboru Jakarta Messaging, tedy v souboru IBM MQ classes for Jakarta Messaging. IBM MQ classes for JMS se doporučuje pouze pro údržbu a rozšíření existujících aplikací JMS 2.0 . IBM MQ classes for Jakarta Messaging by měla být upřednostňovanou technologií pro nový vývoj.
- **Stabilized** IBM MQ classes for Java jsou funkčně stabilizovány na úrovni dodávané v produktu IBM MQ 8.0. Existující aplikace, které používají produkt IBM MQ classes for Java , budou i nadále plně podporovány, ale toto rozhraní API je stabilizováno, takže nové funkce nebudou přidány a požadavky na vylepšení budou odmítnuty. Plně podporováno znamená, že defekty budou opraveny společně se změnami, které jsou vynucené změnami systémových požadavků IBM MQ .

V 9.3.0 **JM 3.0** **V 9.3.0** V produktu IBM MQ 9.3 jsou IBM MQ classes for Java, IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging sestaveny pomocí Java 8. Běžová prostředí produktu Java na těchto úrovních nebo nad nimi musí být použita ke spuštění aplikací používajících tato rozhraní.

Související pojmy

[Přístup k produktu IBM MQ z Java -Volba rozhraní API](#)

V 9.3.0 **V 9.3.0** [Proč bych měl používat třídy IBM MQ pro systém zpráv Jakarta?](#)

[Proč bych měl používat IBM MQ classes for JMS?](#)

[Proč bych měl používat IBM MQ classes for Java?](#)

IBM MQ classes for JMS/Jakarta Messaging

IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging jsou poskytovatelé systému zpráv, kteří jsou dodáni s produktem IBM MQ. Každý z těchto poskytovatelů také poskytuje dvě sady rozšíření rozhraní API systému zpráv. Tyto poskytovatele systému zpráv mohou používat aplikace Java Platform, Standard Edition (Java SE) i Java Platform, Enterprise Edition (Java EE).

V 9.3.0 **JM 3.0** **V 9.3.0** IBM MQ 9.3.0 zavádí podporu pro produkt [Jakarta Messaging 3.0](#). JMS 2.0 je stále plně podporován.

Specifikace JMS a Jakarta Messaging definují sadu rozhraní, která mohou aplikace používat k provádění operací systému zpráv. V produktu IBM MQ 8.0 produkt podporuje verzi JMS 2.0 standardu JMS . Tato implementace nabízí všechny funkce klasického rozhraní API, ale vyžaduje méně rozhraní a je jednodušší ji používat. Další informace viz “JMS a Jakarta Messaging model” na stránce 147 a specifikace JMS 2.0 na [Java.net](#). **JM 3.0** Od IBM MQ 9.3.0 je také podporován Jakarta Messaging .

Balík jakarta.jms (Jakarta Messaging 3.0) nebo javax.jms (JMS 2.0) uvádí podrobnosti rozhraní systému zpráv a poskytovatel systému zpráv implementuje tato rozhraní pro specifický produkt systému zpráv. Příklad:

- IBM MQ classes for JMS je poskytovatel JMS , který implementuje rozhraní JMS pro produkt IBM MQ a také poskytuje následující dvě sady rozšíření pro rozhraní API JMS :
 - IBM MQ JMS rozšíření
 - IBM JMS rozšíření
- Továrnu připojení, frontu nebo objekt tématu vytvořený pomocí rozhraní javax.jms nebo jakarta.jms nebo libovolné sady rozšíření JMS lze adresovat pomocí kteréhokoli z těchto rozhraní API.

To znamená, že je lze přetypovat na libovolné rozhraní. Chcete-li zachovat přenositelnost aplikace na nejvyšší úrovni, použijte nejobecnější rozhraní API, které je vhodné pro vaše požadavky.

Protože JMS a Jakarta Messaging sdílejí mnoho společného, lze další odkazy na JMS v tomto tématu považovat za odkazy na oboje. Případné rozdíly jsou podle potřeby zvýrazněny.

IBM MQ JMS rozšíření

Produkt IBM MQ classes for JMS také poskytuje rozšíření rozhraní API JMS . IBM MQ classes for JMS obsahuje rozšíření, která jsou implementována v objektech MQConnectionFactory, MQQueue a MQTopic. Tyto objekty mají vlastnosti a metody specifické pro IBM MQ. Objekty mohou být spravovány objekty, nebo aplikace může vytvářet objekty dynamicky za běhu. Tato rozšíření se nazývají rozšíření produktu IBM MQ JMS . Všimněte si, že v této dokumentaci nejsou objekty, které jsou vytvářeny dynamicky aplikací za běhu, považovány za spravované objekty.

IBM JMS rozšíření

Kromě rozšíření IBM MQ JMS poskytuje produkt IBM MQ classes for JMS obecnější sadu rozšíření rozhraní JMS API nebo Java jako použitý programovací jazyk. Tato rozšíření se nazývají rozšíření produktu IBM JMS a mají následující obecné cíle:

- Chcete-li zajistit vyšší úroveň konzistence mezi poskytovateli produktu IBM JMS .
- Usnadnit zápis aplikace mostu mezi dvěma systémy systému zpráv IBM .
- Usnadnit portování aplikace z jednoho poskytovatele produktu IBM JMS na jiného.

Hlavní zaměření těchto rozšíření se týká vytváření a konfigurace továren připojení a míst určení dynamicky za běhu, ale tato rozšíření také poskytují funkci, která přímo nesouvisí se systémem zpráv, například funkci pro určování problémů.

Související úlohy

[Použití tříd IBM MQ pro systém zpráv JMS/Jakarta](#)

[Konfigurace prostředků systému zpráv JMS a Jakarta](#)

IBM MQ classes for Jakarta Messaging: přehled

Produkt IBM MQ 9.3.0 zavádí podporu pro produkt Jakarta Messaging. V případě systému Jakarta Messaging 3.0 se řízení specifikace JMS přesunulo z Oracle do procesu komunity Java . Produkt Oracle si však zachovává řízení názvu "javax", který se používá v jiných technologiích Java . Takže i když je Jakarta Messaging 3.0 funkčně ekvivalentní s JMS 2.0, existují určité rozdíly v pojmenování. Oficiální název pro verzi 3.0 je Jakarta Messaging spíše než Java Message Servicea názvy balíků a konstant mají předponu jakarta spíše než javax.

Pozadí

Po mnoho let se platforma Java nabízí ve dvou formách- Standard Edition a Enterprise Edition.

Java Platform, Standard Edition (někdy zkrácený jako Java SE) je základní jazyk a knihovny tříd, které lze spustit v samostatném kontextu. Většina Java balíků v Java SE má názvy začínající na "java."

Funkce Java Platform, Enterprise Edition (Java EE) rozšiřuje tuto funkci o funkce, jako je systém zpráv, různé objekty typu bean, transakce atd. Některé z těchto technologií lze také použít v kontextu Java SE . Většina balíků Java v produktu Java EE má historicky názvy začínající na "javax."-existuje však křížový přechod, takže některé balíky Java SE mají hodnotu "javax." jako předponu k jejich názvu.

Java Message Service (JMS) je součástí Java Platform, Enterprise Edition. Java EE 7 zahrnuje JMS 2.0.

Až do Java EE 7, technologie byly pod dohledem Oracle.

Technologie Java EE se nedávno přesunuly ze správcovství produktu Oracle do komunitního procesu, na který dohlíží nadace Eclipse Foundation.

Jako "oštěp". název nelze přesunout do nového projektu, nové pojmenování bylo přijato-všechny balíky a názvy vlastností jsou nyní opatřeny předponou "jakarta". a soubor Java Platform, Enterprise Edition bude v budoucnu nazván "Jakarta EE". Číslování verzí pokračovalo: Verze 8 byla prozatímní verzí, kterou lze z velké části ignorovat, a Jakarta EE 9 je bod, ve kterém "jakarta". Předpona se projeví.

Hlavní technologie Jakarta EE , která se používá v kontextu IBM MQ , je Jakarta Messaging 3.0 -následník Java Message Service (JMS) 2.0. Takže Jakarta EE 9 zahrnuje Jakarta Messaging 3.0.

IBM MQ nadále podporuje Java EE 7 a JMS 2.0a zavádí podporu pro Jakarta EE 9 a Jakarta Messaging 3.0.

Co je dodáno: Java SE

Pro systém Java Platform, Standard Edition kromě systému IBM MQ classes for JMS (který podporuje operace JMS 2.0 s produktem IBM MQ) IBM MQ 9.3.0 poskytuje IBM MQ classes for Jakarta Messaging. Tyto třídy poskytují poskytovatele Jakarta Messaging 3.0 , který se integruje s produktem IBM MQ, což umožňuje použití správců front IBM MQ k usnadnění operací Jakarta Messaging .

Ty jsou poskytovány jako standardní soubor JAR `com.ibm.mq.jakarta.client.jar` v podadresáři `java/lib` instalace produktu IBM MQ .

Pro použití v kontejnerech OSGi, jako např. Apache Felix nebo Eclipse Equinox, IBM MQ také poskytuje dvojici svazků balíků OSGi:

- `com.ibm.mq.osgi.jms30.clientprereqs_V.R.M.F.jar`
- `com.ibm.mq.osgi.jms30.client_V.R.M.F.jar`

kde *V.R.M.F* představuje verzi produktu IBM MQ, například 9.3.0.0. Tyto balíky lze nalézt v podadresáři `java/lib/OSGi` instalace produktu IBM MQ .

Co je dodáno: Jakarta EE 9

Pro podporu systému zpráv založeného na systému IBM MQ na Jakarta EE 9 kompatibilním aplikačním serveru IBM MQ poskytuje Jakarta EE 9-kompatibilní adaptér prostředků: `wmq.jakarta.jmsra.rar`. To lze nalézt v podadresáři `java/lib/jca` instalace produktu IBM MQ .

Produkt IBM MQ pokračuje v poskytování Java EE 7 kompatibilního adaptéru prostředků `wmq.jmsra.rar` v podadresáři `java/lib/jca` instalace produktu IBM MQ .

Jak jsou tyto artefakty doručovány

Tyto soubory JAR a soubor RAR pro adaptér prostředků jsou zabaleny s již existujícími artefakty na obvyklém instalačním médiu produktu IBM MQ -jak instalační médium specifické pro platformu, například soubory ".rpm", tak redistribuovatelné médium, například samorozbalovací redistribuovatelné soubory JAR klienta.

Co se změnilo mezi JMS 2.0 a Jakarta Messaging 3.0

Jakarta EE 9 a Jakarta Messaging 3.0 nepředstavují žádné nové funkce. Vše, co se změní, jsou názvy. Například, když použijete "javax.jms.Connection" v souboru JMS 2.0, použijete "jakarta.jms.Connection" v souboru Jakarta Messaging 3.0.

Vzhledem k tomu, že produkt Eclipse Foundation postoupí platformu Jakarta EE , bude stavět na této základně a tato konvence pojmenování bude použita pro nové funkce, které budou zavedeny v budoucnu.

Co se změnilo mezi IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging

Souhrn

Produkt IBM MQ classes for JMS, který poskytuje podporu pro produkt JMS 2.0, zůstává k dispozici a doporučuje se především pro údržbu a rozšiřování existujících aplikací. Jsou plně podporovány.

IBM MQ classes for Jakarta Messaging, které poskytují podporu pro produkt Jakarta Messaging 3.0, se doporučují pro nový vývoj.

V produktu IBM MQ 9.3.0 jsou tyto dvě nabídky funkčně ekvivalentní. Liší se pouze názvy. Je však pravděpodobnější, že se v produktu IBM MQ classes for Jakarta Messaging objeví nová funkčnost systému zpráv, než v produktu IBM MQ classes for JMS.

Tyto dvě nabídky jsou interoperabilní. Zprávy vyprodukované produktem IBM MQ classes for JMS mohou být spotřebovány produktem IBM MQ classes for Jakarta Messaginga naopak. Ale tyto dvě nabídky nesmí koexistovat v jediné aplikaci.

Změny pojmenování

<i>Tabulka 16. Změny názvů balíčků</i>	
IBM MQ classes for JMS název sady programů	IBM MQ classes for Jakarta Messaging název sady programů
com.ibm.mq.jms[*]	com.ibm.mq.jakarta.jms[*]
com.ibm.jms	com.ibm.jakarta.jms
com.ibm.msg.client.jms.*	com.ibm.msg.client.jakarta.jms.*
com.ibm.msg.client.wmq.*	com.ibm.msg.client.jakarta.wmq.*

Balíky související se společnými službami (trasování, protokolování, podpora národních jazyků atd.) a implementacemi JMQUI (lokální a vzdálené) jsou společné pro IBM MQ classes for JMS i IBM MQ classes for Jakarta Messaging, takže v těchto oblastech nejsou nutné žádné změny.

Všimněte si, že se také změnila názvy vlastností. Například vlastnost pro povolení rozšíření IBM MQ v souboru IBM MQ classes for Jakarta Messaging je **com.ibm.mq.jakarta.jms.SupportMQExtensions**.

Názvy vlastností, které jsou nezávislé na IBM MQ classes for JMS nebo IBM MQ classes for Jakarta Messaging, jako jsou různé vlastnosti **com.ibm.msg.client.commonservices.trace.***, platí stejně pro obě nabídky.

Administrativní obslužné programy

Obslužné programy **crtmqenv** a **setmqenv** nyní přijímají volbu, která určuje, zda má být cesta ke třídě konfigurována pro IBM MQ classes for JMS (-j 2.0) nebo IBM MQ classes for Jakarta Messaging (-j 3.0), a existují varianty IBM MQ classes for Jakarta Messaging obslužných programů **runjms** s názvem **runjms30** a podobnými názvy.

Obslužný program **dspmqver**, je-li požádán o hlášení o komponentách Java, zahrne do svého výstupu IBM MQ classes for Jakarta Messaging.

Chcete-li konfigurovat objekty IBM MQ classes for Jakarta Messaging, které mají být načteny prostřednictvím rozhraní JNDI, je nový obslužný program **JMS30Admin** ekvivalentní obslužnému programu **JMSAdmin** pro systém IBM MQ classes for JMS.

Všimněte si, že základní objekty jsou z různých balíčků. Definice rozhraní JNDI vytvořené produktem **JMSAdmin** nemohou být použity produktem IBM MQ classes for Jakarta Messaging ani ty, které vytvořil produkt **JMS30Admin**, nemohou být použity produktem IBM MQ classes for JMS.

Poznámka: Neexistuje žádná podpora pro objekty IBM MQ classes for Jakarta Messaging poskytované produktem IBM MQ Explorer; jejich integrace JNDI je pouze pro IBM MQ classes for JMS.

Související pojmy

[Proč bych měl používat třídy IBM MQ pro systém zpráv Jakarta?](#)

JMS a Jakarta Messaging model

Modely JMS a Jakarta Messaging definují sadu rozhraní, která mohou aplikace Java používat k provádění operací systému zpráv. IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging jsou

poskytovatelé systému zpráv, kteří definují, jak Java objekty systému zpráv souvisí s koncepty IBM MQ . Specifikace JMS a Jakarta Messaging očekávají, že určité objekty systému zpráv budou spravovány.

V produktu IBM MQ 8.0 produkt podporuje verzi JMS 2.0 standardu JMS, který zavedl zjednodušené rozhraní API a zároveň zachoval klasické rozhraní API z produktu JMS 1.1.

V 9.3.0 **JM 3.0** **V 9.3.0** IBM MQ 9.3.0 zavádí podporu pro produkt [Jakarta Messaging 3.0](#). JMS 2.0 je stále plně podporován. Protože JMS a Jakarta Messaging sdílejí mnoho společného, lze další odkazy na JMS v tomto tématu považovat za odkazy na oboje. Případné rozdíly jsou podle potřeby zvýrazněny.

Zjednodušené rozhraní API

Produkt JMS 2.0 zavedl zjednodušené rozhraní API a zároveň zachoval rozhraní specifická pro doménu a nezávislá na doméně z produktu JMS 1.1. Zjednodušené rozhraní API snižuje počet objektů, které jsou potřebné pro odesílání a příjem zpráv, a skládá se z následujících rozhraní:

ConnectionFactory

ConnectionFactory je spravovaný objekt, který používá klient JMS k vytvoření připojení. Toto rozhraní se také používá v klasickém rozhraní API.

JMSKontext

Tento objekt kombinuje objekty připojení a relace klasického rozhraní API. JMSKontextové objekty lze vytvářet z jiných objektů JMSKontext, přičemž základní připojení je duplikováno.

JMSProducent

Producent JMS je vytvořen kontextem JMS a používá se k odesílání zpráv do fronty nebo tématu. Objekt Producent JMS způsobí vytvoření objektů, které jsou nezbytné pro odeslání zprávy.

JMSSpotřebitel

Konzument JMS je vytvořen kontextem JMS a používá se k přijímání zpráv z tématu nebo fronty.

Zjednodušené rozhraní API má řadu efektů:

- Objekt kontextu JMS vždy automaticky spustí základní připojení.
- JMSProducenti a JMSSpotřebitelé mohou nyní pracovat přímo s těly zpráv, aniž by museli získat celý objekt zprávy, pomocí metody `getBody` zprávy.
- Vlastnosti zprávy lze nastavit na objektu Producent JMS pomocí řetězení metod před odesláním 'těla' obsahu zprávy. Producent JMS zpracuje vytvoření všech objektů, které jsou potřebné k odeslání zprávy. Pomocí funkce JMS 2.0 lze nastavit vlastnosti a odeslat zprávu následujícím způsobem:

```
context.createProducer().
setProperty("foo", "bar").
setTimeToLive(10000).
setDeliveryMode(NON_PERSISTENT).
setDisableMessageTimestamp(true).
send(dataQueue, body);
```

Produkt JMS 2.0 také zavedl sdílené odběry, ve kterých mohou být zprávy sdíleny mezi více spotřebiteli. Se všemi odběry produktu JMS 1.1 se zachází jako s nesdílenými odběry.

Klasické rozhraní API

Následující seznam shrnuje hlavní rozhraní JMS klasického rozhraní API:

Místo určení

Místo určení je místo, kam aplikace odesílá zprávy, nebo je to zdroj, ze kterého aplikace přijímá zprávy, nebo obojí.

ConnectionFactory

Objekt ConnectionFactory zapouzdřuje sadu vlastností konfigurace pro připojení. Aplikace používá továrnu připojení k vytvoření připojení.

Připojení

Objekt připojení zapouzdřuje aktivní připojení aplikace k serveru systému zpráv. Aplikace používá připojení k vytváření relací.

Relace

Relace je kontext s jedním podprocesem pro odesílání a příjem zpráv. Aplikace používá relaci k vytváření zpráv, producentů zpráv a spotřebitelů zpráv. Relace je buď transakční, nebo netransakční.

Zpráva

Objekt zprávy zapouzdřuje zprávu, kterou aplikace odesílá nebo přijímá.

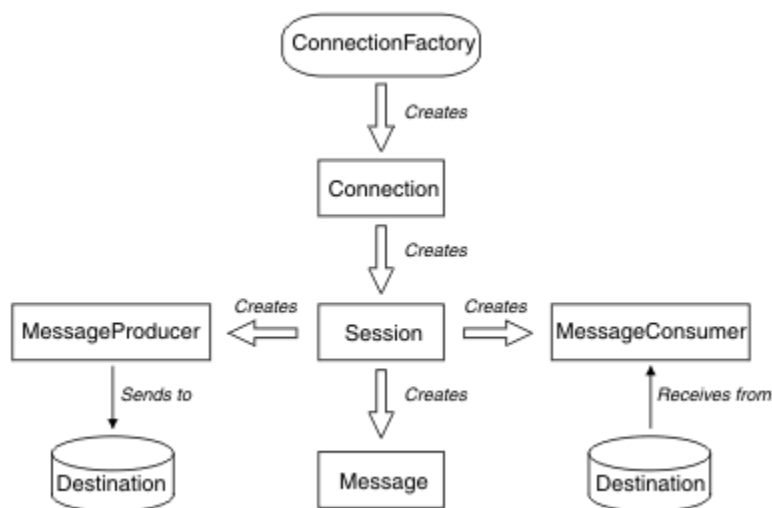
MessageProducer

Aplikace používá producenta zpráv k odesílání zpráv do místa určení.

MessageConsumer

Aplikace používá spotřebitele zpráv k přijímání zpráv odeslaných do místa určení.

Obrázek 53 na stránce 149 zobrazuje tyto objekty a jejich vztahy.



Obrázek 53. Objekty JMS a jejich vztahy

Diagram zobrazuje hlavní rozhraní: ConnectionFactory, Connection, Session, MessageProducer, MessageConsumer, Message a Destination. Aplikace používá továrnu připojení k vytvoření připojení a používá připojení k vytvoření relací. Aplikace pak může pomocí relace vytvářet zprávy, producenty zpráv a spotřebitele zpráv. Aplikace používá producenta zpráv k odesílání zpráv do místa určení a spotřebitele zpráv používá k přijímání zpráv odeslaných do místa určení.

Objekt Destination, ConnectionFactory nebo Connection může být souběžně používán různými podprocesy aplikace s podporou podprocesů, ale objekty Session, MessageProducer nebo MessageConsumer nemohou být souběžně používány různými podprocesy. Nejjednodušším způsobem, jak zajistit, aby se objekt Session, MessageProducer nebo MessageConsumer souběžně nepoužíval, je vytvořit pro každý podproces samostatný objekt relace.

Produkt JMS podporuje dva styly systému zpráv:

- Dvoubodový systém zpráv
- Systém zpráv publikování/odběru

Tyto styly systému zpráv jsou také označovány jako *domény systému zpráva* v aplikaci můžete kombinovat oba styly systému zpráv. V doméně typu point-to-point je cílem fronta a v doméně publikování/odběru je cílem téma.

U verzí produktu JMS před JMS 1.1 programování pro doménu typu point-to-point používá jednu sadu rozhraní a metod a programování pro doménu publikování/odběru používá jinou sadu. Tyto dvě sady jsou podobné, ale oddělené. V produktu JMS 1.1 můžete použít společnou sadu rozhraní a metod, které podporují obě domény systému zpráv. Společná rozhraní poskytují zobrazení každé domény systému

zpráv nezávislé na doméně. Tabulka 17 na stránce 150 vypíše JMS rozhraní nezávislá na doméně a jejich odpovídající rozhraní specifická pro doménu.

Tabulka 17. Rozhraní nezávislá na doméně JMS a odpovídající rozhraní specifická pro doménu

Rozhraní nezávislá na doméně	Rozhraní specifická pro doménu pro doménu typu point-to-point	Specifická rozhraní domény pro doménu publikování/odběru
ConnectionFactory	Továrna QueueConnection	Továrna TopicConnection
Připojení	QueueConnection	TopicConnection
Místo určení	Fronta	Téma
Relace	QueueSession	TopicSession
MessageProducer	QueueSender	TopicPublisher
MessageConsumer	QueueReceiver QueueBrowser	TopicSubscriber

JMS 2.0 Produkt IBM MQ classes for JMS 2.0 podporuje jak dřívější rozhraní specifická pro doménu JMS 1.1, tak zjednodušené rozhraní API produktu JMS 2.0. Produkt IBM MQ classes for JMS 2.0 lze proto použít pro údržbu existujících aplikací, včetně vývoje nových funkcí ve stávajících aplikacích.

JM 3.0 Produkt IBM MQ classes for Jakarta Messaging 3.0 podporuje verze Jakarta Messaging stejných rozhraní a doporučuje se pro vývoj nových aplikací.

V systémech IBM MQ classes for JMS a IBM MQ classes for Jakarta Messagingse objekty JMS vztahují k konceptům IBM MQ následujícími způsoby:

- Objekt připojení má vlastnosti odvozené od vlastností továrny připojení, která byla použita k vytvoření připojení. Tyto vlastnosti řídí způsob připojení aplikace ke správci front. Příklady těchto vlastností jsou název správce front a v případě aplikace, která se připojuje ke správci front v režimu klienta, název hostitele nebo adresa IP systému, v němž je správce front spuštěn.
- Objekt relace zapouzdřuje manipulátor připojení IBM MQ, který proto definuje transakční rozsah relace.
- Objekt MessageProducer a objekt MessageConsumer zapouzdřují popisovač objektu IBM MQ.

Při použití IBM MQ classes for JMS nebo IBM MQ classes for Jakarta Messagingplatí všechna normální pravidla IBM MQ. Všimněte si zejména, že aplikace může odeslat zprávu do vzdálené fronty, ale může přijmout zprávu pouze z fronty vlastněné správcem front, ke kterému je aplikace připojena.

Specifikace JMS očekává, že objekty ConnectionFactory a Destination budou spravovány. Administrátor vytvoří a udržuje spravované objekty v centrálním úložišti a aplikace JMS načte tyto objekty pomocí rozhraní JNDI (Java Naming and Directory Interface).

V produktu IBM MQ classes for JMS a IBM MQ classes for Jakarta Messagingje implementace rozhraní Destination abstraktní supertřída Queue a Topic, takže instance Destination je buď objekt Queue, nebo objekt Topic. Rozhraní nezávislá na doméně považují frontu nebo téma za cíl. Doména systému zpráv pro objekt MessageProducer nebo MessageConsumer je určena podle toho, zda je cílem fronta nebo téma.

V operačním systému IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging proto mohou být objekty následujících typů spravovány:

- ConnectionFactory
- Továrna QueueConnection
- Továrna TopicConnection
- Fronta
- Téma
- XAConnectionFactory

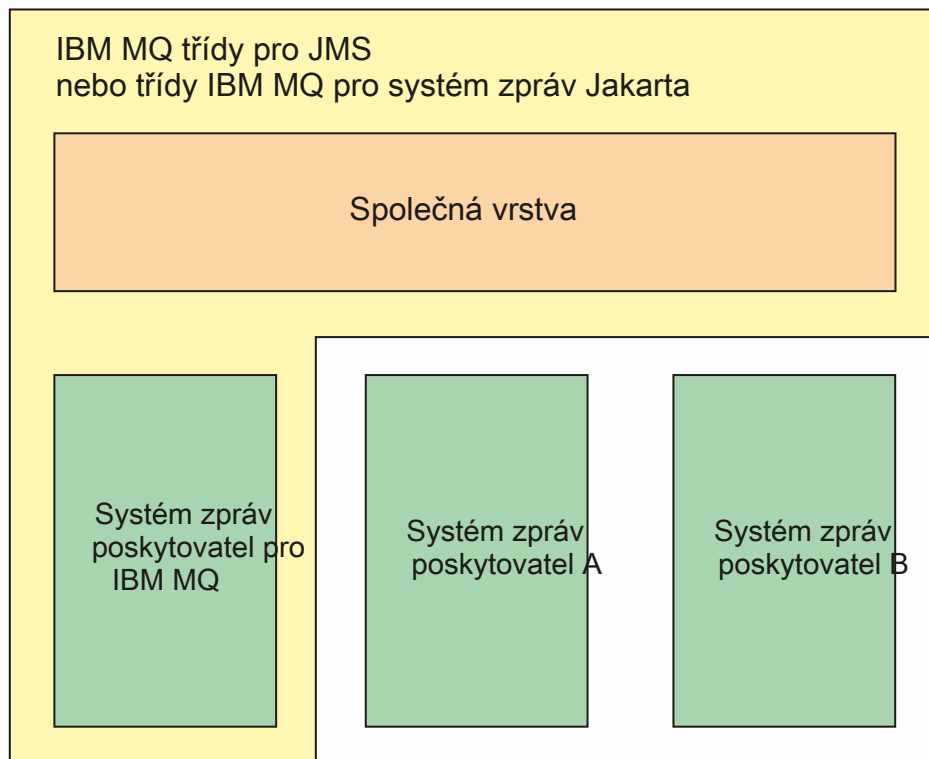
- Továrna XAQueueConnection
- Továrna XATopicConnection

IBM MQ classes for JMS/Jakarta Messaging architektura

IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging mají vrstvenou architekturu. Nejvyšší vrstva kódu je společná vrstva, kterou může používat kterýkoli poskytovatel systému zpráv produktu IBM Java .

V 9.3.0 **JM 3.0** **V 9.3.0** IBM MQ 9.3.0 zavádí podporu pro produkt Jakarta Messaging 3.0. JMS 2.0 je stále plně podporován.

IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging mají vrstvenou architekturu, jak ukazuje diagram Obrázek 54 na stránce 151. Nejvyšší vrstva kódu je společná vrstva, kterou může použít libovolný poskytovatel produktu IBM JMS nebo Jakarta Messaging. Když aplikace volá metodu JMS nebo Jakarta Messaging, jakékoli zpracování volání, které není specifické pro systém zpráv, provede společná vrstva, která také poskytuje konzistentní odezvu na volání. Jakékoli zpracování volání, které je specifické pro systém zpráv, je delegováno na nižší vrstvu. V následujícím diagramu je poskytovatel systému zpráv IBM MQ zobrazen v dolní vrstvě spolu se dvěma dalšími poskytovateli systému zpráv (poskytovatel systému zpráv A a poskytovatel systému zpráv B.)



Obrázek 54. Vrstvená architektura pro poskytovatele IBM JMS a Jakarta Messaging

Vrstvená architektura plní následující cíle:

- Chcete-li zlepšit konzistenci chování různých poskytovatelů IBM JMS a Jakarta Messaging .
- Usnadnění zápisu aplikace mostu mezi dvěma systémy systému zpráv IBM
- Usnadnění portování aplikace z jednoho poskytovatele produktu IBM JMS nebo Jakarta Messaging na jiného poskytovatele.

Související úlohy

Použití tříd IBM MQ pro systém zpráv JMS/Jakarta

Podpora pro spravované objekty

IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging podporují použití spravovaných objektů.

V 9.3.0 **JM 3.0** **V 9.3.0** Od IBM MQ 9.3.0, Jakarta Messaging 3.0 je podporován pro vývoj nových aplikací. Produkt IBM MQ 9.3.0 nadále podporuje produkt JMS 2.0 pro existující aplikace. Použití rozhraní API Jakarta Messaging 3.0 a rozhraní API JMS 2.0 ve stejné aplikaci není podporováno. Další informace naleznete v tématu [Použití tříd IBM MQ pro systém zpráv JMS/Jakarta](#).

Tok logiky v rámci aplikace JMS nebo IBM MQ classes for Jakarta Messaging začíná objektem `ConnectionFactory` a cílovými objekty. Aplikace používá objekt `ConnectionFactory` k vytvoření objektu připojení, který představuje aktivní připojení z aplikace k serveru systému zpráv. Aplikace používá objekt připojení k vytvoření objektu relace, což je kontext s jedním podprocesem pro vytváření a příjem zpráv. Aplikace pak může pomocí objektu relace a cílového objektu vytvořit objekt `MessageProducer`, který aplikace používá k odesílání zpráv do určeného místa určení. Cíl je buď fronta, nebo téma v systému zpráv a je zapouzdřen objektem `Destination`. Aplikace může také použít objekt `Session` a objekt `Destination` k vytvoření objektu `MessageConsumer`, který aplikace používá k přijímání zpráv, které byly odeslány do určeného místa určení.

Specifikace JMS a Jakarta Messaging očekávají, že objekty `ConnectionFactory` a objekty `Destination` budou spravovány. Administrátor vytvoří a udržuje spravované objekty v centrálním úložišti a aplikace JMS nebo Jakarta Messaging tyto objekty načte pomocí Java Naming Directory Interface (JNDI). Úložiště spravovaných objektů může být v rozsahu od jednoduchého souboru až po adresář LDAP (Lightweight Directory Access Protocol).

IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging podporují použití spravovaných objektů. Aplikace může používat všechny funkce produktu IBM MQ classes for JMS nebo IBM MQ classes for Jakarta Messaging, které jsou vystaveny prostřednictvím produktu IBM MQ, aniž by měla jakékoli informace specifické pro produkt IBM MQ pevně zakódované do samotné aplikace. Toto uspořádání poskytuje aplikaci určitý stupeň nezávislosti na základní konfiguraci produktu IBM MQ.

K dosažení této nezávislosti může aplikace použít rozhraní JNDI k načtení továren připojení a míst určení, která jsou uložena jako spravované objekty, a k provádění operací systému zpráv používat pouze rozhraní definovaná v balíku `javax.jms` (JMS 2.0) nebo `jakarta.jms` (Jakarta Messaging 3.0).

JMS 2.0 V případě systému JMS 2.0 může administrátor použít nástroj pro administraci IBM MQ JMS **JMSAdmin** nebo IBM MQ Explorer k vytvoření a údržbě spravovaných objektů v centrálním úložišti.

JM 3.0 Pro Jakarta Messaging 3.0 můžete spravovat rozhraní JNDI pomocí IBM MQ Explorer. Administrace rozhraní JNDI je podporována Jakarta Messaging 3.0 variantou **JMSAdmin**, což je **JMS30Admin**.

Aplikační server obvykle poskytuje vlastní úložiště pro spravované objekty a vlastní nástroje pro vytváření a údržbu objektů. Aplikace Java EE **JM 3.0** nebo Jakarta EE proto může použít JNDI k načtení spravovaných objektů buď z úložiště aplikačního serveru, nebo z centrálního úložiště.

Související úlohy

[Konfigurace prostředků systému zpráv JMS a Jakarta](#)

Podporované typy komunikace na platformách Java EE a Jakarta EE

Na platformách Java EE a Jakarta EE podporují systémy IBM MQ classes for JMS a IBM MQ classes for Jakarta Messaging dva typy komunikace mezi komponentou aplikace a správcem front IBM MQ.

V 9.3.0 **JM 3.0** **V 9.3.0** IBM MQ 9.3.0 zavádí podporu pro produkt [Jakarta Messaging 3.0](#). JMS 2.0 je stále plně podporován. Protože JMS a Jakarta Messaging sdílejí mnoho společného, lze další odkazy na JMS v tomto tématu považovat za odkazy na oboje. Případné rozdíly jsou podle potřeby zvýrazněny.

Podporovány jsou následující dva typy komunikace mezi komponentou aplikace a správcem front IBM MQ :

- Odchozí komunikace
- Příchozí komunikace

Odchozí komunikace

Pomocí rozhraní API JMS nebo Jakarta Messaging přímo vytvoří aplikační komponenta připojení ke správci front a poté odešle a přijme zprávy.

Aplikační komponentou může být například aplikační klient, servlet, stránka Java Server Page (JSP), objekt typu enterprise Java bean (EJB) nebo objekt typu message-driven bean (MDB). V tomto typu komunikace poskytuje kontejner aplikačního serveru pouze nízkouúrovňové funkce pro podporu operací systému zpráv, například sdružování připojení a správu podprocesů.

Příchozí komunikace

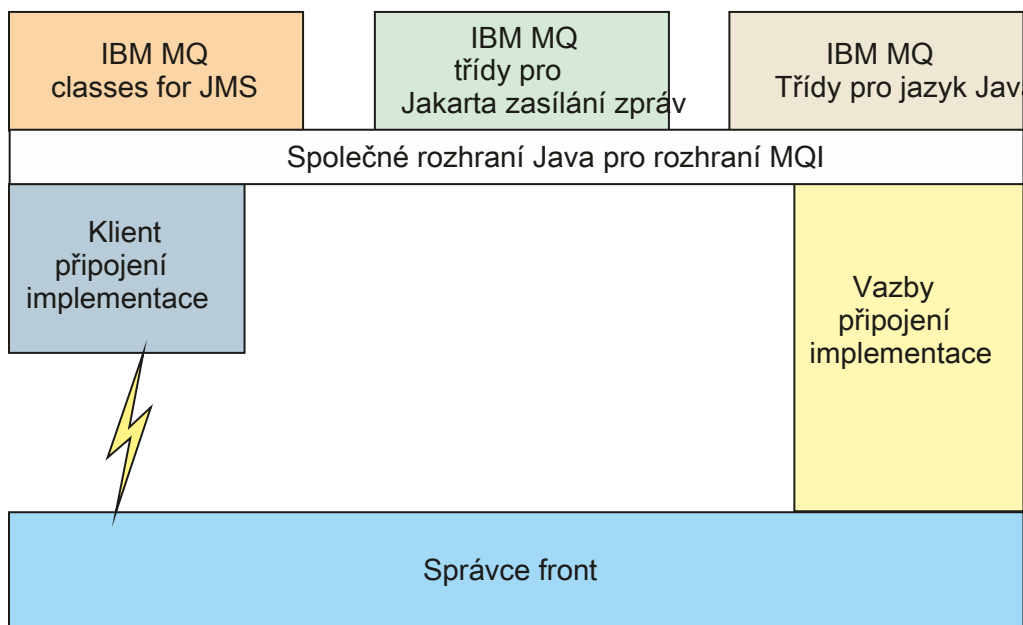
V případě příchozí komunikace je zpráva přicházející do místa určení doručena do objektu MDB, který zprávu poté zpracuje.

Aplikace Java EE **JM 3.0** a Jakarta EE používají objekty MDB k asynchronnímu zpracování zpráv. Objekt MDB vystupuje jako modul listener pro zprávy JMS a je implementován metodou `onMessage()`, která definuje způsob zpracování zprávy. Objekt MDB je implementován v kontejneru EJB aplikačního serveru. Přesný způsob konfigurace objektu typu message-driven bean závisí na tom, který aplikační server používáte, ale informace o konfiguraci musí určovat, ke kterému správci front se má připojit, jak se má připojit ke správci front, který cíl pro monitorování zpráv a transakční chování objektu typu message-driven bean. Tyto informace pak použije kontejner EJB. Pokud zpráva splňující kritéria výběru objektu MDB dorazí do určeného místa určení, kontejner EJB použije k načtení zprávy ze správce front IBM MQ classes for JMS nebo IBM MQ classes for Jakarta Messaging a poté zprávu doručí do objektu MDB voláním příslušné metody `onMessage()`.

Relace s IBM MQ classes for Java

IBM MQ classes for Java, IBM MQ classes for Jakarta Messaging a IBM MQ classes for JMS jsou peery, které používají společné rozhraní Java pro rozhraní MQI.

Obrázek 55 na stránce 153 ukazuje vztah mezi IBM MQ classes for JMS, IBM MQ classes for Jakarta Messaging a IBM MQ classes for Java.



Obrázek 55. Vztah mezi IBM MQ classes for JMS, IBM MQ classes for Jakarta Messaging a IBM MQ classes for Java

Obecně platí, že programy Java by měly používat pouze jedno rozhraní pro rozhraní s IBM MQ - IBM MQ classes for Java, IBM MQ classes for Jakarta Messaging nebo IBM MQ classes for JMS. Míchání rozhraní není podporováno, s jednou výjimkou. Chcete-li zachovat kompatibilitu s verzemi před produktem IBM WebSphere MQ 7.0, mohou třídy uživatelských procedur kanálu napsané v produktu Java nadále používat rozhraní IBM MQ classes for Java , a to i v případě, že jsou třídy uživatelských procedur kanálu volány z produktu IBM MQ classes for JMS. Avšak použití rozhraní IBM MQ classes for Java znamená, že vaše aplikace jsou stále závislé na:

- ▶ **JMS 2.0** Soubor IBM MQ classes for Java JAR, `com.ibm.mq.jar`. Pokud ve své cestě ke třídám nechcete `com.ibm.mq.jar`, můžete místo toho použít sadu rozhraní v balíku `com.ibm.mq.exits`.
- ▶ **V 9.3.0** ▶ **JM 3.0** ▶ **V 9.3.0** Použití konzoly `com.ibm.mq.jakarta.client.jar` při spolupráci s produktem IBM MQ classes for Jakarta Messaging.

Související pojmy

▶ **V 9.3.0** ▶ **V 9.3.0** [Proč bych měl používat třídy IBM MQ pro systém zpráv Jakarta?](#)

[Proč bych měl používat třídy IBM MQ pro platformu JMS?](#)

[Proč bych měl používat třídy IBM MQ pro Java?](#)

IBM MQ Poskytovatel systému zpráv

Poskytovatel systému zpráv IBM MQ má tři provozní režimy: normální režim, normální režim s omezeními a režim migrace.

Poskytovatel systému zpráv IBM MQ má tři provozní režimy:

- Normální režim poskytovatele systému zpráv IBM MQ
- IBM MQ s omezeními
- Režim migrace poskytovatele systému zpráv IBM MQ

Normální režim poskytovatele systému zpráv IBM MQ používá k implementaci všechny funkce IBM MQ správce front JMS. Tento režim je optimalizován pro použití rozhraní API a funkcí JMS

2.0 ▶ **V 9.3.0** ▶ **JM 3.0** ▶ **V 9.3.0** nebo [Jakarta Messaging 3.0](#).

Pokud:

- Klient uvádí verzi poskytovatele 6 na serveru **ConnectionFactory**, klient se chová způsobem, který je kompatibilní s klientem poskytovaným s produktem IBM WebSphere MQ 6.0. Jsou podporována pouze rozhraní JMS 1.1 a JMS 2, ale některé funkce JMS 2, jako například sdílené odběry, prodleva doručení a asynchronní odeslání, jsou zakázány. Neexistuje žádné sdílení připojení.
- Klient uvádí verzi poskytovatele 7 na systému **ConnectionFactory**, rozhraní JMS 1.1 i rozhraní JMS 2 jsou plně podporována.
- Není určena žádná verze poskytovatele, dojde k pokusu o připojení k poskytovateli verze 7. Pokud se to nezdaří, provede se další pokus s poskytovatelem verze 6.

Chcete-li se připojit k produktu IBM Integration Bus pomocí produktu IBM MQ Enterprise Transport, použijte režim migrace. Používáte-li přenos v reálném čase IBM MQ, je režim migrace automaticky vybrán, protože jste explicitně vybrali vlastnosti v objektu továrny připojení. Připojení k produktu IBM Integration Bus pomocí přenosu IBM MQ Enterprise Transport se řídí obecnými pravidly pro výběr režimu, která jsou popsána v tématu [Konfigurace vlastnosti JMS PROVIDERVERSION](#).

Související úlohy

[Konfigurace prostředků JMS](#)

▶ z/OS IBM MQ for z/OS koncepce

Některé koncepty používané produktem IBM MQ for z/OS jsou jedinečné pro platformu z/OS. Například mechanismus protokolování, metody správy úložiště, dispozice jednotky zotavení a skupiny sdílení front

jsou poskytovány pouze s produktem IBM MQ for z/OS. Toto téma použijte jako úvod k dalším informacím o těchto koncepcích.

Koncepce zahrnují přehled objektů, které produkt IBM MQ for z/OS používá, včetně:

- Správce front
- Inicializátor kanálu
- Sdílené fronty a skupiny sdílení front
- Použití front v rámci skupiny

Následující témata také zahrnují různé postupy, které potřebujete, včetně:

- Definice systému na systému z/OS
- Správa úložiště
- Obnova a restart
- Koncepce zabezpečení v produktu IBM MQ for z/OS

Související pojmy

[“Správce front v systému z/OS” na stránce 156](#)

Než budete moci nechat aplikační programy používat produkt IBM MQ ve vašem systému z/OS, musíte nainstalovat produkt IBM MQ for z/OS a spustit správce front. Správce front vlastní a spravuje sadu prostředků používaných produktem IBM MQ.

[“Inicializátor kanálu v systému z/OS” na stránce 157](#)

Inicializátor kanálu poskytuje a spravuje prostředky, které umožňují distribuované řazení do front systému IBM MQ. Produkt IBM MQ používá k odesílání zpráv z jednoho správce front do jiného agenta MCA (*Message Channel Agents*).

[“Podmínky a úlohy pro správu produktu IBM MQ for z/OS” na stránce 159](#)

Toto téma použijte jako úvod do terminologie a úloh, které jsou specifické pro IBM MQ for z/OS.

[“Sdílené fronty a skupiny sdílení front” na stránce 161](#)

K implementaci vysoké dostupnosti prostředků IBM MQ můžete použít sdílené fronty a skupiny sdílení front. Sdílené fronty a skupiny sdílení front jsou funkce jedinečné pro produkt IBM MQ for z/OS na platformě z/OS.

[“Použití front v rámci skupiny” na stránce 205](#)

Tento oddíl popisuje řazení do front v rámci skupiny, což je funkce IBM MQ for z/OS jedinečná pro platformu z/OS. Tato funkce je k dispozici pouze pro správce front definované pro skupinu sdílení front.

[“Správa úložiště na systému z/OS” na stránce 218](#)

Produkt IBM MQ for z/OS vyžaduje trvalé a dočasné datové struktury a k uložení těchto dat používá sady stránek a vyrovnávací paměti. Tato témata poskytují další podrobnosti o tom, jak produkt IBM MQ používá tyto sady stránek a vyrovnávací paměti.

[“přihlášení IBM MQ for z/OS” na stránce 222](#)

IBM MQ udržuje *protokoly* změn dat a významných událostí, jak k nim dochází. Tyto protokoly lze v případě potřeby použít k obnovení dat do předchozího stavu.

[“Obnova a restart na systému z/OS” na stránce 244](#)

Pomocí odkazů v tomto tématu zjistíte informace o funkcích produktu IBM MQ for z/OS pro restart a obnovu.

[“Koncepce zabezpečení v produktu IBM MQ for z/OS” na stránce 260](#)

Prostřednictvím tohoto tématu můžete porozumět důležitosti zabezpečení produktu IBM MQ a důsledkům neexistence odpovídajícího nastavení zabezpečení ve vašem systému.

[“Dostupnost na z/OS” na stránce 267](#)

Produkt IBM MQ for z/OS má mnoho funkcí pro vysokou dostupnost. Toto téma popisuje některé aspekty dostupnosti.

[“Dispozice jednotky zotavení na systému z/OS” na stránce 271](#)

Určité transakční aplikace mohou při připojení ke správci front ve skupině sdílení front (QSG) namísto správce front QMGR používat dispoziční jednotku zotavení GROUP, a to zadáním názvu skupiny sdílení

front při připojení namísto názvu správce front. To umožňuje, aby zotavení transakcí bylo flexibilnější a robustnější, a to odebráním požadavku na opětovné připojení ke stejnému správci front v rámci skupiny sdílení front.

Související odkazy

[“Definice systému v systému z/OS” na stránce 233](#)

Produkt IBM MQ for z/OS používá mnoho výchozích definic objektů a poskytuje ukázkový soubor JCL k vytvoření těchto výchozích objektů. Toto téma slouží k pochopení těchto výchozích objektů a ukázkového kódu JCL.

[“Monitorování a statistiky na IBM MQ for z/OS” na stránce 270](#)

Produkt IBM MQ for z/OS má sadu prostředků pro monitorování správce front a shromažďování statistických údajů.

z/OS

Správce front v systému z/OS

Než budete moci nechat aplikační programy používat produkt IBM MQ ve vašem systému z/OS, musíte nainstalovat produkt IBM MQ for z/OS a spustit správce front. Správce front vlastní a spravuje sadu prostředků používaných produktem IBM MQ.

Správce front

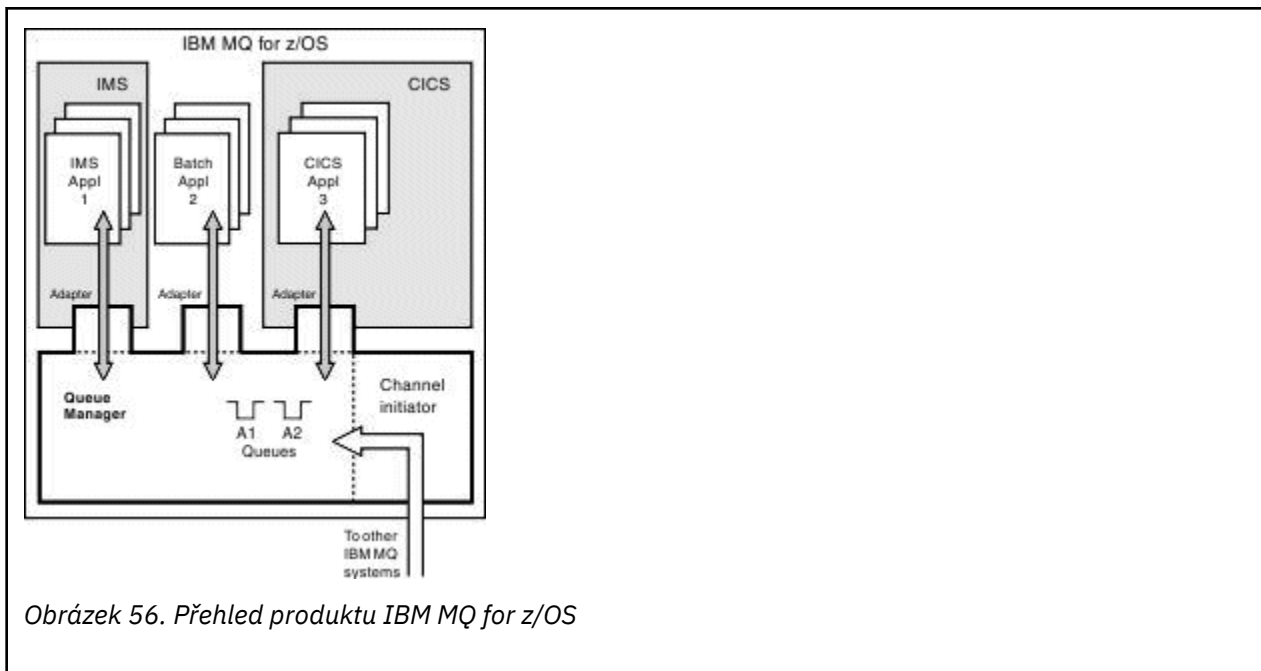
Správce front je program, který poskytuje služby systému zpráv aplikacím. Aplikace, které používají rozhraní MQI (Message Queue Interface), mohou vkládat zprávy do front a získávat je z nich. Správce front zajišťuje, aby byly zprávy odeslány do správné fronty nebo směrovány do jiného správce front. Ve správci front jsou zpracována volání předaná z rozhraní MQI i příkazy odeslané z libovolného zdroje. Správce front generuje pro každé volání nebo příkaz příslušné kódy dokončení.

Mezi prostředky spravované správcem front patří:

- Sady stránek, které zadržují definice objektů IBM MQ a data zpráv
- Protokoly, které se používají k obnovení zpráv a objektů v případě selhání správce front
- Úložiště procesoru
- Připojení, jejichž prostřednictvím mohou různá aplikační prostředí (CICS, IMSa Batch) přistupovat k rozhraní API produktu IBM MQ.
- Inicializátor kanálu IBM MQ, který umožňuje komunikaci mezi produktem IBM MQ v systému z/OS a ostatními systémy.

Správce front má název a aplikace se k němu mohou připojit pomocí tohoto názvu.

[Obrázek 56 na stránce 157](#) ilustruje správce front, zobrazuje připojení k různým prostředím aplikací a iniciátor kanálu.



Subsystém správce front v systému z/OS

V systému z/OS se systém IBM MQ spouští jako subsystém z/OS, který je spuštěn v době IPL.

V subsystému je správce front spuštěn provedením procedury JCL, která určuje datové sady z/OS obsahující informace o protokolech a obsahující definice objektů a data zpráv (sady stránek). Subsystém a správce front mají stejný název, maximálně čtyři znaky. Všichni správci front ve vaší síti musí mít jedinečné názvy, a to i v případě, že se nacházejí v různých systémech, prostředí sysplex nebo platformách.

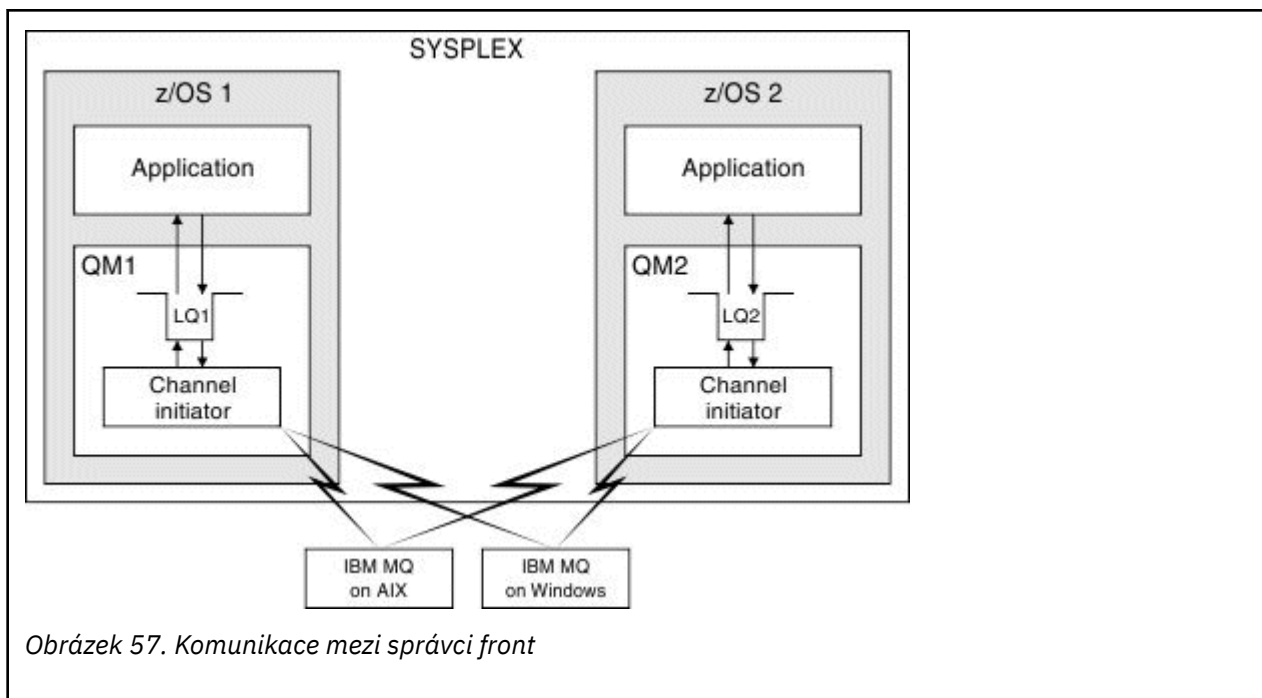
z/OS Inicializátor kanálu v systému z/OS

Inicializátor kanálu poskytuje a spravuje prostředky, které umožňují distribuované řazení do front systému IBM MQ. Produkt IBM MQ používá k odesílání zpráv z jednoho správce front do jiného agenta MCA (*Message Channel Agents*).

Chcete-li odesílat zprávy ze správce front A do správce front B, *odesílající* adaptér MCA ve správci front A musí nastavit komunikační odkaz na správce front B. Aby bylo možné přijímat zprávy z komunikačního propojení, musí být ve správci front B spuštěn modul MCA *přijímající*. Tato jednosměrná cesta sestávající z odesílajícího agenta MCA, komunikačního spoje a přijímajícího agenta MCA se nazývá *kanál*. Odesílající agent MCA převezme zprávy z přenosové fronty a odešle je do přijímajícího agenta MCA. Přijímající agent MCA přijímá zprávy a vkládá je do cílových front.

V systému IBM MQ for z/OS se všechny odesílající a přijímající MCA spouštějí uvnitř inicializátoru kanálu (inicializátor kanálu je také znám jako *přesouvač*). Inicializátor kanálu je spuštěn jako adresní prostor z/OS pod kontrolou správce front. Ke správci front může být připojen pouze jeden inicializátor kanálu, který je spuštěn ve stejném obrazu z/OS jako správce front. V inicializátoru kanálu mohou být souběžně spuštěny tisíce procesů MCA.

Obrázek 57 na stránce 158 zobrazuje dva správce front v prostředí sysplex. Každý správce front má inicializátor kanálu a lokální frontu. Zprávy odeslané správci front v systému AIX a Windows jsou umístěny do lokální fronty, odkud jsou načteny aplikací. Zprávy odpovědí jsou vráceny podobnou přenosovou cestou.



Iniciátor kanálu také obsahuje další procesy týkající se správy kanálů. Mezi tyto procesy patří:

Moduly listener

Tyto procesy naslouchají příchozím požadavkům kanálu v komunikačním subsystému, jako je například protokol TCP, a při přijetí příchozího požadavku spustí pojmenovaného agenta MCA.

Supervizor

To spravuje adresní prostor inicializátoru kanálu, například je zodpovědný za restartování kanálů po selhání.

Server názvů

Používá se k rozlišení názvů TCP na adresy.

Úlohy TLS

Používají se k šifrování a dešifrování a kontrole seznamů odvolaných certifikátů.

z/OS Záznamy SMF pro inicializátor kanálu

Iniciátor kanálu (CHINIT) může vytvářet statistické záznamy SMF a evidenční záznamy s informacemi o úlohách a kanálech.

CHINIT může vytvářet statistické záznamy SMF a účetní záznamy s následujícími typy informací:

- Úlohy: dispečer, adaptér, server DNS (Domain Name Server) a SSL. Tyto úlohy tvoří tzv. statistiku CHINIT.
- Kanály: poskytuje účetní informace podobné těm, které jsou k dispozici v příkazu DIS CHSTATUS. Tomu se říká evidence kanálů.

Produkt IBM MQ for Multiplatforms poskytuje podobné informace zápisem zpráv PCF do systému SYSTEM.ADMIN.STATISTICS.QUEUE. Další informace o zaznamenávání statistických informací v systému IBM MQ for Multiplatforms naleznete v tématu [Data zpráv statistiky kanálu](#).

Data statistiky

Tyto informace můžete použít k vyhledání následujících informací:

- Zda potřebujete více úloh CHINIT, jako např. počet TCB SSL a kolik CPU používají tyto úlohy.
- Průměrná doba požadavků na tyto úlohy.

- Nejdelší doba trvání požadavku v intervalu a denní doba, kdy k tomu došlo, pro úlohy DNS a SSL. Tuto denní dobu můžete korelovat s problémy, které se mohou vyskytnout s kanálem.

Účetní data

Tyto informace můžete použít k monitorování využití kanálů a k vyhledání následujících informací:

- Kanály s nejvyšší propustností.
- Rychlost, jakou byly zprávy odeslány, a rychlost odesílání dat v megabajtech za sekundu.
- Dosažená velikost dávky. Pokud se dosažená velikost dávky blíží velikosti dávky zadané pro kanál, může se kanál blížit svému limitu pro odesílání zpráv.

Pomocí příkazů `START TRACE` a `STOP TRACE` můžete řídit shromažďování trasování evidence a trasování statistiky. Pomocí voleb `STATCHL` a `STATACLS` v kanálu a správci front můžete řídit, zda kanály produkují data SMF.

z/OS

Podmínky a úlohy pro správu produktu IBM MQ for z/OS

Toto téma použijte jako úvod do terminologie a úloh, které jsou specifické pro IBM MQ for z/OS.

Některé podmínky a úlohy nezbytné pro správu produktu IBM MQ for z/OS jsou specifické pro platformu z/OS. Následující seznam obsahuje některé z těchto termínů a úloh.

- [Sdílené fronty](#)
- [Sady stránek a fondy vyrovnávacích pamětí](#)
- [Protokolování](#)
- [přizpůsobení prostředí správce front](#)
- [Restartovat a obnovit](#)
- [Zabezpečení](#)
- [Dostupnost](#)
- [Práce s objekty](#)
- [Monitorování a statistika](#)
- [Aplikační prostředí](#)

Sdílené fronty

Fronty mohou být *nesdílené*vlastněné a přístupné pouze jednomu správci front nebo *sdílené*vlastněné *skupinou sdílení front*. Skupina sdílení front se skládá z několika správců front spuštěných v rámci jednoho prostředí sysplex produktu z/OS, kteří mohou souběžně přistupovat ke stejným definicím objektů a datům zpráv produktu IBM MQ. V rámci skupiny sdílení front jsou sdílené definice objektů uloženy ve sdílené databázi Db2. Zprávy sdílené fronty jsou uloženy v jedné nebo více strukturách prostředku Coupling Facility (struktury prostředku Coupling Facility). Pokud jsou data zprávy příliš velká na to, aby je bylo možné uložit přímo ve struktuře (velikost větší než 63 kB), nebo pokud je zpráva dostatečně velká, aby ji pravidla definovaná v instalaci vybrali pro odlehčování, jsou řídicí informace zprávy stále uloženy v položce prostředku Coupling Facility, ale data zprávy jsou odložena do sdílené datové sady zpráv (SMDS) nebo do sdílené databáze Db2. Sdílené datové sady zpráv, sdílená databáze Db2 a struktury prostředku Coupling Facility jsou prostředky, které jsou společně spravovány všemi správci front ve skupině.

Sady stránek a fondy vyrovnávacích pamětí

Je-li zpráva vložena do nesdílené fronty, správce front uloží data na sadu stránek tak, aby je bylo možné načíst, když následná operace získá zprávu ze stejné fronty. Pokud je zpráva odebrána z fronty, prostor v sadě stránek, která uchovává data, je později uvolněn pro opětovné použití. S rostoucím počtem zpráv

zadrženyých ve frontě se zvětšuje prostor použitý v sadě stránek a s zmenšením počtu zpráv ve frontě se zmenšuje prostor použitý v sadě stránek.

Aby se snížily náklady na výkon při zápisu dat do sad stránek a jejich čtení ze sad stránek, správce front ukládá aktualizace do paměti procesoru. Množství paměti použité pro vyrovnávací paměť pro přístup k sadě stránek je řízeno prostřednictvím objektů IBM MQ , které se nazývají *fondy vyrovnávacích paměti*.

Další informace o sadách stránek a fondech vyrovnávacích paměti naleznete v tématu [Správa úložiště](#).

Protokolování

Veškeré změny objektů uložených v sadách stránek a operace s trvalými zprávami jsou zaznamenány jako záznamy protokolu. Tyto záznamy protokolu se zapisují do datové sady protokolu nazvané *aktivní protokol*. Název a velikost datové sady aktivního protokolu jsou uloženy v datové sadě s názvem *datová sada samozavedení* (BSDS).

Po zaplnění datové sady aktivního protokolu se správce front přepne na jinou datovou sadu protokolu, aby mohlo pokračovat v protokolování, a zkopíruje obsah datové sady úplného aktivního protokolu do datové sady *protokolu archivu* . Informace o těchto akcích, včetně názvu datové sady protokolu archivace, jsou uloženy v datové sadě zaváděcího programu. Koncepčně existuje řada datových sad aktivního protokolu, kterými správce front prochází; je-li aktivní protokol naplněn, jsou data protokolu odložena do protokolu archivu a datová sada aktivního protokolu je k dispozici pro opětovné použití.

Další informace o datových sadách protokolu a zaváděcího programu viz [“přihlášení IBM MQ for z/OS”](#) na stránce 222.

Přizpůsobení prostředí správce front

Při spuštění správce front je načtena sada inicializačních parametrů, které řídí způsob práce správce front. Kromě toho se čtou datové sady obsahující příkazy IBM MQ a provedou se příkazy, které obsahují. Tyto datové sady obvykle obsahují definice systémových objektů, které jsou nezbytné pro spuštění produktu IBM MQ , a můžete je upravit tak, aby definovaly nebo inicializovaly objekty IBM MQ nezbytné pro vaše provozní prostředí. Po přečtení těchto datových sad jsou všechny jimi definované objekty uloženy buď v sadě stránek, nebo v souboru Db2.

Další informace o inicializačních parametrech a systémových objektech viz [“Definice systému v systému z/OS”](#) na stránce 233.

Obnova a restart

Kdykoli během operace produktu IBM MQ mohou být v úložišti procesoru uloženy změny, které dosud nebyly zapsány do sady stránek. Tyto změny jsou zapsány do sady stránek, která je v poslední době používána úlohou na pozadí v rámci správce front.

Pokud dojde k nestandardnímu ukončení správce front, může fáze zotavení při restartování správce front obnovit ztracené změny sady stránek, protože data trvalých zpráv jsou uložena v záznamech protokolu. To znamená, že produkt IBM MQ může obnovit data trvalých zpráv a změny objektů až do bodu selhání.

Pokud správce front, který je členem skupiny sdílení front, zjistí selhání prostředku Coupling Facility, lze trvalé zprávy v této frontě obnovit pouze v případě, že jste záložovali strukturu prostředku Coupling Facility.

Další informace o obnově a restartování viz [“Obnova a restart na systému z/OS”](#) na stránce 244.

Zabezpečení

Můžete použít externího správce zabezpečení, například server zabezpečení (dříve označovaný jako RACF). k ochraně prostředků, které produkt IBM MQ vlastní a spravuje, před přístupem neautorizovaných

uživatelů. Pro zabezpečení kanálu můžete také použít protokol TLS (Transport Layer Security). Protokol TLS je součástí produktu IBM MQ .

Další informace o zabezpečení IBM MQ viz [“Koncepte zabezpečení v produktu IBM MQ for z/OS”](#) na stránce 260.

Dostupnost

Existuje několik funkcí produktu IBM MQ , které jsou navrženy pro zvýšení dostupnosti systému v případě selhání správce front nebo komunikačního subsystému. Další informace o těchto funkcích viz [“Dostupnost na z/OS”](#) na stránce 267.

Manipulace s objekty

Když je správce front spuštěn, můžete manipulovat s objekty IBM MQ buď prostřednictvím rozhraní konzoly z/OS , nebo pomocí obslužného programu administrace, který používá služby ISPF pod TSO. Oba mechanismy vám umožňují definovat, pozměnit nebo odstranit objekty IBM MQ . Můžete také řídit a zobrazit stav různých funkcí produktu IBM MQ a správce front.

Další informace o těchto zařízeních naleznete v tématu [Zdroje, ze kterých můžete zadat příkazy MQSC a PCF v systému IBM MQ for z/OS](#).

S objekty IBM MQ můžete manipulovat také pomocí grafického uživatelského rozhraní IBM MQ Explorer, které poskytuje vizuální způsob práce s frontami, správci front a dalšími objekty.

Monitorování a statistiky

Pro monitorování správců front a inicializátorů kanálů je k dispozici několik zařízení. Můžete také shromažďovat statistiky pro účely vyhodnocení výkonu a účtování.

Další informace o těchto zařízeních viz [“Monitorování a statistiky na IBM MQ for z/OS”](#) na stránce 270.

Aplikační prostředí

Po spuštění správce front se k němu mohou aplikace připojit a začít používat rozhraní API IBM MQ . Může se jednat o aplikace CICS, IMS, Batch nebo WebSphere Application Server . Aplikace IBM MQ mohou také přistupovat k aplikacím v systémech CICS a IMS , které si nejsou vědomy produktu IBM MQ, pomocí mostů CICS a IMS .

Další informace o těchto zařízeních viz [“IBM MQ a další produkty z/OS”](#) na stránce 273.

Chcete-li získat informace o psaní aplikací IBM MQ , prohlédněte si následující dokumentaci:

- [Vývoj aplikací](#)
- [Použití C++](#)
- [Použití IBM MQ classes for Java](#)

▶ z/OS

Sdílené fronty a skupiny sdílení front

K implementaci vysoké dostupnosti prostředků IBM MQ můžete použít sdílené fronty a skupiny sdílení front. Sdílené fronty a skupiny sdílení front jsou funkce jedinečné pro produkt IBM MQ for z/OS na platformě z/OS .

Tento oddíl popisuje atributy a výhody a nabízí informace o tom, jak může několik správců front sdílet stejné fronty a zprávy v těchto frontách.

Co je to sdílená fronta?

Sdílená fronta je typem lokální fronty. Ke zprávám v této frontě může přistupovat jeden nebo více správců front, kteří jsou v prostředí sysplex.

Skupina sdílení front

Správci front, kteří mají přístup ke stejné sadě sdílených front, tvoří skupinu s názvem *skupina sdílení front*.

Každý správce front má přístup ke zprávám

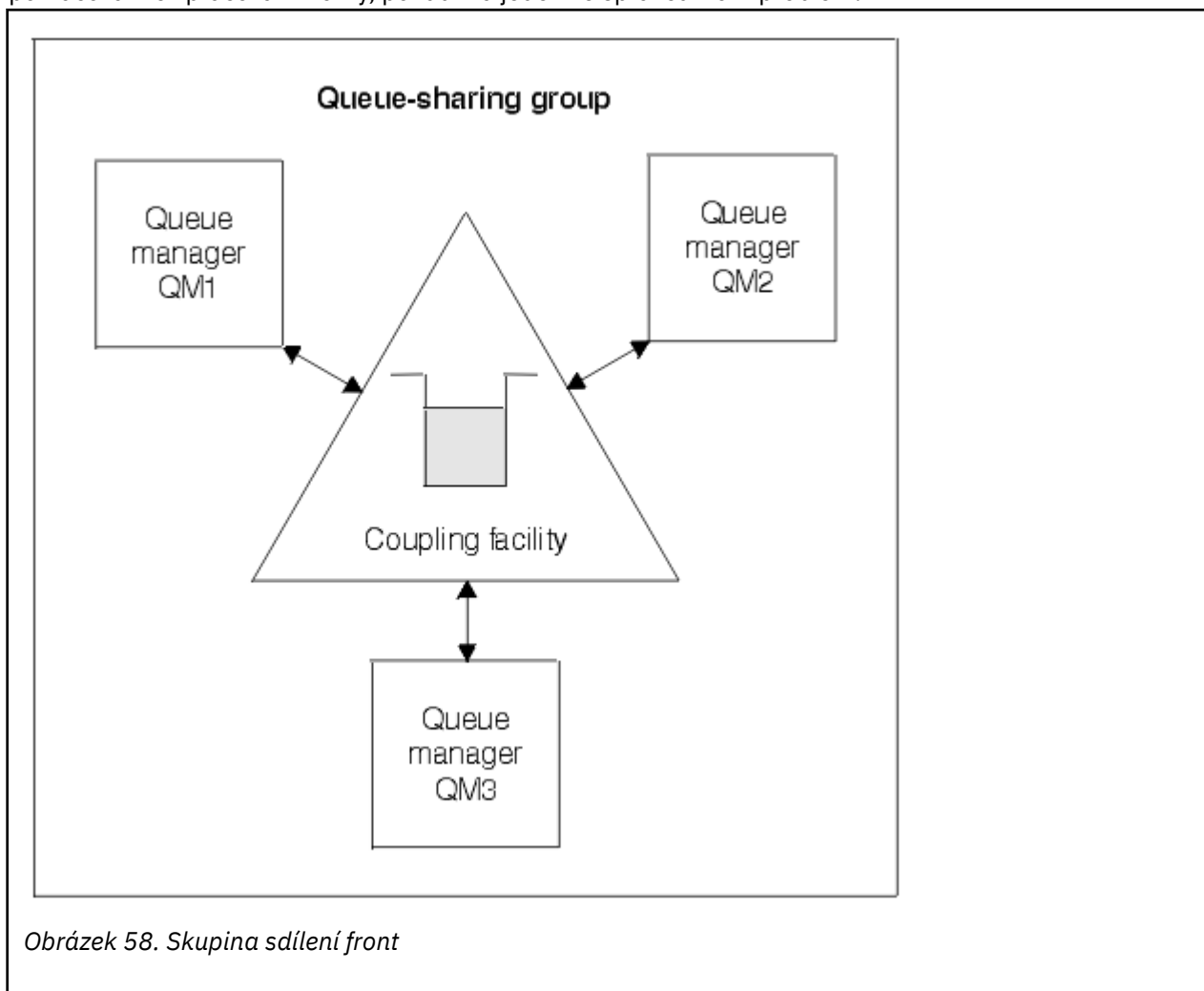
Ke sdílené frontě může přistupovat libovolný správce front ve skupině sdílení front. To znamená, že můžete vložit zprávu do sdílené fronty v jednom správci front a získat stejnou zprávu z fronty z jiného správce front. To poskytuje rychlý mechanismus komunikace v rámci skupiny sdílení front, která nevyžaduje, aby byly kanály mezi správci front aktivní.

Produkt IBM MQ podporuje odlehčování zpráv do produktu Db2 nebo datové sady sdílených zpráv (SMDS). Odlehčování zpráv libovolné velikosti je konfigurovatelné.

V části [Obrázek 58 na stránce 162](#) jsou uvedeni tři správci front a prostředek CF (coupling facility), který vytváří skupinu sdílení front. Všichni tři správci front mohou přistupovat ke sdílené frontě v prostředku Coupling Facility.

Aplikace se může připojit k libovolnému správci front v rámci skupiny sdílení front. Vzhledem k tomu, že všichni správci front ve skupině sdílení front mají přístup ke všem sdíleným frontám, aplikace nezávisí na dostupnosti konkrétního správce front; kterýkoli správce front ve skupině sdílení front může frontu obsluhovat.

To poskytuje větší dostupnost, protože všichni ostatní správci front ve skupině sdílení front mohou pokračovat ve zpracování fronty, pokud má jeden ze správců front problém.



Obrázek 58. Skupina sdílení front

Definice fronty je sdílána všemi správci front

Definice sdílených front jsou uloženy v databázové tabulce Db2 OBJ_B_QUEUE. Z tohoto důvodu je třeba definovat frontu pouze jednou a poté k ní mohou přistupovat všichni správci front ve skupině sdílení front. To znamená, že je třeba vytvořit méně definic.

Naproti tomu definice nesdílené fronty je uložena v sadě stránek nula správce front, který frontu vlastní (jak je popsáno v tématu [Sady stránek](#)).

Sdílenou frontu nelze definovat, pokud fronta s tímto názvem již byla definována v sadách stránek definujícího správce front. Podobně nelze definovat lokální verzi fronty v sadách stránek správce front, pokud existuje sdílená fronta se stejným názvem.

Co je to skupina sdílení front?

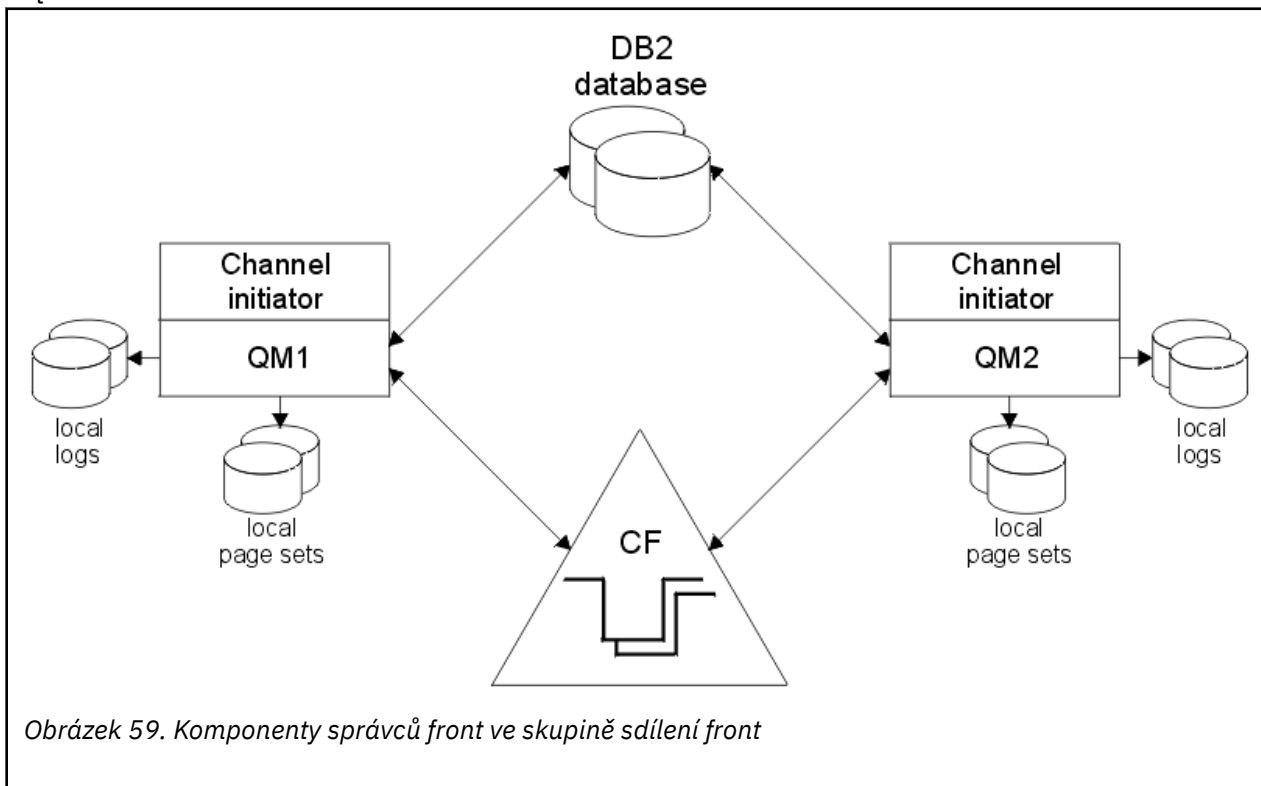
Skupina správců front, kteří mají přístup ke stejným sdíleným frontám, se nazývá skupina sdílení front. Každý člen skupiny sdílení front má přístup ke stejné sadě sdílených front.

Názvy skupin sdílení front jsou tvořeny nejvýše čtyřmi znaky. Název musí být v síti jedinečný a nesmí být shodný s žádným názvem správce front.

Obrázek [Obrázek 59 na stránce 163](#) znázorňuje skupinu sdílení front, která obsahuje dva správce front. Každý správce front má inicializátor kanálu a vlastní lokální sady stránek a datové sady protokolu.

Každý člen skupiny sdílení front se musí také připojit k systému Db2. Všechny systémy Db2 musí být ve stejné skupině sdílení dat Db2, aby měli správci front přístup ke sdílenému úložišti Db2 používaném k uchování definic sdílených objektů. Jedná se o definice libovolného typu objektu IBM MQ (například fronty a kanály), které jsou definovány pouze jednou a pak je může použít libovolný správce front ve skupině. Ty se nazývají *globální* definice a jsou popsány v části [Soukromé a globální definice](#).

Na konkrétní skupinu sdílení dat může odkazovat více než jedna skupina sdílení front. Při spuštění zadejte název subsystému Db2 a skupinu sdílení dat, kterou správce front používá, v parametrech systému IBM MQ.



Pokud se správce front připojí ke skupině sdílení front, má přístup ke sdíleným objektům definovaným pro tuto skupinu a tohoto správce front můžete použít k definování nových sdílených objektů v rámci skupiny. Jsou-li v rámci skupiny definovány sdílené fronty, můžete pomocí tohoto správce front vkládat zprávy

do těchto sdílených front a získávat z nich zprávy. Kterýkoli správce front ve skupině může načíst zprávy uchovávané ve sdílené frontě.

Příkaz MQSC můžete zadat jednou a nechat jej spustit ve všech správcích front v rámci skupiny sdílení front, jako by byl zadán v jednotlivých správcích front. Pro tento účel se použije atribut *command scope*. Tento atribut je popsán v části [Adresování příkazů do různých správců front](#).

Pokud je správce front spuštěn jako člen skupiny sdílení front, musí být možné rozlišovat mezi objekty IBM MQ definovanými soukromě pro daného správce front a objekty IBM MQ definovanými globálně, které jsou k dispozici pro všechny správce front ve skupině sdílení front. Pro tento účel se používá atribut *dispozice skupiny sdílení front*. Tento atribut je popsán v části [Soukromé a globální definice](#).

Můžete definovat jednu sadu profilů zabezpečení, které řídí přístup k objektům IBM MQ kdekoli ve skupině. To znamená, že počet profilů, které musíte definovat, je výrazně snížen.

Správce front může patřit pouze do jedné skupiny sdílení front a všichni správci front ve skupině musí být ve stejném prostředí sysplex. Určete, do které skupiny sdílení front patří správce front, v parametrech systému při spuštění.

Související pojmy

[“Kde jsou zadrženy sdílené zprávy fronty?” na stránce 164](#)

Každá zpráva ve sdílené frontě je reprezentována položkou ve struktuře seznamu prostředku Coupling Facility systému z/OS. Pokud jsou data zprávy příliš velká na to, aby se vešla do stejné položky, jsou odložena buď do sdílené datové sady zpráv (SMDS), nebo do Db2.

[“Výhody používání sdílených front” na stránce 180](#)

Sdílená fronta umožňuje, aby aplikace IBM MQ byly rozšiřitelné, vysoce dostupné a aby bylo možné implementovat vyrovnávání pracovní zátěže.

[“Skupiny distribuovaného řazení do front a sdílení front” na stránce 200](#)

Distribuované řazení do front a skupiny sdílení front jsou dvě techniky, které můžete použít ke zvýšení dostupnosti vašich aplikačních systémů. V tomto tématu naleznete další informace o těchto technikách.

[“Ovlivňování distribuce pracovní zátěže pomocí sdílených front” na stránce 203](#)

Prostřednictvím tohoto tématu můžete porozumět faktorům, které ovlivňují distribuci pracovní zátěže se sdílenými frontami ve skupině sdílení front.

Související odkazy

[“Kde najít další informace o sdílených frontách a skupinách sdílení front” na stránce 204](#)

V tabulce v tomto tématu naleznete další informace o tom, jak produkt IBM MQ for z/OS používá sdílené fronty a skupiny sdílení front.

Kde jsou zadrženy sdílené zprávy fronty?

Každá zpráva ve sdílené frontě je reprezentována položkou ve struktuře seznamu prostředku Coupling Facility systému z/OS. Pokud jsou data zprávy příliš velká na to, aby se vešla do stejné položky, jsou odložena buď do sdílené datové sady zpráv (SMDS), nebo do Db2.

Pokud byla struktura prostředku CF nakonfigurována tak, aby používala paměť SCM (System Class Memory), produkt IBM MQ ji může používat bez další konfigurace.

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Úložiště zpráv sdílené fronty

Zprávy vkládané do sdílených front nejsou uloženy v sadách stránek a nepoužívají fondy vyrovnávacích pamětí.

Zprávy ve sdílených frontách mají položky v seznamových strukturách v prostředku CF (z/OS coupling facility). Mnoho správců front ve stejném prostředí sysplex může přistupovat k těmto zprávám pomocí struktury seznamu prostředku CF.

Data zpráv pro malé sdílené fronty jsou obvykle zahrnuta v položce prostředku Coupling Facility. V případě větších zpráv mohou být data zpráv uložena buď ve sdílené datové sadě zpráv (SMDS), nebo jako jeden či více binárních velkých objektů (BLOB) v tabulce Db2 , kterou sdílí skupina sdílení dat Db2 . Data zprávy větší než 63 kB jsou vždy odložena do SMDS nebo Db2. Menší zprávy lze také volitelně odlehčovat stejným způsobem, aby se ušetřilo místo ve struktuře prostředku Coupling Facility. Další informace viz část [“Určení voleb odlehčování pro sdílené zprávy”](#) na stránce 166.

Na zprávy vložené do sdílené fronty se odkazuje ve struktuře prostředku Coupling Facility, dokud nejsou načteny příkazem MQGET. Operace prostředku Coupling Facility se používají k:

- Vyhledat další zprávu, kterou lze načíst
- Zamknout nepotvrzené zprávy ve sdílených frontách
- Upozornit zainteresované správce front na příchod potvrzených zpráv

Operace MQPUT a MQGET pro trvalé zprávy jsou zaznamenány do protokolu správce front provádějícího tuto operaci. Tím se minimalizuje riziko ztráty dat v případě selhání prostředku Coupling Facility.

Prostředek Coupling Facility

Na zprávy uchovávané ve sdílených frontách se odkazuje v prostředku Coupling Facility. Prostředek CF (coupling facility) se nachází mimo některý z obrazů z/OS v prostředí sysplex a je obvykle konfigurován tak, aby běžel na jiném napájecím zdroji. Prostředek Coupling Facility je proto odolný vůči selháním softwaru a lze jej konfigurovat tak, aby byl odolný vůči selháním hardwaru nebo výpadkům napájení. To znamená, že zprávy uložené v prostředku Coupling Facility jsou vysoce dostupné.

Každá struktura seznamu prostředku Coupling Facility používaná v produktu IBM MQ je vyhrazena pro specifickou skupinu sdílení front, avšak prostředek CF může obsahovat struktury pro více než jednu skupinu sdílení front. Správci front v různých skupinách sdílení front nemohou sdílet data. Ke struktuře seznamu prostředku Coupling Facility se může současně připojit až 32 správců front ve skupině sdílení front.

Jedna struktura seznamu prostředku Coupling Facility může obsahovat až 512 sdílených front. Celkové množství dat zpráv uložených ve struktuře je omezeno kapacitou struktury. S parametrem **CFLEVEL (5)** však můžete použít parametry odlehčování pro odlehčování dat pro zprávy o velikosti menší než 63 kB, čímž zvýšíte počet zpráv, které mohou být uloženy ve struktuře, ačkoli každá zpráva stále vyžaduje alespoň položku prostředku Coupling Facility plus alespoň 768 bajtů dat, tvořených 256 bajty pro položku a 512 bajty pro dva prvky záhlaví a deskriptoru.

Velikost struktury seznamu je omezena následujícími faktory:

- Musí být v rámci jednoho prostředku Coupling Facility.
- Může sdílet dostupné úložiště prostředku Coupling Facility s dalšími strukturami pro produkt IBM MQ a další produkty.

Struktury seznamu prostředku Coupling Facility mohou mít k sobě přidruženou paměť paměťové třídy. V určitých situacích může být tato paměť paměťové třídy užitečná při použití se sdílenými frontami. Další informace viz [“Použití paměti paměťové třídy se sdílenými frontami”](#) na stránce 182.

Plánování velikosti struktury prostředku CF

Pokud potřebujete poradit s dimenzováním struktur prostředku CF, můžete použít podporu [MP16: IBM MQ for z/OS Plánování a vyladění kapacity](#) . Můžete také použít webový nástroj [CFSizer](#), který poskytuje produkt IBM jako pomůcku pro velikost prostředku CF.

Objekt struktury prostředku CF

Použití struktury prostředku Coupling Facility správcem front je určeno v objektu struktury CF (CFSTRUCT) IBM MQ .

Tyto objekty struktury jsou uloženy v adresáři Db2.

Při použití příkazů nebo definic týkajících se struktury prostředku Coupling Facility systému z/OS jsou vyžadovány první čtyři znaky názvu skupiny sdílení front. Objekt IBM MQ CFSTRUCT však vždy existuje v rámci jedné skupiny sdílení front, a proto jeho název neobsahuje první čtyři znaky názvu skupiny sdílení front. Například CFSTRUCT (MYDATA) definovaný ve skupině sdílení front začínající na SQ03 by používal strukturu seznamu prostředku Coupling Facility SQ03MYDATA.

Struktury prostředku CF mají atribut CFLEVEL, který určuje jejich funkční schopnost:

- 1, 2-lze použít pro přechodné zprávy menší než 63 kB
- 3-lze použít pro trvalé a přechodné zprávy menší než 63 kB
- 4-lze použít pro trvalé a přechodné zprávy až do velikosti 100 MB
- 5-lze použít pro trvalé a přechodné zprávy až do velikosti 100 MB a selektivně odlehčovat do sdílených datových sad zpráv (SMDS) nebo Db2.

Poznámka: Při použití IBM MQ můžete šifrovat strukturu prostředku Coupling Facility. Další informace viz [Šifrování dat struktury prostředku Coupling Facility](#) .

Zálohování a obnova prostředku Coupling Facility

Struktury seznamu prostředku Coupling Facility můžete zálohovat pomocí příkazu IBM MQ BACKUP CFSTRUCT. Tím se kopie trvalých zpráv, které jsou aktuálně ve struktuře prostředku CF, vloží do datové sady aktivního protokolu správce front, který zálohu vytváří, a zapíše záznam zálohy do souboru Db2.

Pokud prostředek CF selže, můžete použít příkaz IBM MQ RECOVER CFSTRUCT. Tato volba používá záznam zálohy z produktu Db2 k vyhledání a obnově trvalých zpráv ze zálohy struktury prostředku CF. Všechny aktivity od poslední zálohy jsou přehrávány pomocí protokolů všech správců front ve skupině sdílení front a struktura prostředku CF je poté obnovena až do bodu před selháním.

Další podrobnosti viz příkazy [BACKUP CFSTRUCT](#) a [RECOVER CFSTRUCT](#) .

Související pojmy

[“Určení voleb odlehčování pro sdílené zprávy” na stránce 166](#)

Můžete zvolit, kde jsou data zpráv pro sdílenou zprávu fronty uložena, buď v tabulce Db2 , nebo v datové sadě sdílených zpráv (SMDS). Můžete také vybrat, které zprávy jsou odlehčeny, na základě velikosti zprávy a aktuálního využití struktury prostředku Coupling Facility (CF).

[“Správa prostředí sdílené datové sady zpráv \(SMDS\)” na stránce 169](#)

Vyberete-li sdílené datové sady zpráv pro odlehčování velkých zpráv, musíte mít také na paměti informace, které produkt IBM MQ používá ke správě těchto datových sad, a příkazy používané pro práci s těmito informacemi. V tomto tématu se dozvíte, jak spravovat datové sady sdílených zpráv.

Určení voleb odlehčování pro sdílené zprávy

Můžete zvolit, kde jsou data zpráv pro sdílenou zprávu fronty uložena, buď v tabulce Db2 , nebo v datové sadě sdílených zpráv (SMDS). Můžete také vybrat, které zprávy jsou odlehčeny, na základě velikosti zprávy a aktuálního využití struktury prostředku Coupling Facility (CF).

Data zpráv pro sdílené fronty lze odlehčovat z prostředku Coupling Facility a uložit je v tabulce Db2 nebo ve spravované datové sadě systému IBM MQ nazvané *sdílená datová sada zpráv* (SMDS).

U zpráv, které jsou větší než velikost položky prostředku Coupling Facility 63 kB, může odlehčování dat zprávy do SMDS vést k výraznému zlepšení výkonu v porovnání s odlehčováním do systému Db2.

Každá sdílená zpráva fronty je stále spravována pomocí položky seznamu ve struktuře prostředku Coupling Facility, ale když jsou data zprávy odložena do SMDS, obsahuje položka prostředku Coupling Facility pouze některé řídicí informace a seznam odkazů na příslušné diskové bloky, kde je zpráva uložena. Použití tohoto mechanismu znamená, že velikost úložiště prvků prostředku Coupling Facility vyžadovaná pro každou zprávu představuje pouze zlomek skutečné velikosti zprávy.

Výběr místa uložení zpráv sdílené fronty

Výběr sdíleného úložiště zpráv SMDS nebo Db2 je řízen parametrem **OFFLOAD(SMDS|DB2)** v definici **CFSTRUCT**. **OFFLOAD(SMDS)** je výchozí hodnota.

Tento parametr také vyžaduje, aby **CFSTRUCT** používal **CFLEVEL(5)** nebo vyšší.

Parametr **OFFLOAD** je platný pouze od **CFLEVEL(5)**. Další podrobnosti viz DEFINE CFSTRUCT.

Produkt **OFFLOAD(DB2)** je primárně podporován pro účely migrace.

Výběr zpráv sdílené fronty, které jsou odlehčeny

Data zprávy jsou odložena do SMDS nebo Db2 na základě velikosti dat zprávy a aktuálního využití struktury prostředku Coupling Facility. Existují tři pravidla a každé pravidlo určuje odpovídající dvojici parametrů. Tyto parametry jsou odpovídající procentní část prahové hodnoty využití struktury prostředku Coupling Facility (**OFFLDnTH**) a omezení velikosti zprávy (**OFFLDnSZ**).

Aktuální implementace tří pravidel je určena pomocí následujících dvojic klíčových slov:

- OFFLD1TH a OFFLD1SZ
- OFFLD2TH a OFFLD2SZ
- OFFLD3TH a OFFLD3SZ

Dvojice pravidel	Výchozí hodnota	Popis
Dvojice pravidel 1	OFFLD1TH(70) a OFFLD1SZ(32K)	Pokud je struktura prostředku Coupling Facility více než 70% dat úplného odlehčování pro zprávy větší než 32 kB
Dvojice pravidel 2	OFFLD2TH(80) a OFFLD2SZ(4K)	Je-li struktura prostředku Coupling Facility větší než 80% dat úplného odlehčování pro zprávy větší než 4 kB
Dvojice pravidel 3	OFFLD3TH(90) a OFFLD3SZ(0K)	Pokud je struktura prostředku Coupling Facility větší než 90% dat úplného odlehčování pro zprávy větší než 0 kB (všechny zprávy)

Pokud má pravidlo odlehčování hodnotu OFFLD x SZ 64K, znamená to, že pravidlo není v platnosti. V tomto případě budou zprávy odlehčeny pouze v případě, že je v platnosti jiné pravidlo odlehčování nebo pokud je zpráva větší než 63.75 kB, a proto je příliš velká pro uložení ve struktuře.

Každá zpráva, která je odlehčena, stále vyžaduje 0.75 kB úložiště v prostředku Coupling Facility.

Tři pravidla odlehčování, která mohou být specifikována pro každou konstrukci, mají být použita následujícím způsobem.

- Výkon
 - Je-li ve struktuře aplikace dostatek místa, měla by být data zpráv odložena pouze v případě, že jsou příliš velká na to, aby mohla být uložena ve struktuře, nebo pokud překročí nižší prahovou hodnotu velikosti zprávy tak, aby hodnota výkonu uložení ve struktuře nestála za velikost prostoru struktury, který by potřebovala.

- Je-li vyžadována specifická prahová hodnota velikosti zprávy, je obvykle určena pomocí prvního pravidla odlehčování.
- kapacita
 - Pokud je ve struktuře aplikace velmi málo místa, maximální množství dat zprávy by mělo být odlehčeno, aby bylo možné co nejlépe využít zbývající prostor.
 - Třetí pravidlo odlehčování se obvykle používá k označení, že když je struktura téměř plná, většina zpráv by měla být odlehčena, takže položky ve struktuře aplikace budou mít obvykle minimální velikost (což vyžaduje přibližně 0.75K bajtů).
 - Parametr prahové hodnoty využití by měl být vybrán na základě velikosti struktury aplikace a maximálního očekávaného nevyřízených požadavků. Pokud je například maximální očekávaný počet nevyřízených zpráv 1M , bude velikost úložiště struktury požadovaná pro tento počet zpráv přibližně 0.75G bajtů. To například znamená, že pokud je struktura přibližně 10G bajtů, musí být prahová hodnota využití pro odlehčování všech zpráv nastavena na 92% nebo nižší.
 - Prostor struktury je rozdělen na prvky a položky, a i když může být dostatek prostoru celkově, jeden z nich může dojít před druhým. Systém poskytuje funkce AUTOALTER pro úpravu poměru v případě potřeby, ale to není příliš citlivé, takže množství místa, které je ve skutečnosti k dispozici, může být o něco menší. Proto může být lepší zaměřit se na maximálně 90% maximálního prostoru struktury, takže v předchozím příkladu by byla prahová hodnota využití pro odlehčování všech zpráv lépe nastavena kolem 80%.
- Polstrovaný přechod:
 - Vzhledem k tomu, že se prostor ve struktuře prostředku Coupling Facility zmenšuje, bylo by nežádoucí, kdyby došlo k velké náhlé změně výkonových charakteristik. Je také nežádoucí, aby správa prostředku Coupling Facility měla náhlou změnu prahové hodnoty v typickém poměru vstupů k používaným prvkům.
 - Druhé pravidlo odlehčování se obvykle používá k zajištění mezilehlého polštáře mezi výkonem a pravidly odlehčování s předpojatou kapacitou. Může být nastaven tak, aby způsobil významné zvýšení aktivity odlehčování, pokud prostor použitý ve struktuře prostředku Coupling Facility překročí mezilehlou prahovou hodnotu. To znamená, že zbývající prostor je spotřebován pomaleji, a poskytuje prostředku Coupling Facility automatické zpracování změn více času na přizpůsobení se vyšším úrovním využití.

Pokud strukturu prostředku Coupling Facility nelze rozšířit a je třeba uložit alespoň určitý předem určený počet zpráv, lze třetí pravidlo podle potřeby upravit tak, aby se zajistilo, že odlehčování dat pro všechny zprávy začne na příslušné prahové hodnotě, aby se zajistilo, že prostor bude vyhrazen pro daný předem určený počet zpráv.

Je-li například velikost struktury prostředku Coupling Facility 4 GB a předem určený počet zpráv je 1 milion, je zapotřebí $1\ 000\ 000 * 0.75\ \text{kB}$, což je 768 MB, 18.75% ze 4 GB. V tomto případě je třeba nastavit prahovou hodnotu pro odlehčování všech zpráv přibližně na 80%, nikoli na 90%. Tím získáte parametry OFFLD3TH(80) a OFFLD3SZ(OK). Ostatní parametry odlehčování by také měly být upraveny.

Je-li zjištěno, že odlehčování velmi malých zpráv má významný dopad na výkon, ale relativní dopad je menší pro větší zprávy, lze snížit prahové hodnoty využití pro ostatní pravidla, aby bylo možné dříve odlehčovat větší zprávy, a ponechat tak více prostoru ve struktuře pro malé zprávy, než budou muset být odlehčeny.

Pokud se například zprávy, které přesahují 32KB , vyskytují často, ale výkon v uplynulém čase pro jejich odlehčování (jak je určen statistikou RMF nebo výkonem aplikace) je velmi podobný tomu, aby byly uchovány v prostředku Coupling Facility, může být prahová hodnota pro první pravidlo nastavena na 0%, aby se všechny tyto zprávy odlehčily. To poskytuje parametry OFFLD1TH(0) a OFFLD1SZ(32K). Je třeba znovu upravit ostatní parametry odlehčování.

Pokud existuje mnoho zpráv okolo specifických mezilehlých velikostí, například 16 kB a 6 kB, může být užitečné změnit volbu velikosti zprávy pro druhé pravidlo tak, aby větší zprávy byly odlehčeny při poměrně nízké prahové hodnotě využití, což ušetří značné množství místa, ale menší z nich jsou stále uloženy pouze v prostředku CF.

Vyberete-li sdílené datové sady zpráv pro odlehčování velkých zpráv, musíte mít také na paměti informace, které produkt IBM MQ používá ke správě těchto datových sad, a příkazy používané pro práci s těmito informacemi. V tomto tématu se dozvíte, jak spravovat datové sady sdílených zpráv.

Objekty SMDS

Vlastnosti a stav každé datové sady sdílených zpráv jsou sledovány ve sdíleném objektu SMDS, který lze aktualizovat prostřednictvím libovolného správce front ve skupině sdílení front.

Pro každého správce front existuje jedna sdílená datová sada zpráv, která má přístup ke každé struktuře aplikace prostředku Coupling Facility. Datová sada sdílených zpráv je identifikována názvem vlastního správce front, uvedeným pomocí klíčového slova SMDS, a názvem struktury aplikace, uvedeným pomocí klíčového slova CFSTRUCT.

Poznámka: Při definování datových sad SMDS pro strukturu musíte mít jednu pro každého správce front.

Objekt SMDS je uložen v poli (s jednou položkou pro každého správce front ve skupině), které tvoří rozšíření odpovídajícího objektu CFSTRUCT uloženého v adresáři Db2.

Neexistuje žádný příkaz DEFINE nebo DELETE objektu SMDS, protože je vytvořen nebo odstraněn jako součást objektu CFSTRUCT, ale existuje příkaz ALTER pro změnu nastavení pro individuálního vlastního správce front.

Další informace o příkazech SMDS viz [“Příkazy související se SMDS” na stránce 179](#)

Informace SMDSCONN

Je možné, že datová sada sdílených zpráv bude v normálním stavu, ale jeden nebo více správců front se k ní nebude moci připojit, například kvůli problému s definicí zabezpečení nebo s konektivitou zařízení s přímým přístupem. Proto je nezbytné, aby každý správce front sledoval stav připojení a informace o dostupnosti pro každou datovou sadu sdílených zpráv, například zda se k ní může aktuálně připojit, a pokud ne, proč ne.

Informace SMDSCONN představují připojení správce front ke sdílené datové sadě zpráv. Pokud jde o samotnou datovou sadu sdílených zpráv, je identifikována správcem front, který vlastní datovou sadu sdílených zpráv (jak je uvedeno v klíčovém slově SMDS pro samotný sdílený objekt) v kombinaci s názvem CFSTRUCT.

Neexistuje žádný parametr pro identifikaci připojovacího se správce front, protože příkazy adresované specifickému správci front mohou odkazovat pouze na informace SMDSCONN pro téhož správce front.

Položky informací SMDSCONN jsou udržovány v hlavním úložišti ve vlastním správcem front a jsou znovu vytvořeny při restartování správce front. Pokud však bylo připojení od jednotlivého správce front explicitně zastaveno, jsou tyto informace uloženy také jako příznak v poli připojení v příslušném objektu CFSTRUCT nebo SMDS, aby byly zachovány i po restartu správce front.

Informace o stavu a dostupnosti

Informace o stavu označují stav prostředku nebo připojení (například zda se ještě nepoužívá, je v normálním použití nebo potřebuje obnovu). Obvykle se popisuje pomocí klíčového slova STATUS. Možné hodnoty závisí na typu objektu.

Informace o stavu se obvykle aktualizují automaticky, například když je zjištěna chyba při použití prostředku nebo připojení. V některých případech však lze k aktualizaci stavu použít také příkaz, který umožňuje v případech, kdy není možné, aby správce front automaticky určil správný stav.

Informace o dostupnosti označují, zda lze prostředek nebo připojení použít, a obvykle jsou primárně určeny informacemi o stavu. Pro prostředky nebo typy připojení používané v podpoře datových sad sdílených zpráv jsou implementovány tři úrovně dostupnosti:

K dispozici

To znamená, že prostředek je k dispozici pro běžné použití. To nemusí nutně znamenat, že se v současné době používá (což lze určit místo hodnoty STATUS). Pokud pro datovou sadu vyžaduje zpracování restartu, umožní to vlastnímu správci front její otevření, ale ostatní správci front musí počkat, dokud se datová sada nevrátí do stavu AKTIVNÍ.

Nedostupný kvůli chybě

To znamená, že prostředek byl automaticky znepřístupněn kvůli chybě a neočekává se, že bude znovu k dispozici, dokud nebude provedena nějaká forma zpracování opravy nebo obnovy. Pokusy o opětovné zpřístupnění jsou však povoleny bez zásahu operátora. Takový pokus může být také spuštěn příkazem, který označí prostředek jako povolený, nebo příkazem, který změní stav tak, aby indikoval, že zpracování obnovy bylo dokončeno.

Důvod, proč byl prostředek znepřístupněn, je obvykle zřejmý ze související hodnoty STATUS, ale v některých případech mohou existovat jiné důvody pro znepřístupnění prostředku. V takovém případě je poskytnuta oddělená hodnota REASON, která označuje příčinu.

Nedostupné kvůli příkazu operátora

To znamená, že přístup k prostředku byl příkazem explicitně zakázán. Lze ji zpřístupnit pouze pomocí příkazu, který ji znovu povolí.

Dostupnost SMDS

Pro sdílený objekt SMDS je dostupnost popsána klíčovým slovem ACCESS s možnými hodnotami ENABLED, POZASTAVENO a DISABLED.

Dostupnost lze aktualizovat pomocí příkazu **RESET SMDS** pro příslušný sdílený objekt z libovolného správce front ve skupině pro nastavení ACCESS (ENABLED) nebo ACCESS (DISABLED).

Pokud byla dostupnost dříve ACCESS (POZASTAVENO), její změna na ACCESS (POVOLENO) spustí nový pokus o použití datové sady sdílených zpráv, ale pokud je předchozí chyba stále přítomna, bude dostupnost resetována zpět na ACCESS (POZASTAVENO).

Dostupnost SMDSCONN

Pro lokální položku informací SMDSCONN je dostupnost popsána klíčovým slovem AVAIL s možnými hodnotami NORMAL, ERROR nebo STOPPED. Dostupnost lze aktualizovat pomocí příkazu **START SMDSCONN** nebo **STOP SMDSCONN** adresovaného konkrétnímu správci front, který povolí nebo zakáže jeho připojení.

Pokud byla dostupnost dříve AVAIL (ERROR), její změna na AVAIL (NORMAL) spustí nový pokus o použití datové sady sdílených zpráv, ale pokud je předchozí chyba stále přítomna, bude dostupnost resetována zpět na AVAIL (ERROR).

Sdílený stav a dostupnost datové sady zpráv

Dostupnost každé datové sady sdílených zpráv je spravována v rámci skupiny pomocí informací o sdíleném stavu, které lze zobrazit pomocí příkazu **DISPLAY CFSTATUS** s TYPE (SMDS). Zobrazí informace o stavu pro každého správce front, který aktivoval datovou sadu pro každou strukturu. Každá datová sada může být v jednom z následujících stavů:

NotFound

To znamená, že příslušná datová sada dosud nebyla aktivována. Tento stav se zobrazí pouze v případě, že je určen konkrétní správce front, protože datové sady, které nebyly aktivovány, jsou při výběru všech správců front vynechány.

NOVÁ VERZE

Datová sada se otevírá a inicializuje poprvé a je připravena k aktivaci.

ACTIVE

To znamená, že datová sada je plně k dispozici a měla by být přidělena a otevřena všemi aktivními správci front pro danou strukturu.

FAILED

To znamená, že datová sada není vůbec k dispozici (s výjimkou zpracování obnovy) a musí být uzavřena a zrušena všemi správci front.

Probíhá obnovení

To znamená, že pro tuto datovou sadu probíhá obnova médií (pomocí příkazu RECOVER CFSTRUCT).

Zotaveno

To znamená, že byl vydán příkaz pro přepnutí nezdařené datové sady zpět do aktivního stavu, ale je vyžadováno další zpracování restartu, které ještě není dokončeno, takže datovou sadu může otevřít pouze vlastníci správce front pro zpracování restartu.

PRÁZDNÝ

Datová sada neobsahuje žádné zprávy. Datová sada je uvedena do tohoto stavu, pokud je normálně zavřena vlastníci správce front v době, kdy neobsahuje žádné zprávy. Může být také převeden do stavu EMPTY, když má být předchozí obsah datové sady vyřazen, protože struktura aplikace byla vyprázdněna (pomocí **RECOVER CFSTRUCT** s TYPE PURGE nebo, pouze v případě nezotavitelné struktury, odstraněním předchozí instance struktury). Při příštím otevření datové sady vlastníci správce front se mapa prostoru resetuje na prázdnou hodnotu a stav se změní na AKTIVNÍ.

Vzhledem k tomu, že předchozí obsah datové sady již není požadován, datovou sadu v tomto stavu lze nahradit nově přidělenou datovou sadou, například pro změnu přidělení prostoru nebo jeho přesunutí na jiný svazek.

Výstup příkazu zahrnuje datum a čas, kdy bylo povoleno protokolování obnovy, pokud existuje, a datum a čas, kdy datová sada selhala, pokud není momentálně aktivní.

Datovou sadu sdílených zpráv lze uvést do stavu SELHÁNÍ buď příkazem **RESET SMDS**, nebo automaticky při zjištění některého z následujících typů chyb:

- Datovou sadu nelze přidělit nebo otevřít vlastníci správce front.
- Ověření záhlaví datové sady se nezdaří poté, co bylo úspěšně otevřeno libovolným správce front.
- Pokud vlastníci správce front čte nebo zapisuje data, dojde k trvalé chybě vstupu/výstupu.
- K trvalé chybě I/O dojde, když jiný správce front čte data z datové sady, která úspěšně dokončila otevřené zpracování a ověření.

Když je datová sada ve stavu FAILED nebo INRECOVER, není k dispozici pro normální použití, takže pokud je stav dostupnosti ACCESS (ENABLED), změní se na ACCESS (POZASTAVENO).

Pokud byla datová sada převedena do stavu FAILED, ale není požadována žádná obnova médií, například protože data byla stále platná, ale úložné zařízení bylo dočasně offline, lze použít příkaz **RESET SMDS** k požadování změny stavu přímo do stavu RECOVERY.

Když datová sada přejde do stavu RECOVERY, buď po dokončení zpracování obnovy, nebo v důsledku příkazu **RESET SMDS**, je připravena k opětovnému použití po dokončení zpracování restartu. Pokud byl ve stavu ACCESS (POZASTAVENO), je automaticky přepnut zpět do stavu ACCESS (POVOLENO), což umožňuje vlastníci správce front provádět zpracování restartu. Po dokončení zpracování restartu se stav změní na AKTIVNÍ a všichni ostatní správce front se pak mohou znovu připojit k datové sadě.

Stav a dostupnost připojení datové sady sdílených zpráv

Každý správce front uchovává informace o lokálním stavu a dostupnosti pro své připojení ke každé datové sadě sdílených zpráv, kterou vlastní sám a ostatní správce front ve skupině. Tyto informace lze zobrazit pomocí příkazu **DISPLAY SMDSCONN**.

Pokud nemůže přistupovat ke sdílené datové sadě zpráv ve stavu ACTIVE, která patří jinému správce front, označí připojení jako nedostupné z vlastního pohledu.

Pokud chyba rozhodně označuje problém se samotnou datovou sadou, správce front také automaticky změní sdílený stav, aby označil, že datová sada je nyní ve stavu SELHÁNÍ. Pokud však může být chyba způsobena problémem prostředí, například neautorizováním k otevření datové sady, správce front vydá chybové zprávy a zachází s datovou sadou jako s nedostupnou, ale nezmění stav sdílené datové sady. Pokud se chyba prostředí ukáže jako problém s datovou sadou (například byla přidělena na zařízení, ke

kterému někteří správci front nemají přístup), pak může operátor použít příkaz **RESET SMDS** uvádějící **STATUS (FAILED)**, který umožní obnovit nebo opravit datovou sadu podle potřeby.

Pokud nebylo možné vytvořit připojení ke sdílené datové sadě zpráv, ale datová sada se zdá být platná, lze spustit nový pokus o její použití zadáním příkazu **START SMDSCONN** pro vlastního správce front.

Pokud existuje provozní potřeba dočasně ukončit připojení mezi specifickým správcem front a datovou sadou, ale samotná datová sada není poškozena, lze datovou sadu zavřít a zrušit její přidělení pomocí příkazu **STOP SMDSCONN**. Pokud je datová sada používána, správce front ji normálně zavře (ačkoli všechny požadavky na data v této datové sadě budou odmítnuty s návratovým kódem). Jedná-li se o vlastněnou datovou sadu, uloží správce front mapu prostoru během zpracování **CLOSE**, aniž by bylo nutné znovu spustit zpracování.

Pokud je třeba datovou sadu dočasně vyřadit ze služby ze všech správců front (například ji přesunout), ale není poškozena, je vhodné použít **STOP SMDSCONN** pro příslušnou datovou sadu s volbou **CMDSCOPE (*)**, aby se nejprve zastavili správci front, kteří ji používají, protože tím se vyhnete nutnosti restartovat zpracování, když je datová sada znovu uvedena do provozu. Naopak, je-li datová sada označena jako **SELHÁNÍ**, oznámí to správcům front, že ji musí okamžitě přestat používat, což znamená, že mapa prostoru nebude uložena a bude třeba ji znovu sestavit zpracováním restartu.

Po restartování správce front bude zopakován přístup ke všem sdíleným datovým sadám zpráv, které byly dříve ve stavu **ACCESS (POZASTAVENO)**.

Protokolování obnovy datové sady sdílených zpráv

Trvalé sdílené zprávy jsou protokolovány pro účely obnovy médií. To znamená, že zprávy lze obnovit po selhání struktury prostředku Coupling Facility nebo datových sad sdílených zpráv za předpokladu, že protokoly pro zotavení jsou stále neporušené. Trvalé zprávy lze také znovu vytvořit z protokolů pro zotavení na jiném serveru pro účely zotavení z havárie.

Při zápisu dat zprávy do sdílené datové sady zpráv je každý blok zapsán do datové sady zaprotokolován odděleně a položka zprávy (včetně mapování dat) je zapsána do prostředku Coupling Facility. Proces obnovy vždy obnoví strukturu prostředku Coupling Facility, ale nemusí obnovit jednotlivé sdílené datové sady zpráv, kromě případů, kdy je stav datové sady **SELHÁNÍ**, nebo kdy je stav **AKTIVNÍ**, ale záznam záhlaví datové sady již není platný, což znamená, že datová sada byla znovu vytvořena. Datová sada není vybrána pro obnovu, pokud je její stav **AKTIVNÍ** a záhlaví datové sady je stále platné, nebo pokud je její stav **PRÁZDNÝ**, což znamená, že v době selhání v ní nebyly uloženy žádné zprávy.

Zálohy datových sad sdílených zpráv

Při použití příkazu **BACKUP CFSTRUCT** k vytvoření zálohy sdílených zpráv ve struktuře aplikace jsou všechna data pro trvalé zprávy uložena ve sdílených datových sadách zpráv zálohována současně, stejně jako pro trvalé sdílené zprávy dříve uložené v databázi.

Obnova datové sady sdílených zpráv

Je-li datová sada sdílených zpráv poškozena nebo ztracena, je třeba ji uvést do stavu **SELHÁNÍ**, aby ji správci front nepoužívali, dokud nebude opravena. K tomu obvykle dochází automaticky, ale lze to provést také pomocí příkazu **RESET SMDS**, který uvádí **STATUS (FAILED)**.

Pokud datová sada sdílených zpráv obsahovala trvalé zprávy, lze je obnovit pomocí příkazu **RECOVER CFSTRUCT**. Tento příkaz nejprve obnoví data trvalých zpráv pro tuto datovou sadu sdílených zpráv z nejnovějšího příkazu **BACKUP CFSTRUCT** a poté použije všechny zaprotokolované změny od té doby. Pokud od doby, kdy byla datová sada poprvé aktivována, nebyl proveden žádný příkaz **BACKUP CFSTRUCT**, bude resetován na prázdnou hodnotu a poté budou použity všechny změny od aktivace.

Pokud není k dispozici obsah **CFSTRUCT** a všechny sdílené datové sady zpráv, například v situaci zotavení z havárie, všechny mohou být obnoveny v jediném příkazu **RECOVER CFSTRUCT**.

Pokud je sdílená datová sada zpráv poškozena, ale náprava nebyla aktivní pro **CFSTRUCT**, nebo je protokol obsahující nejnovější **BACKUP CFSTRUCT** nedostupný nebo nepoužitelný, pak zprávy odložené do této datové sady nelze obnovit. V tomto případě lze pomocí příkazu **RECOVER CFSTRUCT** s parametrem **TYPE**

(PURGE) označit datovou sadu sdílených zpráv jako prázdnou a odstranit všechny zprávy ze struktury, která měla data uložená v této datové sadě.

Když je vydán příkaz **RECOVER CFSTRUCT**, stav datové sady sdílených zpráv se změní z SELHÁNÍ na INRECOVER. Pokud je obnova úspěšně dokončena, stav se automaticky změní na OBNOVENO, jinak se změní zpět na SELHÁNÍ.

Je-li datová sada změněna do stavu ZOTAVEN, informuje správce front, který ji vlastní, o tom, že se nyní může pokusit otevřít datovou sadu a provést zpracování restartu.

Obnova a synchronizační body datové sady sdílených zpráv

Proces obnovy datové sady sdílených zpráv znovu použije změny pro všechny úplné záznamy protokolu až do konce protokolu bez ohledu na synchronizační body.

Pokud byly provedeny změny v synchronizačním bodu, restart nebo zpracování obnovy pro CFSTRUCT může vést k zálohování nepotvrzených požadavků, takže některé z obnovených změn nemusí být ve skutečnosti použity, ale přesto nedošlo k žádným škodám při jejich obnově.

Je také možné, že do struktury byla zapsána nepotvrzená zpráva MQPUT, ale odpovídající data nebyla zapsána do datové sady nebo protokolu (protože dokončení vstupu/výstupu je vynuceno pouze na začátku zpracování synchronizačního bodu). To je neškodné, protože zpracování restartu vrátí záznam zprávy ve struktuře, takže na skutečnosti, že odkazuje na neobnovená data, nezáleží.

Zpracování restartu sdílené datové sady zpráv

Pokud se připojení správce front k systému CFSTRUCT ukončí normálně, zapíše správce front mapu volného prostoru bloků pro každou datovou sadu sdílených zpráv do oblasti kontrolních bodů v rámci datové sady těsně před uzavřením datové sady. Mapu prostoru pak lze znovu načíst při restartu připojení za předpokladu, že ani CFSTRUCT, ani datová sada sdílených zpráv nevyžadují žádné zpracování obnovy před dalším restartem.

Pokud však dojde k nestandardnímu ukončení správce front nebo pokud struktura či datová sada vyžadují zpracování zotavení, je při opětovném spuštění připojení správce front ke struktuře vyžadováno další zpracování k dynamickému opětovnému sestavení prostorové mapy.

Za předpokladu, že nebylo nutné obnovit samotnou datovou sadu, restart správce front jednoduše skenuje aktuální obsah struktury, vyhledá odkazy na data zpráv vlastněná aktuálním správcem front a označí příslušné datové bloky jako vlastněné v prostorové mapě. Ostatní správci front mohou i nadále používat strukturu a číst data vlastněná restartovaným správcem front při opětovném vytváření prostorové mapy.

Restart datové sady sdílených zpráv po zotavení

Pokud bylo nutné obnovit datovou sadu sdílených zpráv ze zálohy, budou všechny přechodné zprávy uložené v datové sadě ztraceny a pokud byla datová sada obnovena s použitím TYPE (PURGE), budou ztraceny všechny zprávy uložené v datové sadě. Až do dokončení zotavení bude datová sada označena jako FAILED nebo INRECOVER, takže jakýkoli pokus o čtení jedné z ovlivněných zpráv z jiného správce front vrátí kód chyby označující, že datová sada je dočasně nedostupná.

Po obnovení datové sady se stav změní na ZOTAVEN, což umožňuje vlastnímu správci front otevřít jej pro zpracování restartu, ale datová sada zůstává pro ostatní správce front nedostupná. Restartování správce front prochází strukturu a znovu sestaví mapu prostoru pro všechny zbývající zprávy. Skenování také zkontroluje zprávy, pro které byla data ztracena, a odstraní je ze struktury (nebo je-li to nutné, označí je jako ztracené, aby byly odstraněny později).

Stav datové sady se automaticky změní z hodnoty OBNOVENO na hodnotu AKTIVNÍ po dokončení tohoto restartu skenování. V tomto okamžiku jej mohou znovu začít používat jiní správci front.

Informace o využití datové sady sdílených zpráv

Příkaz DISPLAY USAGE nyní také zobrazuje informace o prostoru datové sady sdílených zpráv a využití fondu vyrovnávacích pamětí pro všechny aktuálně otevřené datové sady sdílených zpráv. Tato informace se zobrazí, pokud je uvedena buď nová volba TYPE (SMDS), nebo existující volba TYPE (ALL).

Aspekty výkonu a kapacity sdílených dat zpráv

Monitorování využití datové sady

Aktuální procentní část zaplnění každé vlastněné datové sady sdílených zpráv lze zobrazit pomocí příkazu **DISPLAY USAGE** s volbou **TYPE(SMDS)**.

Správce front obvykle automaticky rozbalí datovou sadu sdílených zpráv, jakmile dosáhne 90% zaplnění, za předpokladu, že pro definici SMDS platí volba **DSEXPAND(YES)**. To platí v případě, že je volba SMDS nastavena na hodnotu **DSEXPAND(YES)** nebo volba SMDS je nastavena na hodnotu **DSEXPAND(DEFAULT)** a výchozí volba CFSTRUCT je nastavena na hodnotu **DSEXPAND(YES)**.

Pokud pokus o expanzi selže, protože při vytvoření datové sady nebyla zadána žádná sekundární velikost alokace (zpráva IEC070I s kódem příčiny 203) Správce front opakuje požadavek na rozšíření s použitím potlačení sekundární alokace přibližně 20% aktuální velikosti.

Když je datová sada rozbalena, nové oblasti datové sady jsou formátovány jako součást zpracování rozšíření, což může trvat desítky sekund, nebo dokonce minuty pro velmi velké oblasti. Po dokončení formátování bude nový prostor k dispozici pro použití a katalog bude aktualizován tak, aby zobrazoval nový vysoce používaný řídicí interval.

Pokud jsou nové zprávy vytvářeny velmi rychle, je možné, aby se existující datová sada stala plnou před dokončením zpracování expanze. V tomto případě je jakýkoli požadavek, který nemohl přidělit prostor, dočasně pozastaven, dokud se nedokončí pokus o rozšíření a nový prostor nebude k dispozici pro použití. Pokud bylo rozšíření úspěšné, požadavek se automaticky zopakuje.

Pokud se pokus o rozšíření nezdaří z důvodu nedostatku dostupného prostoru nebo z důvodu dosažení maximálního počtu oblastí, zobrazí se zpráva s informací o příčině selhání. Volba potlačení pro ovlivněný SMDS se automaticky změní na **DSEXPAND(NO)**, aby se zabránilo dalším pokusům o rozšíření. V tomto případě existuje riziko, že se datová sada zaplní. V takovém případě může být zapotřebí další akce, jak je popsáno v tématu [Datová sada se zaplní](#).

Monitorování využití struktury aplikace

Úroveň využití struktury aplikace lze zobrazit pomocí příkazu MVS **DISPLAY XCF, STRUCTURE**, který určuje úplný název struktury aplikace (včetně předpony skupiny sdílení front). Zpráva s odpovědí IXC360I zobrazuje aktuální využití prvků a položek.

Když využití struktury překročí hodnotu **FULLTHRESHOLD** uvedenou v zásadě CFRM, systém vydá zprávu IXC585E a může provádět automatické akce **ALTER**, pokud jsou zadány, což může buď změnit poměr položky k prvku, nebo zvýšit velikost struktury.

Optimalizace velikostí fondu vyrovnávacích pamětí

Každá vyrovnávací paměť ve sdíleném fondu vyrovnávacích pamětí se používá ke čtení nebo zápisu souvislého rozsahu stránek pro jednu zprávu až do velikosti logického bloku. Pokud se zpráva přelévá do dalších bloků, každý rozsah stránek v samostatném bloku vyžaduje samostatnou vyrovnávací paměť.

Vyrovnávací paměti obsahující data zpráv po operaci zápisu nebo čtení jsou uchovány v úložišti a znovu použity s použitím schématu mezipaměti LRU (Lelonejnověji použité), takže požadavek na čtení stejných dat znovu krátce poté nebude muset přejít na disk. To poskytuje významnou optimalizaci při zápisu sdílených zpráv a jejich následném čtení aplikacemi spuštěnými na stejném systému. Pokud jsou zprávy vlastněné jiným správcem front procházeny pro účely výběru, pak jsou načteny, což také eliminuje potřebu znovu načíst zprávu z disku.

To znamená, že počet vyrovnávacích pamětí požadovaných pro každou strukturu aplikace je jeden pro každý souběžný požadavek rozhraní API, který čte nebo zapisuje velké zprávy pro tuto strukturu

aplikace plus určitý počet dalších vyrovnávacích pamětí, které se použijí k uložení dat, ke kterým se nedávno přistupovalo, aby se optimalizovaly následné přístupy pro čtení.

Pro sdílené fondy vyrovnávacích pamětí platí, že pokud nejsou k dispozici dostatečné vyrovnávací paměti, budou požadavky rozhraní API čekat pouze v případě, že vyrovnávací paměť není okamžitě k dispozici. Této situaci je však třeba se vyhnout, protože může způsobit výrazně degradovaný výkon.

Statistika příkazu **DISPLAY USAGE** pro fondy sdílených vyrovnávacích pamětí zobrazuje, zda v aktuálním statistickém intervalu existovala nějaká čekání na vyrovnávací paměť, a také zobrazuje nejnižší počet volných vyrovnávacích pamětí (nebo zápornou hodnotu označující maximální počet podprocesů, které kdykoli čekaly na vyrovnávací paměť), počet vyrovnávacích pamětí, které mají uložená data, a procento případů, kdy požadavek na vyrovnávací paměť úspěšně nalezl uložená data v řetězu LRU ("přístupy LRU"). místo toho, abyste jej museli číst ("LRU chybí")¹.

- Pokud došlo k čekání, počet vyrovnávacích pamětí by měl být zvýšen.
- Pokud existuje mnoho nepoužívaných vyrovnávacích pamětí, může být počet vyrovnávacích pamětí snížen, aby bylo v oblasti k dispozici více paměti pro jiné účely.
- Pokud existuje mnoho vyrovnávacích pamětí obsahujících uložená data, ale podíl čtení, která byla proti uloženým datům použita, je velmi malý, počet vyrovnávacích pamětí může být snížen, pokud by úložiště mohlo být lépe využito pro jiné účely. Počet vyrovnávacích pamětí by však neměl být snížen o více než nejnižší počet volných vyrovnávacích pamětí, protože by mohlo dojít k čekání na spuštění, a měl by být pokud možno dostatečně vysoký, aby nejnižší počet volných vyrovnávacích pamětí byl obvykle výrazně nad nulou.

Odstranění sdílených datových sad zpráv

Příkaz DELETE CFSTRUCT (který je povolen pouze tehdy, když jsou všechny sdílené fronty ve struktuře prázdné a zavřené) neodstraní samotné sdílené datové sady zpráv, ale mohou být odstraněny obvyklým způsobem po dokončení tohoto příkazu. Má-li být stejná datová sada znovu použita jako sdílená datová sada zpráv, musí být nejprve přeformátována, aby byla obnovena do prázdného stavu.

Situace výjimek pro sdílené datové sady zpráv

Existuje řada situací výjimek, které se mohou vyskytnout během normálního používání, i když není přítomna žádná softwarová nebo hardwarová chyba.

Datová sada se naplní

Pokud se datová sada zaplní, ale nelze ji rozbalit, nebo pokud se pokus o rozšíření nezdaří, aplikace používající příslušného správce front k zápisu velkých zpráv do odpovídající struktury aplikace obdrží chybu 2192, MQRC_STORAGE_MEDIUM_FULL (také známý jako MQRC_PAGESET_FULL).

Datová sada se může zaplnit kvůli selhání v aplikaci, která má data zpracovat, což způsobí hromadění velkého množství nevyřízených zpráv. Pokud ano, další rozšíření datové sady bude pouze dočasným řešením a je důležité, aby se aplikace zpracování vrátila co nejdříve.

Pokud lze zpřístupnit více prostoru, lze použít příkaz **ALTER SMDS** k nastavení **DSEXPAND(YES)** nebo **DSEXPAND(DEFAULT)** (za předpokladu, že hodnota YES byla nastavena nebo předpokládána jako předvolba **DSEXPAND** pro definici CFSTRUCT), aby se spustil nový pokus. Pokud však příčinou selhání bylo dosažení maximálního počtu oblastí pro rozšíření, bude nový pokus o expanzi odmítnut se zprávou a parametr **DSEXPAND(NO)** bude znovu nastaven. V tomto případě je jediným způsobem, jak ji dále rozšířit, přerozdělit ji, což zahrnuje její dočasnou nedostupnost, jak je popsáno dále.

Datovou sadu je třeba přesunout nebo znovu přidělit.

Pokud je třeba datovou sadu přesunout nebo rozbalit, ale jinak se používá normálně, lze ji dočasně vyndat, aby bylo možné ji přesunout nebo znovu přidělit. Jakýkoli požadavek rozhraní API, který se pokusí použít datovou sadu, zatímco je nedostupná, obdrží kód příčiny MQRC_DATA_SET_NOT_AVAILABLE.

¹ (Hits / (Hits+Misses))* 100

1. Pomocí příkazu **RESET SMDS** označíte datovou sadu jako **ACCESS (DISABLED)**. To způsobí, že bude uzavřen normálně a bude dealokován všemi aktuálně připojenými správci front.
2. Přesuňte nebo znovu přiřadíte datovou sadu podle potřeby a zkopírujte starý obsah do nově přidělené datové sady, například pomocí příkazu Access Method Services (AMS) **REPRO**.

Nepokoušejte se předformátovat novou datovou sadu před zkopírováním starých dat do ní, protože by to mělo za následek připojení kopírovaných dat na konec formátované datové sady.

3. Pomocí příkazu **RESET SMDS** označíte datovou sadu znovu jako **ACCESS (ENABLED)**, čímž ji znovu použijete.

Pokud je starý obsah menší než velikost nové datové sady, zbytek prostoru bude automaticky naformátován při otevření nové datové sady.

Pokud byl starý obsah větší než velikost nové datové sady, musí správce front procházet zprávy ve struktuře prostředku Coupling Facility a znovu sestavit mapu prostoru, aby se zajistilo, že nebyla ztracena žádná aktivní data. Pokud je nalezen jakýkoli odkaz na datový blok, který je mimo nové oblasti, datová sada je označena jako **STATUS (FAILED)** a musí být opravena nahrazením datové sady jednou ze správné velikosti a buď opětovným zkopírováním staré datové sady, nebo pomocí produktu **RECOVER CFSTRUCT** obnovením trvalých zpráv.

Struktura prostředku Coupling Facility má málo prostoru.

Pokud struktura prostředku Coupling Facility nemá dostatek prostoru, což způsobuje zprávu IXC585E, stojí za to zkontrolovat, zda byla nastavena pravidla odlehčování, aby se zajistilo, že v tomto případě dochází k odlehčování maximálního množství dat. Pokud ne, lze pravidla odlehčování upravit pomocí příkazu **ALTER CFSTRUCT**.

Chybové situace pro sdílené datové sady zpráv

Je třeba si uvědomit řadu problémů, které mohou být způsobeny pouze chybami a nevyskytují se v běžných provozních situacích.

Vlastněnou datovou sadu nelze otevřít

Pokud správce front, který vlastní datovou sadu sdílených zpráv, ji nemůže přiřadit nebo otevřít, nebo pokud nejsou podporovány atributy datových sad, nastaví správce front odpovídající hodnotu stavu **SMDSCONN ALLOCFAIL** nebo **OPENFAIL** a nastaví **SMDSCONN** dostupnost na **AVAIL (ERROR)**. Také nastaví dostupnost SMDS na **ACCESS (SUSPENDED)**. Po odstranění chyby pomocí příkazu **RESET SMDS** nastavte pro spuštění opakovaného pokusu příkaz **ACCESS (ENABLED)** nebo zadejte příkaz **START SMDSCONN** vlastnímu správci front.

Datovou sadu jen pro čtení nelze otevřít

Pokud správce front nemůže přiřadit nebo otevřít datovou sadu sdílených zpráv vlastněnou jiným správcem front a označenou jako **STATUS (ACTIVE)**, předpokládá se, že je to pravděpodobně kvůli specifickému problému s připojením k datové sadě (reprezentované objektem **SMDSCONN**), spíše než kvůli problému se samotnou datovou sadou.

Označí **SMDSCONN** jako **STATUS (ALLOCFAIL)** nebo **STATUS (OPENFAIL)** jako odpovídající a označí dostupnost **SMDSCONN** jako **AVAIL (ERROR)**, aby se zabránilo dalším pokusům o její použití.

Pokud lze problém opravit bez ovlivnění stavu samotné datové sady, použijte příkaz **START SMDSCONN** ke spuštění opakovaného pokusu.

Pokud se problém ukáže jako problém se samotnou datovou sadou, pak lze použít příkaz **RESET SMDS** k označení datové sady jako **STATUS (FAILED)**, dokud nebude obnovena. Po obnovení datové sady akce změny stavu zpět na hodnotu **STATUS (ACTIVE)** způsobí, že ostatní správci front budou upozorněni. Pokud je **SMDSCONN** označeno jako **AVAIL (ERROR)**, automaticky se změní zpět na **AVAIL (NORMAL)**, aby se spustil nový pokus o otevření datové sady.

Záhlaví datové sady je poškozené

Pokud byla datová sada úspěšně otevřena, ale formát informací záhlaví je nesprávný, správce front se zavře a dealokuje datovou sadu a nastaví stav na **STATUS (FAILED)** a dostupnost na **ACCESS (SUSPENDED)**. To umožňuje použití **RECOVER CFSTRUCT** k obnovení obsahu.

Pokud k chybě došlo, protože datová sada obsahovala zbytková data z jiného použití a nebyla následně předformátována, předformátujte datovou sadu a pomocí příkazu **RESET SMDS** změňte stav na **STATUS (RECOVERED)**.

Jinak musí být datová sada obnovena.

Datová sada je neočekávaně prázdná

Pokud správce front otevře datovou sadu, která je označena jako **STATUS (ACTIVE)**, ale zjistí, že je neinicializovaná nebo nově naformátovaná, ale jinak platná, zavře a dealokuje datovou sadu sdílených zpráv, pak nastaví stav na **STATUS (FAILED)** a dostupnost na **ACCESS (SUSPENDED)**.

Datová sada obsahuje trvalé chyby I/O

Pokud má datová sada trvalé chyby I/O po úspěšném zpracování **OPEN**, pravděpodobně potřebuje obnovu. Správce front označí datovou sadu jako **STATUS (FAILED)**, takže všichni aktuálně připojení správci front ji zavřou a dealokují.

Datová sada obsahuje opravitelné chyby I/O

Pokud dojde k hardwarovým problémům s datovou sadou, je možné, že to může vést k opravitelným chybám I/O, které se neodrazí zpět do správce front, ale které způsobí významné snížení výkonu, a také k riziku trvalých chyb I/O v blízké budoucnosti.

V tomto případě může být datová sada pro obnovu vypnuta pomocí příkazu **RESET SMDS**, který ji označí jako **STATUS (FAILED)**. To způsobí, že bude uzavřen a uvolněn všemi správci front, takže může být například přesunut na nový svazek, než bude znovu zpřístupněn.

Je-li datová sada tímto způsobem znepřístupněna, mapa prostoru se neuloží, takže zpracování restartu připojení správce front bude muset skenovat strukturu prostředku Coupling Facility a vyhledat zprávy v datové sadě a znovu sestavit mapu prostoru, než bude možné datovou sadu znovu zpřístupnit. Alternativně, je-li sdílená datová sada zpráv stále použitelná, lze ji znepřístupnit jemněji pomocí příkazu **RESET SMDS** k označení datové sady **ACCESS (DISABLED)**, dokud nebude připravena k opětovnému zpřístupnění.

Obsah datové sady je nesprávný

Správce front nemůže přímo zjistit, že datová sada obsahuje nesprávná data nebo není aktuální, například proto, že svazek obsahující tuto datovou sadu musel být obnoven ze záloh. Provádí však kontroly integrity, kvůli kterým je velmi nepravděpodobné, že by jakékoli takové chyby mohly vést k nesprávným datům zpráv, která jsou zobrazována aplikačními programy.

Pro účely kontroly integrity je před každým blokem zprávy v datové sadě uvedena kopie příslušného ID položky prostředku Coupling Facility, včetně jedinečného časového razítka, které je před předáním dat zprávy uživatelskému programu kontrolováno při každém čtení bloku zprávy. Pokud se předpona bloku zprávy neshoduje s ID záznamu (a záznam prostředku Coupling Facility nebyl ve střední době odstraněn), předpokládá se, že blok zprávy je poškozen a nepoužitelný.

Pokud byla poškozená zpráva trvalá, datová sada je označena jako **STATUS (FAILED)** a obsah struktury musí být obnoven pomocí příkazu **RECOVER CFSTRUCT**. Pokud byla poškozená zpráva dočasná, neexistuje žádný způsob, jak ji obnovit, takže je vydána diagnostická zpráva a příslušná položka zprávy prostředku Coupling Facility je odstraněna.

Není-li při otevření datové sady k dispozici žádná mapa uloženého prostoru, je znovu sestavena skenováním struktury prostředku Coupling Facility pro odkazy na data v datové sadě. Během tohoto skenování provádí správce front řadu akcí:

1. Správce front určuje umístění nejnovější zprávy (pokud existuje), která v současné době v datové sadě zůstává.

2. Správce front poté přečte tuto zprávu z datové sady a zajistí, aby předpona bloku odpovídala ID položky zprávy.

Tyto akce zajistí, že správce front zjistí jakýkoli případ, kdy je datová sada nižší než úroveň, a označí datovou sadu jako SELHÁNÍ. Tato kontrola však toleruje případ, kdy byla datová sada obnovena z předchozí kopie a od té doby nebyly přidány žádné nové zprávy, nebo všechny zprávy přidáné od doby, kdy byla tato kopie následně přečtena a odstraněna.

Chcete-li chránit před daty nižší úrovně v případě, že byla datová sada normálně uzavřena, provede správce front řadu akcí:

1. Správce front uloží kopii časového razítka mapy prostoru do objektu SMDS v rámci produktu Db2, když je datová sada uzavřena normálně.
2. Správce front poté při opětovném otevření datové sady zkontroluje, zda je časové razítko prostorové mapy stejné.

Pokud se časové razítko neshoduje, znamená to, že mohla být použita kopie datové sady nižší úrovně, takže správce front ignoruje existující mapu prostoru a znovu ji sestaví, což bude úspěšné pouze v případě, že ve skutečnosti nebyla ztracena žádná data zprávy.

Poznámka: Tyto kontroly integrity nezaručují zjištění nižší úrovně nebo poškozené datové sady ve všech teoreticky možných případech. Například nezjistí případ, kdy je začátek bloku zprávy platný, ale zbytek dat byl částečně přepsán.

Scénáře obnovy pro sdílené datové sady zpráv

Tento oddíl popisuje scénáře obnovy datové sady sdílených zpráv.

Obnova datové sady, kde nebyla ztracena žádná data

V některých případech lze obnovit správný obsah nezdařené datové sady bez nutnosti skutečné obnovy. Jedním příkladem je situace, kdy datová sada obsahuje zbytková data z předchozího použití a nebyla znovu naformátována, což lze opravit jejím předformátováním. Další případ je, když byla datová sada přesunuta, ale v procesu kopírování dat došlo k chybě, kterou lze opravit opětovným zkopírováním dat správně.

V takových případech lze opravenou datovou sadu znovu zpřístupnit pomocí příkazu **RESET SMDS** k nastavení **STATUS (RECOVERED)**. Pokud je dostupnost momentálně **ACCESS (SUSPENDED)**, automaticky se nastaví zpět na **ACCESS (ENABLED)**.

Když je vlastníci správci front oznámeno, že datová sada byla obnovena, prochází obsah struktury, aby rekonstruoval prostorové mapy, a poté změni stav na **STATUS (ACTIVE)**. Ostatní správci front pak mohou znovu začít číst datovou sadu.

Obnova datové sady s TYPE (NORMAL)

Pokud došlo ke ztrátě obsahu datové sady, ale struktura aplikace byla definována s parametrem **RECOVER (YES)** a jsou k dispozici příslušné protokoly pro zotavení, lze pomocí příkazu **RECOVER CFSTRUCT** obnovit všechny trvalé zprávy uložené ve struktuře včetně trvalých dat zpráv přenášených do datových sad sdílených zpráv. Tento příkaz obnoví aktuální stav pomocí informací zaznamenaných příkazem **BACKUP CFSTRUCT** plus všechny zaprotokolované změny trvalých zpráv od doby zálohování.

Příkaz **RECOVER CFSTRUCT** vždy obnoví všechny trvalé zprávy ve struktuře prostředku Coupling Facility spolu s daty odlehčených zpráv uloženými v adresáři Db2. V případě odlehčených dat uložených ve sdílených datových sadách zpráv je každá datová sada vybrána pro zpracování obnovy pouze v případě, že je již označena jako **STATUS (FAILED)** nebo je-li zjištěno, že je neočekávaně prázdná nebo jinak neplatná při otevření zpracováním obnovy. Jakákoli sdílená datová sada zpráv, která je označena jako aktivní a která projde ověřovacími kontrolami, nemusí být obnovena, protože existující data zprávy jsou již správná, ale záhlaví je aktualizováno, aby označilo, že po obnově bude třeba znovu sestavit mapu uloženého prostoru.

Zpracování obnovy je možné pouze v případě, že byla struktura označena jako neúspěšná, protože úplný obsah struktury je třeba rekonstruovat zpracováním obnovy. Pokud však byla alespoň jedna

datová sada sdílených zpráv označena jako nezdařená, příkaz **RECOVER CFSTRUCT** v případě potřeby automaticky označí strukturu jako neúspěšnou, aby bylo možné pokračovat ve zpracování obnovy.

Zotavení lze provést z libovolného správce front ve skupině sdílení front za předpokladu, že mu byl udělen přístup pro zápis k příslušným datovým sadám.

Jsou zálohovány a protokolovány pouze trvalé zprávy, takže normální zpracování obnovy obnoví všechny trvalé zprávy, ale způsobí, že všechny přechodné zprávy ve struktuře budou ztraceny.

Po dokončení obnovy se každá datová sada, která byla vybrána pro obnovu, automaticky změní na **STATUS (RECOVERED)**, a pokud byla dostupnost **ACCESS (SUSPENDED)**, změní se na **ACCESS (ENABLED)**. Správce front znovu sestaví mapu prostoru pro každou datovou sadu skenováním zpráv v prostředí Coupling Facility a označí datovou sadu jako **STATUS (ACTIVE)**, aby ji bylo možné znovu použít.

Obnova datové sady s TYPE (PURGE)

V případě obnovitelné struktury platí, že pokud byl obsah datové sady ztracen, ale z nějakého důvodu není možné provést obnovu, například protože protokoly pro zotavení nejsou k dispozici nebo by obnova trvala příliš dlouho, lze příkaz **RECOVER CFSTRUCT** použít spolu s příkazem **TYPE (PURGE)**, aby se struktura vrátila do použitelného stavu. Tím se struktura resetuje na prázdný stav a všechny přidružené datové sady se označí jako **STATUS (EMPTY)**.

Odstranění struktury aplikace

Pokud je neobnovitelná struktura aplikace odstraněna pomocí příkazu MVS **SETXCF FORCE** nebo v důsledku selhání struktury, pak při příštím připojení struktury je vydána zpráva CSQE028I, která říká, že struktura byla resetována a všechny existující zprávy byly vyřazeny a všechny existující datové sady jsou také automaticky resetovány na **STATUS (EMPTY)**. Tato akce učiní nezotavitelnou strukturu znovu použitelnou po ztrátě dat buď ve struktuře, nebo v přidružených datových sadách.

Je-li odstraněna obnovitelná struktura aplikace, bude s ní zacházeno stejným způsobem, jako kdyby se struktura nezdařila.

Obnova datové sady se nezdařila

Pokud nelze produkt **RECOVER CFSTRUCT** z nějakého důvodu dokončit, například proto, že datová sada protokolu již není k dispozici, nebo proto, že byl správce front ukončen během probíhajícího zotavení, budou všechny datové sady, pro které bylo zotavení alespoň spuštěno, označeny v záhlaví, aby se zobrazilo, že byl proveden pokus o částečné zotavení, a datová sada bude ponechána ve stavu **STATUS (FAILED)**.

V tomto případě je možné zopakovat původní požadavek na obnovu nebo provést obnovu pomocí produktu **TYPE (PURGE)** a vyřazení existujících dat.

Pokud se pokusíte označit datovou sadu jako **STATUS (RECOVERED)**, aniž byste ji skutečně obnovili, pak při příštím otevření správce front uvidí, že záhlaví označuje neúplnou obnovu, a označí ji znovu jako **STATUS (FAILED)**.

Zotavení z havárie mimo pracoviště

Pro zotavení z havárie mimo server lze trvalé sdílené zprávy znovu vytvořit pouze pomocí protokolů a sdílených objektů Db2 obsahujících definice CFSTRUCT a přidružené informace o stavu SMDS.

Po nastavení tabulek Db2 obsahujících definice lze strukturu aplikace a sdílené datové sady zpráv nastavit jako prázdné. Když se k nim správce front připojí a zjistí, že jsou neočekávaně prázdné, označí je jako nezdařené. Poté lze pomocí jediného příkazu **RECOVER CFSTRUCT** obnovit všechny trvalé zprávy pro všechny ovlivněné struktury.

Příkazy související se SMDS

Toto téma popisuje a poskytuje přístup k příkazům souvisejícím se sdílenými datovými sadami zpráv.

Zobrazte a změňte volby **CFSTRUCT** týkající se velkého odlehčování zpráv (**OFFLOAD** a pravidla odlehčování) a sdílených datových sad zpráv (**DSGROUP**, **DSBLOCK**, **DSBUFS**, **DSEXPA**):

- ZOBRAZENÍ STRUKTURY CFSTRUCT
- DEFINE CFSTRUCT
- ALTER CFSTRUCT
- DELETE CFSTRUCT

Zobrazit **CFSTRUCT** stav týkající se velkého odlehčování zpráv (**OFFLDUSE**):

- ZOBRAZIT STAV CFSTATUS

Zobrazit a změnit volby přepsání datové sady (**DSEXPAND** a **DSBUFS**) pro jednotlivé správce front:

- Zobrazit sadu SMDS
- ALTER SMDS

Zobrazte nebo upravte stav a dostupnost datových sad ve skupině sdílení front:

- DISPLAY CFSTATUS TYPE (SMDS)
- Resetovat SMDS

Zobrazit informace o využití prostoru datové sady SMDS a využití vyrovnávací paměti pro správce front:

- TYP POUŽITÍ ZOBRAZENÍ (SMDS)

Zobrazit nebo upravit stav a dostupnost připojení (**SMDSCONN**) k datovým sadám z jednotlivého správce front:

- ZOBRAZENÍ SMDSCONN
- START SMDSCONN
- STOP SMDSCONN

V případě potřeby zálohujte a obnovte sdílené zprávy, včetně velkých dat zpráv v SMDS:

- ZÁLOŽNÍ STRUKTURA CFSTRUCT
- OBNOVIT STRUKTURU CFSTRUCT

Výhody používání sdílených front

Sdílená fronta umožňuje, aby aplikace IBM MQ byly rozšiřitelné, vysoce dostupné a aby bylo možné implementovat vyrovnávání pracovní zátěže.

Výhody sdílených front

Architektura sdílené fronty, kde klonované servery stahují práci z jedné sdílené fronty, má některé užitečné vlastnosti:

- Je rozšiřitelný přidáním nových instancí serverové aplikace nebo dokonce přidáním nového obrazu z/OS se správcem front (ve skupině sdílení front) a kopií aplikace.
- Je vysoce dostupný.
- Přirozeně provádí vyrovnávání pracovní zátěže *pull* na základě dostupné kapacity zpracování jednotlivých správců front ve skupině sdílení front.

Použití sdílených front pro vysokou dostupnost

V následujících příkladech je uveden způsob použití sdílené fronty ke zvýšení dostupnosti aplikací.

Zvažte scénář systému IBM MQ , ve kterém klientské aplikace spuštěné v síti chtějí vytvářet požadavky na serverové aplikace spuštěné v systému z/OS. Aplikace klienta vytvoří zprávu požadavku a umístí ji do fronty požadavků. Klient pak čeká na odpověď ze serveru, odeslanou do fronty pro odpověď uvedené v deskriptoru zprávy požadavku.

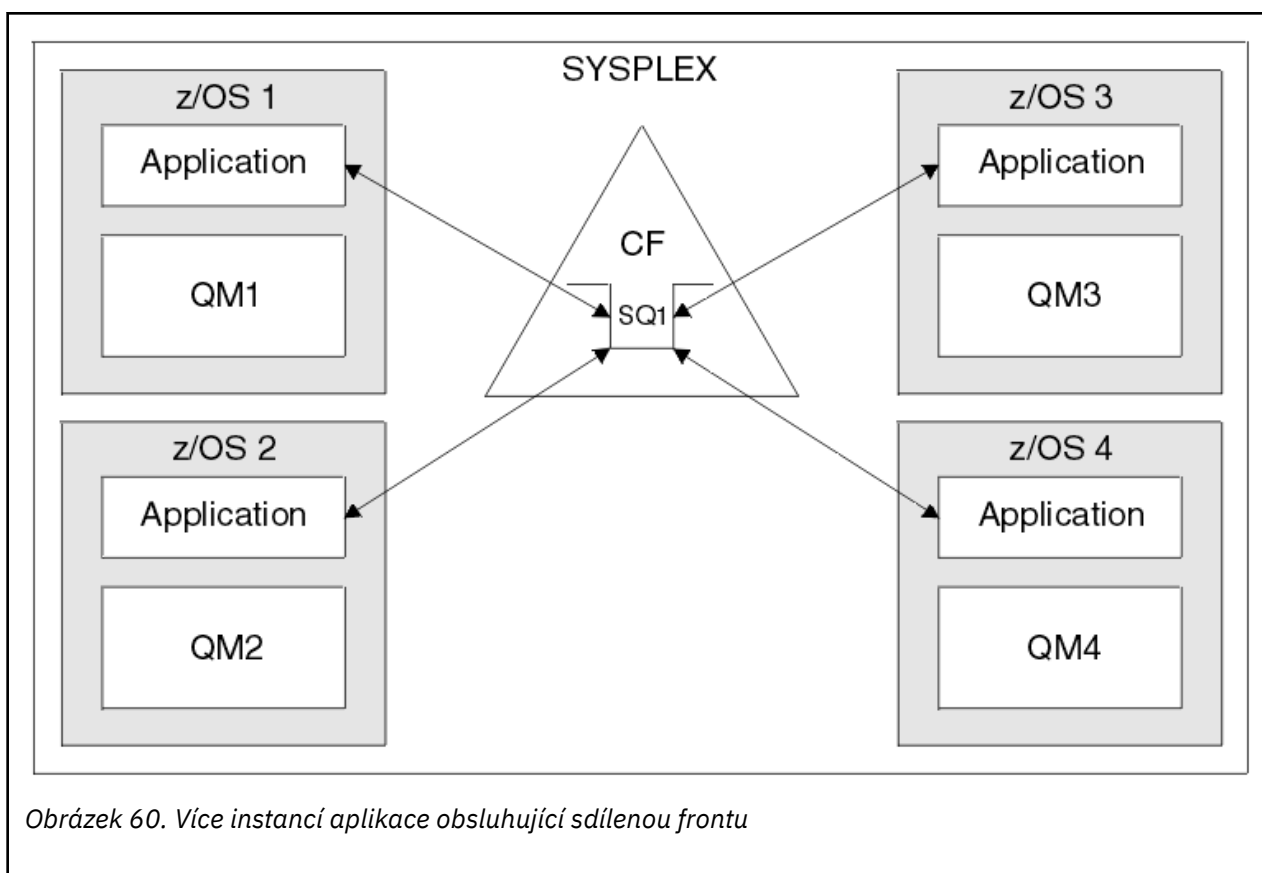
Produkt IBM MQ spravuje přenos zprávy požadavku z klientského počítače do vstupní fronty serveru v systému z/OS a odezvy ze serveru zpět na klienta. Definováním vstupní fronty serveru jako sdílené fronty lze všechny zprávy vložené do fronty načíst v libovolném správci front ve skupině sdílení front. To znamená, že můžete nakonfigurovat správce front na každém obrazu z/OS v prostředí sysplex a připojením všech ke stejné skupině sdílení front může kterýkoli z nich přistupovat ke zprávám ve vstupní frontě serveru.

Zprávy ve vstupní frontě serveru jsou stále k dispozici, a to i v případě, že některý ze správců front skončí nestandardním způsobem nebo že jej musíte zastavit z administrativních důvodů. Můžete převést celý obraz produktu z/OS do režimu offline a zprávy budou stále k dispozici.

Chcete-li využít této dostupnosti zpráv ve sdílené frontě, spusťte instanci serverové aplikace na každém obrazu z/OS v prostředí sysplex, abyste poskytli vyšší kapacitu a dostupnost serverové aplikace, jak ukazuje Obrázek 60 na stránce 181.

Jedna instance serverové aplikace načte zprávu požadavku ze sdílené fronty a na základě obsahu provede její zpracování a vytvoří výsledek, který je odeslán zpět klientovi jako zpráva IBM MQ. Zpráva odpovědi je určena pro správce front pro odpověď a pro odpověď na zprávu uvedenou v deskriptoru zprávy požadavku.

Existuje řada voleb, které můžete použít ke konfiguraci návratové cesty. Další informace o těchto volbách viz [“Skupiny distribuovaného řazení do front a sdílení front”](#) na stránce 200.



Obnova typu peer

V zájmu dalšího rozšíření dostupnosti zpráv ve skupině sdílení front produkt IBM MQ zjistí, zda se jiný správce front ve skupině nestandardně odpojuje od prostředku Coupling Facility, a pokud je to možné, dokončí jednotky práce pro tohoto správce front, které jsou stále nevyřízené. Tato funkce se nazývá *obnova typu peer*.

Předpokládejme, že se správce front ukončí abnormálně v bodě, kde aplikace načetla zprávu požadavku z fronty v synchronizačním bodu, ale dosud nevložila zprávu odpovědi nebo neprovedla transakci. Jiný správce front ve skupině sdílení front zjistil selhání a odvolal probíhající jednotky práce prováděné na správci front, který se nezdařil. To znamená, že zpráva požadavku je vložena zpět do fronty požadavků a je k dispozici pro zpracování jednou z ostatních instancí serveru bez čekání na restartování nezdařeného správce front.

Pokud produkt IBM MQ nemůže vyřešit jednotku práce automaticky, můžete vyřešit sdílenou část ručně, abyste povolili jinému správci front ve skupině sdílení front pokračovat ve zpracování této práce.

Použití paměti paměťové třídy se sdílenými frontami

Použití paměti paměťové třídy (SCM) může být výhodné při použití se sdílenými frontami IBM MQ for z/OS .

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Počítače z13, zEC12a zBC12 umožňují instalaci karet Flash Express. Tyto karty obsahují jednotky SSD (flash solid-state drives). Po instalaci lze úložiště typu flash z karet přidělit jedné nebo více oblastem LPAR, kde se obvykle nazývá SCM.

SCM se nachází mezi skutečným úložištěm a úložným zařízením s přímým přístupem (DASD), a to jak z hlediska latence I/O, tak z hlediska nákladů. Protože SCM nemá pohyblivé části, vykazuje mnohem nižší latence I/O než DASD.

SCM je také mnohem levnější než skutečné skladování. V důsledku toho lze instalovat velké množství úložiště za relativně nízké náklady; například dvojice karet Flash Express obsahuje 1424 GB použitelného úložiště.

Tyto charakteristiky znamenají, že SCM je užitečné, když musí být za krátkou dobu převzato velké množství dat ze skutečného úložiště, protože tato data lze zapsat do SCM mnohem rychleji, než je lze zapsat do DASD. Tento specifický bod může být velmi užitečný při použití struktur seznamu prostředku Coupling Facility (CF), které obsahují sdílené fronty IBM MQ .

Proč se vyplňují struktury seznamu

Je-li definována struktura prostředku CF, je konfigurována s atributem SIZE, který popisuje maximální velikost struktury. Vzhledem k tomu, že struktury CF jsou vždy trvale rezidentní v reálném úložišti, součet atributů SIZE struktur definovaných v prostředku CF by měl být menší než velikost skutečného úložiště přiděleného prostředku CF.

Výsledkem je konstantní tlak, který udržuje hodnotu SIZE libovolné struktury na možné minimální hodnotě, aby se do CF vešlo více struktur. Avšak zajištění toho, aby struktury byly dostatečně velké, aby dosáhly svého účelu, může vést k protichůdnému tlaku, protože příliš malá struktura znamená, že by se mohla zaplnit a narušit aplikace nebo subsystémy, které ji využívají.

Existuje silná potřeba přesně stanovit velikost struktury na základě jejího očekávaného využití. Tuto úlohu je však obtížné provést, protože pracovní zátěž se může v průběhu času měnit a účtování jejich výkyvů není snadné.

Sdílené fronty systému IBM MQ používají k ukládání zpráv struktury seznamu prostředku CF. Produkt IBM MQ volá struktury prostředku CF, které obsahují zprávy a struktury aplikace.

Na struktury aplikace se odkazuje pomocí informací uložených v objektech IBM MQ CFSTRUCT. Je-li do sdílené fronty vložena zpráva o velikosti menší než 63 kB, je zpráva uložena zcela ve struktuře aplikace jako jedna položka seznamu a žádný nebo více prvků seznamu.

Vzhledem k tomu, že sdílené fronty systému IBM MQ používají struktury seznamu, popsané tlaky ovlivňují také sdílené fronty. V tomto případě je maximální počet zpráv, které lze uložit do sdílené fronty, funkcí:

- Velikost zpráv ve frontě

- Maximální velikost struktury
- Počet položek a prvků dostupných ve struktuře

Vzhledem k tomu, že až 512 sdílených front může používat stejnou strukturu a efektivně soutěžit o položky a prvky, to komplikuje věci ještě více.

Fronty IBM MQ se používají pro přenos dat mezi aplikacemi, takže běžnou situací je aplikace vkládající zprávy do fronty, když partnerská aplikace, která by měla tyto zprávy získat, není spuštěna.

Pokud k tomu dojde, počet zpráv ve frontě se v průběhu času zvýší, dokud se nevyskytne jedna nebo více z následujících situací:

- Vkládající aplikace přestane vkládat zprávy.
- Aplikace získávání začne dostávat zprávy.
- Existující zprávy ve frontě začínají vypršením platnosti a jsou z fronty odebrány.
- Fronta dosahuje své maximální hloubky, v takovém případě je vkládající aplikaci vrácen kód příčiny MQRC_Q_FULL.
- Struktura obsahující sdílenou frontu dosahuje své maximální velikosti, nebo prostředek CF obsahující tuto strukturu vyčerpá dostupné úložiště. V obou případech je do vkládající aplikace vrácen kód příčiny MQRC_STORAGE_MEDIUM_FULL.

V posledních třech případech je fronta plná. V tomto bodě vkládající aplikace má problém, protože není kam pro své zprávy jít. Vkládající aplikace obvykle řeší tento problém pomocí jednoho nebo více následujících řešení:

- Opakujte pokus o vložení zprávy, volitelně se zpožděním mezi opakovanými pokusy.
- Umístěte zprávy někam jinam, například do databáze nebo souboru. Ke zprávám lze přistupovat později a vkládat je do fronty jako obvykle.
- Je-li zpráva dočasná, zrušte ji.

Avšak pro některé třídy aplikací, například pro ty, které mají velký objem příchozích zpráv, nebo nemají přístup k systému souborů, nejsou tato řešení praktická. Existuje skutečná potřeba zajistit, aby se fronty nikdy nenaplnily, nebo jsou extrémně nepravděpodobné, a to je zvláště důležité pro sdílené fronty.

Pravidla SMDS a odlehčování

Pravidla odlehčování zavedená v produktu IBM WebSphere MQ 7.1 poskytují způsob, jak snížit pravděpodobnost zaplnění struktury aplikace.

Ke každé struktuře aplikace jsou přidružena tři pravidla, která jsou určena pomocí tří dvojic klíčových slov:

- OFFLD1SZ a OFFLD1TH
- OFFLD2SZ a OFFLD2TH
- OFFLD3SZ a OFFLD3TH

Každé pravidlo určuje podmínky, které musí být splněny, aby byla data zprávy přenesena do mechanismu úložiště, který je přidružen ke struktuře aplikace. V současné době jsou k dispozici dva typy mechanismů ukládání:

- Db2
- Skupina lineárních datových sad VSAM (Virtual Storage Access Method), která IBM MQ volá datovou sadu sdílených zpráv (SMDS).

Následující příklad ukazuje příkaz MQSC pro vytvoření struktury aplikace s názvem LIST1 pomocí příkazu `DEFINE CFSTRUCT`.

Tato struktura obsahuje výchozí pravidla odlehčování a jako mechanismus odlehčování používá SMDS. To znamená, že když je struktura plná z 70% (OFFLD1TH), všechny zprávy, které jsou 32 kB nebo větší (OFFLD1SZ), jsou odloženy do SMDS.


Podobně, když je struktura 80% plná (OFFLD2TH), všechny zprávy, které jsou 4 kB nebo větší (OFFLD2SZ), jsou odlehčeny. Je-li struktura 90% plná (OFFLD3TH), jsou odlehčeny všechny zprávy (OFFLD3SZ).

```
DEFINE CFSTRUCT(LIST1)
CFLEVEL(5)
OFFLOAD(SMDS)
OFFLD1SZ(32K) OFFLD1TH(70)
OFFLD2SZ(4K) OFFLD2TH(80)
OFFLD3SZ(0K) OFFLD3TH(90)
```

Odlehčená zpráva je uložena na médiu pro odlehčování a ukazatel na zprávu je uložen ve struktuře. Zatímco pravidla odlehčování snižují možnost zaplnění struktury tím, že do struktury vkládají méně dat zpráv, protože dochází k ukládání, některá data se stále zapisují do struktury pro každou zprávu. To znamená, že ukazatel na odlehčující zprávu.

Pravidla odlehčování jsou navíc dodávány s náklady na výkon. Zápis zprávy do struktury je poměrně rychlý a je do značné míry ovládan časem stráveným odesláním žádosti o zápis do CF. Skutečný zápis do struktury je rychlý, děje se při skutečné rychlosti skladování.

Zápis zprávy do SMDS je mnohem pomalejší, protože zahrnuje zápis do struktury pro ukazatel zprávy a zápis dat zprávy do SMDS. Tato druhá operace zápisu se provádí s rychlostí DASD a má potenciál přidat latenci. Pokud se jako mechanismus odlehčování používá Db2, náklady na výkon jsou mnohem vyšší.

 *Jak paměť paměťové třídy pracuje s produktem IBM MQ for z/OS*
Přehled použití paměti paměťové třídy (SCM) se sdílenými frontami IBM MQ for z/OS.

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z® pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Prostředek CF (coupling facility), který je na úrovni CFLEVEL 19 nebo vyšší, mu může přidělit SCM. Struktury definované v tomto prostředku CF lze poté nakonfigurovat tak, aby využívaly SCM ke snížení šance na zaplnění struktur (známé jako úplná podmínka struktury). Když se struktura nakonfigurová tak, aby využívala SCM, vyplní za bod určený systémem, prostředek CF začne přesouvat data ze struktury do SCM, což uvolní prostor ve struktuře pro nová data.

Poznámka: Protože se může zaplnit sama SCM, alokace SCM do struktury snižuje pouze pravděpodobnost úplného stavu struktury, ale zcela neodstraňuje možnost výskytu.

Struktura je konfigurována pro použití SCM zadáním klíčových slov **SCMALGORITHM** a **SCMAXSIZE** v zásadě správce prostředků prostředku Coupling Facility (CFRM) obsahující definici této struktury.

Všimněte si, že po zadání těchto klíčových slov a použití zásady CFRM musí být struktura znovu sestavena nebo dealokována, aby se mohla projevit.

Klíčové slovo **SCMALGORITHM**

Vzhledem k tomu, že vstupní/výstupní rychlost SCM je pomalejší než rychlost skutečného úložiště, CF používá algoritmus, který je přizpůsoben očekávanému použití struktury, aby se snížil dopad zápisu do SCM nebo čtení z SCM.

Algoritmus je konfigurován pomocí klíčového slova **SCMALGORITHM** v zásadě CFRM pro strukturu pomocí hodnoty **KEYPRIORITY1**. Všimněte si, že byste měli použít hodnotu **KEYPRIORITY1** pouze se strukturami seznamu používanými sdílenými frontami IBM MQ.

Algoritmus **KEYPRIORITY1** pracuje za předpokladu, že většina aplikací získá zprávy ze sdílené fronty v pořadí podle priority; to znamená, že když aplikace obdrží zprávu, získá nejstarší zprávu s nejvyšší prioritou.

Když se struktura začne zaplňovat za systémem definovanou prahovou hodnotou 90%, prostředek CF začne asynchronně migrovat zprávy, u kterých je nejméně pravděpodobné, že budou přijaty jako další. Jedná se o zprávy s nižšími prioritami, které byly nedávno vloženy do fronty.

Tato asynchronní migrace zpráv ze struktury do SCM se nazývá "předfázování".

Předběžné fázování snižuje náklady na výkon při použití SCM, protože snižuje pravděpodobnost zablokování aplikace během výskytu synchronního vstupu/výstupu do SCM.

Kromě předběžného fázování algoritmus *KEYPRIORITY1* také asynchronně vrací zprávy z SCM a do struktury, když je k dispozici dostatek volného prostoru. V případě algoritmu *KEYPRIORITY1* to znamená, že když je struktura menší nebo rovna 70% zaplnění.

Akt přenesení zpráv z SCM do struktury je znám jako "předběžné načítání".

Předběžné načtení snižuje pravděpodobnost, že se aplikace pokusí získat zprávu, která byla předem fázována do SCM a musela čekat, zatímco prostředek CF synchronně přenáší zprávu zpět do struktury.

Klíčové slovo **SCMMAXSIZE**

Klíčové slovo **SCMMAXSIZE** definuje maximální množství SCM, které může použít struktura. Protože je SCM přidělen struktuře prostředkem CF, když je požadován, je možné určit **SCMMAXSIZE**, který je větší než celkové množství volného SCM, které je k dispozici. Toto je známé jako "nadměrný závazek".

Důležité: Nikdy nepřilížit nepotvrzovat SCM. Pokud tak učiníte, aplikace, které na něj spoléhají, nezískají očekávané chování. Například aplikace IBM MQ používající sdílené fronty mohou získat neočekávané kódy příčiny `MQRC_STORAGE_MEDIUM_FULL`.

Prostředek CF používá různé datové struktury ke sledování svého používání SCM. Tyto datové struktury se nacházejí ve skutečném úložišti, které je přiděleno prostředku CF, a v důsledku toho snižují množství skutečného úložiště, které mohou být použity strukturami. Úložiště používané těmito datovými strukturami je známé jako "rozšířený prostor".

Když je struktura konfigurována s SCM, je malé množství skutečného úložiště přiděleno z prostředku CF do struktury známé jako pevný rozšířený prostor. To je přiděleno i v případě, že struktura nikdy ve skutečnosti nepoužívá žádný SCM. Vzhledem k tomu, že data ze struktury jsou uložena do SCM, bude další dynamický rozšířený prostor přidělen z náhradního reálného úložiště v prostředku CF.

Když jsou data odebrána z SCM, dynamický rozšířený prostor je vrácen do prostředku CF. Rozšířený prostor, buď pevný, nebo dynamický, není nikdy převzat ze skutečného úložiště, které je přiděleno struktuře.

Kromě rozšířeného úložiště, když je struktura nakonfigurována tak, aby používala SCM, se zvyšuje množství řídicího úložiště, které tato struktura používá. To znamená, že struktura seznamu nakonfigurovaná s SCM může obsahovat méně položek a prvků než struktura stejné velikosti bez konfigurace SCM.

Chcete-li porozumět dopadu SCM na nové nebo existující struktury, použijte nástroj [CFSizer](#).

Posledním důležitým bodem, který je třeba poznamenat, je skutečnost, že po přesunutí dat ze struktury do SCM a použití dynamického rozšířeného prostoru nelze strukturu změnit ani ručně, ani automaticky.

To znamená, že množství paměti přidělené struktuře nelze zvýšit nebo snížit, poměr mezi položkou a prvkem, který struktura používá, nelze změnit atd. Aby byla struktura znovu změnitelná, nesmí mít žádná data uložena v SCM a nesmí využívat dynamické rozšířené úložiště.

Proč používat SCM

Nouzové úložiště a zlepšený výkon jsou dva případy použití pro použití SCM s produktem IBM MQ for z/OS.

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z® pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Tato část představuje teorii, která stojí za dvěma možnými scénáři. Další podrobnosti o nastavení scénářů viz:

- [“Nouzové úložiště-základní konfigurace”](#) na stránce 189
- [“Vylepšený výkon-základní konfigurace”](#) na stránce 194

Důležité: Použití SCM se strukturami prostředku CF není závislé na žádné specifické verzi produktu IBM MQ. Scénář nouzového úložiště však pracuje pouze s produktem IBM WebSphere MQ 7.1 a novějším, protože vyžaduje SMDS a pravidla odlehčování.

Nouzové skladování

Nástroj SMDS a odlehčování zpráv lze použít ve spojení s SCM ke snížení pravděpodobnosti vrácení kódu příčiny MQRC_STORAGE_MEDIUM_FULL do aplikace IBM MQ během rozšířeného výpadku.

Přehled

Jedna sdílená fronta je konfigurována ve struktuře aplikace. Vkládající aplikace vkládá zprávy do sdílené fronty. Aplikace provádějící získávání zpráv získává zprávy ze sdílené fronty.

Za normálního běhu se očekává, že hloubka fronty bude téměř nulová, ale obchodní požadavek označuje, že systém musí být schopen tolerovat dvouhodinový výpadek získávající aplikace. To znamená, že sdílená fronta musí být schopna obsahovat dvě hodiny zpráv z vkládající aplikace.

V současné době je tento proces dosažen pomocí výchozích pravidel odlehčování a SMDS, aby byla minimalizována velikost struktury a zároveň sníženy náklady na výkon, které jsou spojeny s odlehčováním.

Očekává se, že rychlost odesílání zpráv do sdílené fronty se v krátkodobém až střednědobém horizontu zdvojnásobí. Ačkoli požadavek, aby byl systém schopen tolerovat dvouhodinový výpadek, stále existuje, v prostředku CF není k dispozici dostatek skutečného úložiště pro zdvojnásobení velikosti struktury.

Vzhledem k tomu, že prostředek CF, který obsahuje strukturu aplikace, je umístěn na počítači zEC12, existuje možnost přidružit ke struktuře dostatečný počet SCM pro uložení dostatečného množství zpráv, aby bylo možné tolerovat dvouhodinový výpadek.

Zvažte, co se stane v určitém časovém období:

1. Zpočátku je systém ve stabilním stavu. Aplikace pro vkládání i získávání jsou spuštěny normálně a hloubka fronty se blíží nule nebo se blíží nule. Výsledkem je, že struktura aplikace je z velké části prázdná.
2. V určitou dobu aplikace získávání trpí neočekávaným selháním a zastaví se. Vkládající aplikace pokračuje ve vkládání zpráv do fronty a struktura aplikace se začne zaplňovat.
3. Poté, co struktura dosáhne 70% zaplnění, jsou splněny podmínky prvního pravidla odlehčování a všechny zprávy s velikostí větší nebo rovnou 32 kB jsou odloženy do SMDS.

Přehled pravidel odlehčování naleznete v části [“Pravidla SMDS a odlehčování”](#) na stránce 183.

4. Vzhledem k tomu, že zprávy jsou i nadále vkládány do sdílené fronty, struktura se nadále vyplňuje (buď z důvodu uložení dat zprávy do struktury, nebo v důsledku ukazatelů na odlehčenou zprávu uloženou ve struktuře).

Když struktura dosáhne 80% zaplnění, začne se používat druhé pravidlo odlehčování a zprávy, které jsou 4 kB nebo větší, budou odloženy do SMDS.

5. Když je struktura plná z 90%, všechny zprávy jsou odloženy do SMDS a do struktury jsou umístěny pouze ukazatele zpráv.

O této době se začne spouštět předfázovací algoritmus a začne přesouvat data ze struktury do SCM. Za předpokladu, že všechny zprávy ve frontě mají stejnou prioritu, nejnovější zprávy jsou předem fázovány.

Vzhledem k tomu, že všechny zprávy jsou nyní přesouvány do SMDS, data přesouvaná do SCM nejsou skutečnými daty zpráv, ale ukazateli na zprávy v SMDS.

V důsledku toho je počet zpráv, které lze uložit na kombinaci struktury, a SCM a SMDS přidružené ke struktuře, velmi velký.

Výkon: Během této fáze výpadku může aplikace uvedení utrpět určitý stupeň snížení výkonu kvůli nutnosti zapisovat do SMDS. V tomto případě by použití SCM nemělo být omezujícím faktorem pro uvedení aplikace z hlediska výkonu. SCM poskytuje další prostor, aby se zabránilo zaplnění struktury.

6. Nakonec je aplikace získávání znovu k dispozici a výpadek je u konce.

Struktura však stále používá SCM. Aplikace provádějící získávání začne číst zprávy z fronty a nejprve získá nejstarší zprávy s nejvyšší prioritou.

Vzhledem k tomu, že tyto zprávy byly napsány před tím, než se struktura začala zaplňovat, vycházejí zcela ze skutečné skladovací části struktury.

7. Vzhledem k tomu, že struktura začíná být prázdná, jde pod prahovou hodnotu, při které je aktivní předběžné fázování, a proto se předběžné fázování zastaví.
8. Využití struktury se sníží pod bod, ve kterém se pravidla odlehčování projeví, takže zprávy již nebudou odloženy do SMDS, pokud nebudou větší než 63 kB.

V tuto chvíli začne algoritmus předběžného načtení přesouvat data z SCM do struktury. Vzhledem k tomu, že aplikace provádějící získávání získává zprávy z fronty v pořadí očekávaném algoritmy SCM, jsou zprávy předávány dříve, než je aplikace provádějící získávání potřebuje.

Výsledkem je, že aplikace získávání nemusí nikdy čekat na synchronně doručené zprávy z SCM.

9. Vzhledem k tomu, že aplikace provádějící získávání pokračuje v přesouvání dolů ve frontě, začne načítat zprávy, které byly odloženy do SMDS.
10. Nakonec je systém opět ve stabilním stavu. V SCM nebo SMDS nejsou uloženy žádné zprávy a hloubka fronty se blíží nule.

Zlepšený výkon

Tento scénář popisuje použití SCM ke zvýšení počtu zpráv, které lze uložit do sdílené fronty, aniž by vznikly náklady na výkon při používání SMDS.

Popis

V tomto scénáři vkládající a získávající aplikace komunikují prostřednictvím sdílené fronty, která je uložena ve struktuře aplikace.

Vkládající aplikace má tendenci běžet v rozpadech, když dává velký počet zpráv v krátkém čase. Pak v delším časovém období nevytvoří žádné zprávy.

Aplikace provádějící získávání postupně zpracovává každou zprávu a provádí na každé z nich složité zpracování. V důsledku toho je většina času, kdy je hloubka fronty nulová, s výjimkou okamžiku, kdy se začne spouštět vkládající aplikace, kdy se hloubka fronty začne zvětšovat s tím, jak jsou zprávy vkládány rychleji, než jsou získávány.

Hloubka fronty se zvyšuje, dokud se nezastaví vkládající aplikace, a aplikace provádějící získávání má dostatek času na zpracování všech zpráv ve frontě.

Notes:

1. V tomto scénáři je klíčovým faktorem výkon. Zprávy odesílané do fronty jsou vždy menší než 63 kB, a proto není nutné je do SMDS odesílat.
2. Velikost struktury aplikace byla nastavena tak, aby byla dostatečně velká na to, aby obsahovala všechny zprávy, které na ni budou umístěny vkládající aplikací v jediném "rozpadu".
3. Všechna pravidla odlehčování musí být zakázána, aby se zprávy neodesílaly do SMDS, a to ani v případě, že se struktura začne zaplňovat. Je to proto, že náklady na výkon, které jsou spojeny se zapisováním zpráv do SMDS a čtením zpráv z SMDS, jsou považovány za nepřijatelné.

V průběhu času se musí počet zpráv, které vkládající aplikace odešle do shluku, zvýšit o několik řádů. Vzhledem k tomu, že aplikace provádějící získávání musí každou zprávu zpracovat postupně, počet zpráv ve frontě se zvýší na místo, kde se struktura naplní.

V tomto bodě obdrží vkládající aplikace při vkládání zprávy kód příčiny (MQRC_STORAGE_MEDIUM_FULL) a operace vložení se nezdaří. Vkládající aplikace může pouze krátce tolerovat období, kdy není schopna vkládat zprávy do fronty. Pokud je období příliš dlouhé, aplikace skončí.

Za předpokladu, že nemáte čas nebo máte k dispozici dovednosti k přepsání vkládající aplikace nebo získání aplikace, tento problém má tři možná řešení:

1. Zvětšete velikost struktury aplikace.
2. Přidejte pravidla odlehčování do struktury aplikace tak, aby se zprávy předávaly do SMDS, jakmile se fronta začne zaplňovat.
3. Přidruzte SCM ke struktuře.

První řešení je rychle implementováno, ale na prostředku CF není k dispozici dostatek skutečného úložiště.

Druhé řešení může být také rychle implementováno, ale dopad na výkon odlehčování do SMDS je považován za příliš významný pro použití této volby.

Třetí řešení, které přidružuje SCM ke struktuře, poskytuje přijatelnou rovnováhu mezi náklady a výkonem.

Přidružení SCM ke struktuře vede k vyššímu využití reálného úložiště v prostředku CF kvůli rozšíření úložiště, které používá operace get. Skutečné množství skutečného úložiště však bude menší než množství použité v první volbě.

Dalším aspektem jsou náklady na SCM. Tato cena je však mnohem levnější než skutečné skladování. Tyto faktory se kombinují, aby třetí možnost byla levnější než první možnost.

Třebaže třetí volba, potenciálně, nemusí fungovat stejně jako první volba, algoritmy před načtením a před fázováním používané prostředkem CF se mohou kombinovat, aby byly rozdíly ve výkonu přijatelné, nebo v některých případech zanedbatelné.

Jistě výkon může být mnohem lepší než použití SMDS k odlehčování zpráv.

Zvažte, co se stane v určitém časovém období:

1. Na počátku je aplikace provádějící získávání aktivní a čeká na doručení zpráv do sdílené fronty. Vkládající aplikace není aktivní a sdílená fronta je prázdná.
2. V určitou dobu se vkládající aplikace stane aktivní a začne vkládat velký počet zpráv do sdílené fronty. Aplikace provádějící získávání začne dostávat zprávy, ale hloubka fronty se rychle začne zvyšovat, protože aplikace provádějící získávání je pomalejší než aplikace provádějící vkládání.

V důsledku toho se začne zaplňovat struktura aplikace.

3. Jak čas roste, vkládající aplikace je stále aktivní. Aplikační struktura se naplní přibližně na 90%.

To je, když algoritmus před fázováním SCM začne přesouvat zprávy ze struktury do SCM, čímž uvolní prostor ve struktuře.

Vzhledem k tomu, že aplikace provádějící získávání získá nejstarší zprávy s nejvyšší prioritou z fronty jako první, vždy získává zprávy ze struktury a nemusí čekat na synchronně přenesené zprávy ze SCM do struktury.

4. Vkládající aplikace je stále aktivní a vkládá zprávy do sdílené fronty. Aplikace však nikdy neobdrží kód příčiny MQRC_STORAGE_MEDIUM_FULL, protože v SCM existuje dostatek prostoru pro uložení všech zpráv, které se nevejdou do struktury.

5. Nakonec se vkládající aplikace zastaví, protože nemá další zprávy, které by bylo možné vložit.

Předfázovací algoritmus se zastaví, protože struktura klesne pod 90% používaných a aplikace získávání pokračuje ve zpracování zpráv ve frontě.

6. Vzhledem k tomu, že aplikace získávání začne uvolnit prostor ve struktuře, začne algoritmus předběžného načtení vracet zprávy z SCM zpět do struktury.

Vzhledem k tomu, že získávání aplikací zpracovává zprávy v pořadí očekávaném algoritmem předběžného načtení, nebude tato aplikace nikdy blokována při čekání na synchronně přenesená data zprávy z SCM do struktury.

7. Nakonec aplikace provádějící získávání zpracuje všechny zprávy ve sdílené frontě a počká, dokud nebude k dispozici další zpráva. Struktura a SCM jsou prázdné pro zprávy.

z/OS Nouzové úložiště-základní konfigurace

Jak nastavíte základní scénář pro nouzové úložiště na systému IBM MQ.

Informace o této úloze

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Nástroj SMDS a odlehčování zpráv lze použít ve spojení s SCM ke snížení pravděpodobnosti vrácení kódu příčiny MQRC_STORAGE_MEDIUM_FULL do aplikace IBM MQ během rozšířeného výpadku.

Váš podnik má například aplikaci, která vkládá zprávy do fronty, a aplikaci, která získává zprávy z fronty. Za normálního běhu očekáváte, že se hloubka fronty bude blížit nule, ale obchodní požadavek označuje, že systém bude schopen tolerovat dvouhodinový výpadek aplikace, která získá zprávy.

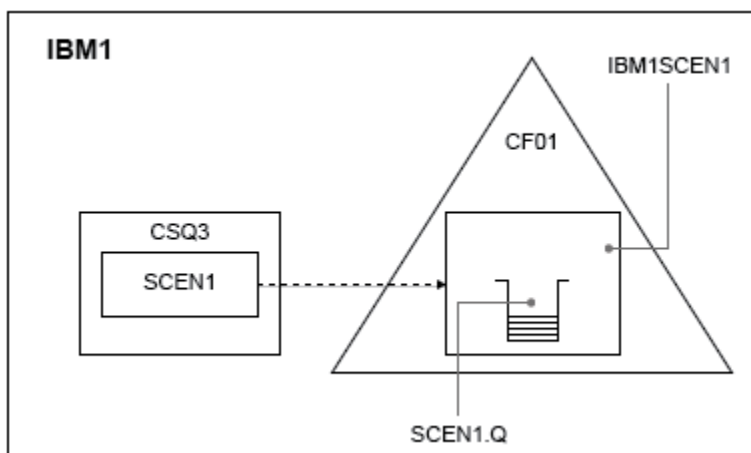
To znamená, že používaná sdílená fronta musí být schopna obsahovat dvě hodiny zpráv z vkládající aplikace. V současné době toho dosáhnete pomocí výchozích pravidel odlehčování a SMDS.

Očekáváte, že se rychlost odesílání zpráv do sdílené fronty v krátkodobém až střednědobém horizontu zdvojnásobí. Ačkoli váš požadavek, aby byl systém schopen tolerovat dvouhodinový výpadek, stále existuje, v prostředku CF není k dispozici dostatek skutečného úložiště pro zdvojnásobení velikosti struktury. Vzhledem k tomu, že prostředek CF obsahující strukturu aplikace je umístěn na počítači zEC12, máte možnost přidružit ke struktuře dostatečný počet SCM pro uložení dostatečného množství zpráv, aby bylo možné tolerovat dvouhodinový výpadek.

Tento počáteční scénář používá:

- Skupina sdílení front IBM1, která obsahuje jednoho správce front CSQ3. Kromě struktury administrace definovala skupina sdílení front jednu strukturu aplikace SCEN1.
- Prostředek CF (Coupling Facility) CF01, ve kterém je struktura aplikace SCEN1 uložena jako struktura IBM1SCEN1. Tato struktura má maximální velikost 1 GB.
- Jedna sdílená fronta SCEN1.Q, kterou používá struktura aplikace.

Tato konfigurace je ilustrována v souboru [Obrázek 61](#) na stránce 189.



Obrázek 61. Základní konfigurace

Dále předpokládejme, že správce front CSQ3 je již jediným členem skupiny sdílení front IBM1.

Definici struktury IBM1SCEN1 musíte přidat do zásady správce prostředků prostředí Coupling Facility (CFRM). Pro zjednodušení je struktura definována tak, aby ji bylo možné vytvořit pouze v jednom prostředí Coupling Facility, CF01, zadáním hodnoty PREFLIST (CF01).



Upozornění: Chcete-li povolit vysokou dostupnost ve vašem produkčním systému, měli byste zahrnout alespoň dva prostředky CF do PREFLIST pro jakékoli struktury, které používá produkt IBM MQ.

Postup

1. Aktualizujte zásadu CFRM pomocí následujícího příkazu:

```
SETXCF START ,POLICY ,TYPE=CFRM ,POLNAME=IBM1SCEN1
```

Ukázková zásada CFRM pro strukturu IBM1SCEN1:

```
STRUCTURE
NAME (IBM1SCEN1)
SIZE (1024M)
INITSIZE (512M)
ALLOWAUTOALT (YES)
FULLTHRESHOLD (85)
PREFLIST (CF01)
ALLOWREALLOCATE (YES)
DUPLEX (DISABLED)
ENFORCEORDER (NO)
```

2. Pomocí následujícího příkazu ověřte, zda byla struktura správně vytvořena:

```
D XCF ,STR ,STRNAME=IBM1SCEN1
```

V tomto bodě nebyla vaše struktura, zobrazená řádkem STATUS , přidělena skupině sdílení front.

3. Nakonfigurujte produkt IBM MQ tak, aby používal strukturu definovanou v zásadě CFRM.

- a. K vytvoření objektu IBM MQ CFSTRUCT použijte příkaz DEFINE CFSTRUCT s názvem struktury SCEN1 :

```
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. Ověřte strukturu pomocí příkazu DISPLAY CFSTRUCT .
- c. Definujte sdílenou frontu SCEN1 . Q pro použití struktury SCEN1 pomocí následujícího příkazu MQSC:

```
DEFINE QLOCAL(SCEN1.Q) QSGDISP(SHARED) CFSTRUCT(SCEN1) MAXDEPTH(999999999)
```

4. Pomocí volby IBM MQ Explorer vložte jednu zprávu do fronty SCEN1 . Q a zprávu znovu vypněte.
5. Zadáním následujícího příkazu zkontrolujte, zda je struktura nyní přidělena:

```
D XCF ,STR ,STRNAME=IBM1SCEN1
```

Zkontrolujte výstup z příkazu, že STATUS řádek zobrazuje ALLOCATED.

Výsledky

Vytvořili jste základní konfiguraci. Nyní můžete získat představu o výkonu úrovně baseline vaší konfigurace pomocí libovolné metody, kterou vyberete.

Jak pokračovat dále

[Přidat SMDS a SCM do počáteční struktury](#)

Související pojmy

[“Použití paměti paměťové třídy se sdílenými frontami” na stránce 182](#)

Použití paměti paměťové třídy (SCM) může být výhodné při použití se sdílenými frontami IBM MQ for z/OS .

z/OS Přidání SMDS a SCM do počáteční struktury

Jak přidáte SMDS a SCM pro nouzové úložiště na systému IBM MQ.

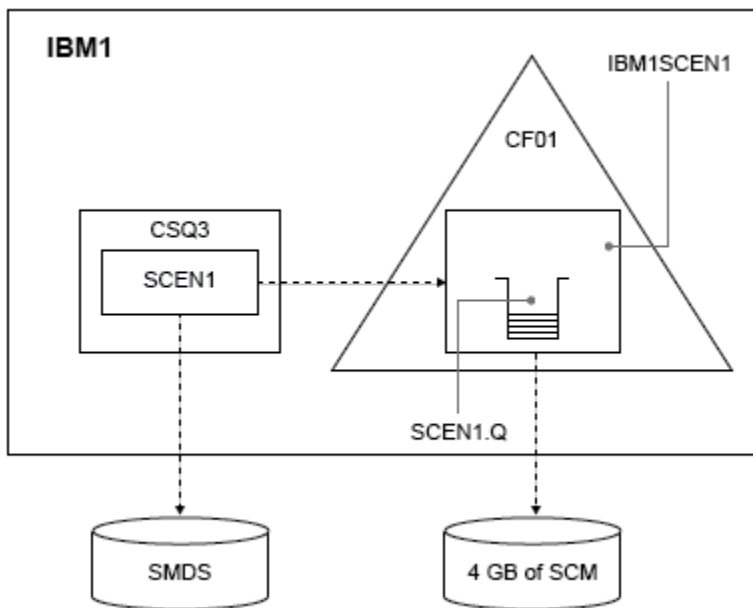
Informace o této úloze

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší strukturu, nebo odlehčovat zprávy SMDS.

Tato část úlohy používá základní konfiguraci popsanou v části “Nouzové úložiště-základní konfigurace” na stránce 189. Scénář popisuje přidání sdílených datových sad zpráv (SMDS) a poté SCM do počáteční struktury.

Tato konečná konfigurace je znázorněna v souboru [Obrázek 62](#) na stránce 191.



Obrázek 62. Konfigurace přidání SMDS a SCM pro nouzové úložiště

Postup

1. Vytvořte datovou sadu SMDS, kterou používá struktura aplikace SCEN1 , úpravou ukázkového JCL **CSQ4SMDS** , jak je uvedeno:

```
//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.SCSQAUT,DISP=SHR
//SYSUT1 DD DISP=OLD,DSN=CSQSMDS.SCEN1.CSQ3.SMDS
//SYSPRINT DD SYSOUT=*
```

2. Zadááním příkazu `ALTER CFSTRUCT` změňte strukturu aplikace SCEN1 tak, aby pro odlehčování používala SMDS a implementovala výchozí pravidla odlehčování:

```
ALTER CFSTRUCT(SCEN1) OFFLOAD(SMDS) OFFLD1SZ(32K) OFFLD2SZ(4K) OFFLD3SZ(0K)
DSGROUP('CSQSMDS.SCEN1.*.SMDS') DSBLOCK(1M)
```

Všimněte si následujícího:

- Protože SCEN1.Q je jedinou sdílenou frontou ve struktuře aplikace SCEN1, byla hodnota **DSBLOCK** nastavena na 1M, což je nejvyšší možná hodnota. To by mělo být neefektivnější nastavení pro náš scénář.
- Vzhledem k tomu, že zprávy odeslané vkládající aplikací jsou 30 kB, odlehčování do SMDS se nespustí, dokud nebude splněno druhé pravidlo odlehčování, když je struktura plná z 80%.

3. Spusťte testovací aplikaci znovu.

Všimněte si zvýšeného úložiště zpráv ve frontě.

4. Přidejte 4 GB SCM do struktury IBM1SCEN1 provedením následujícího postupu:

- a) Zkontrolujte, kolik SCM je nainstalováno a přiděleno pro CF01, zadáním následujícího příkazu:

```
D CF,CFNAME=CF01
```

- b) Zkontrolujte obrázky STORAGE - CLASS MEMORY v sekci STORAGE CONFIGURATION zobrazeného výstupu, abyste viděli dostupné úložiště.
- c) Aktualizujte zásadu CFRM pomocí klíčových slov SCMMAXSIZE a SCMALGORITHM, jak je uvedeno:

```
STRUCTURE
NAME(IBM1SCEN1)
SIZE(1024M)
INITSIZE(512M)
ALLOWAUTOALT(YES)
FULLTHRESHOLD(85)
PREFLIST(CF01)
ALLOWREALLOCATE(YES)
DUPLEX(DISABLED)
ENFORCEORDER(NO)
SCMMAXSIZE(4G)
SCMALGORITHM(KEYPRIORITY1)
```

5. Aktivujte zásadu CFRM zadáním následujícího příkazu:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME=polname
```

6. Znovu sestavte strukturu IBM1SCEN1.

Tuto proceduru musíte provést, protože struktura byla přidělena, když jste provedli předchozí změny.

Chcete-li znovu sestavit strukturu, zadejte následující příkaz:


```
SETXCF START,REBUILD,STRNM=IBM1SCEN1
```

Výsledky

Úspěšně jste přidali SCM do své konfigurace.

Jak pokračovat dále

Optimalizujte výkon svého systému. Další informace viz [“Optimalizace využití paměti paměťové třídy” na stránce 193](#).

Optimalizace využití paměti paměťové třídy

Jak zlepšíte využití paměti paměťové třídy (SCM).

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Spusťte následující příkaz:

```
D XCF,STR,STRNAME=IBM1SCEN1
```

Vzhledem k tomu, že struktura již byla plná s daty zpráv, kvůli předchozím testům část nového sestavení zahrnovala předfázování některých zpráv ze struktury do SCM. Tento proces byl zahájen pomocí předchozího příkazu.

Výstup z tohoto příkazu vytvoří například:

```
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 : IXCL0053
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
```

Všimněte si následujícího z výstupu příkazu:

- Produkt STORAGE_CLASS MEMORY poskytuje potvrzení, že do struktury byla přidána hodnota **MAXIMUM** 4096 MB SCM.
- Obrázek ALLOCATED pro množství souboru STORAGE-CLASS MEMORY použitého pro předfázování. Nyní je ve struktuře volný prostor, kde nebyl žádný před přidáním SCM.
- Množství AUGMENTED SPACE použitého ke sledování využití SCM.

- Bod, ve kterém předfázovací algoritmus začne přesouvat data ze struktury do SCM, je, když je struktura plná z 90%. To je označeno nekonfigurovatelnou vlastností **SCMHIGHTHRESHOLD** .
- Bod, pod kterým algoritmus předběžného načítání začne přesouvat data z SCM do struktury, je, když je struktura plná 70%. To je označeno nekonfigurovatelnou vlastností **SCMLOWTHRESHOLD** .

Nyní můžete testovat různé způsoby optimalizace použití SCM. Všimněte si následujícího:

- Po použití SCM k uložení zpráv nemůžete změnit strukturu, dokud neodeberete všechna data z SCM.

V tomto případě to znamená, že poměr mezi položkou a prvkem je zmrazen na hodnotě, která byla zavedena při prvním použití SCM. Před tím, než algoritmus před fázováním začne přesouvat data do SCM, musíte pečlivě zajistit, aby byla struktura ve stavu, který požadujete.

- Je aktuální velikost struktury správná před použitím SCM?

Například jste zvýšili hodnotu **INITSIZE** z 512 MB na velikost 1 GB?

Pokud tak neučiníte, je možné, že i když jste povolili automatickou změnu struktury, algoritmus před fázováním začne přesouvat data do SCM, než bude mít změna možnost začít. V důsledku toho je struktura zmrazena pomocí 512 MB skutečného úložiště.

- Je poměr mezi položkou a prvkem správný před použitím SCM?

Cílem tohoto scénáře je zvýšit počet odlehčených ukazatelů zpráv, které mohou být uloženy ve struktuře a SCM jako celku, a zachovat co nejvíce zpráv v úložišti struktury. Přístup k těmto zprávám je rychlejší než přístup ke zprávám v SMDS.

Proto musíte mít strukturu, která začíná poměrem položky k prvku, který je vhodný pro ukládání zpráv, a poté přechází na poměr, který je vhodný pro ukládání ukazatelů zpráv před prvním spuštěním algoritmu přednastavení. Tohoto přechodu lze částečně dosáhnout použitím pravidel odlehčování IBM MQ .

Změňte pravidla odlehčování zadáním následujícího příkazu:

```
ALTER CFSTRUCT(SCEN1) OFFLD1SZ(0K)
```

Možná budete muset provést několik spuštění, abyste optimalizovali poměry mezi položkou a prvkem.

Následující tabulka zobrazuje možná zlepšení počtu zpráv vložených do fronty během různých fází scénáře nouzového úložiště.

<i>Tabulka 18. Porovnání výsledků pro scénář nouzového úložiště</i>		
Popis testu	Počet zpráv	Doba zaplnění fronty (sekundy)
Základní konfigurace	27,850	3.2
SMDS s výchozími pravidly odlehčování	205 000	158
SCM s výchozími pravidly odlehčování	828,610	469
SCM s upravenými pravidly odlehčování	1,135,775	679

Poslední řádek v tabulce ukazuje, že úprava pravidel odlehčování měla požadovaný účinek.

Musíte prozkoumat svůj systém, abyste zjistili, zda se můžete v těchto číslech jakýmkoli způsobem zlepšit. Například vám může dojít dostupné úložiště SMDS. Pokud můžete přidělit více úložiště SMDS, měli byste být schopni poměrně výrazně zvýšit počet zpráv ve frontě.

 **Vylepšený výkon-základní konfigurace**

Způsob nastavení základního scénáře pro zlepšení výkonu pomocí sdílených front v systému IBM MQ.

Informace o této úloze

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

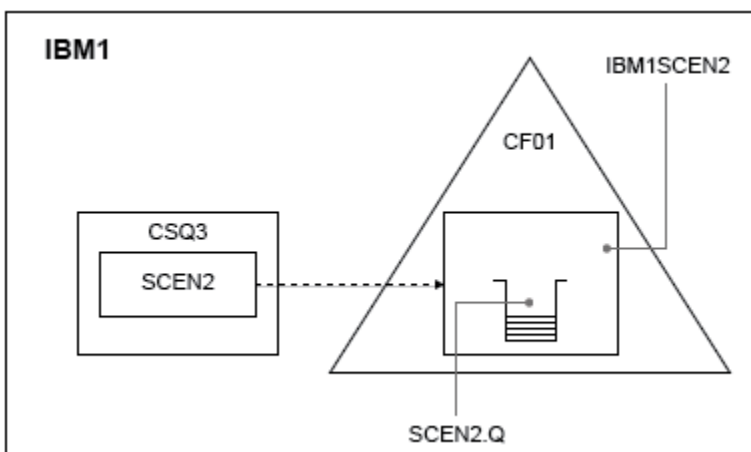
Jako alternativu byste měli buď použít větší struktury, nebo odlehčovat zprávy SMDS.

Tento scénář popisuje použití SCM ke zvýšení počtu zpráv, které lze uložit do sdílené fronty, aniž by vznikly náklady na výkon při používání SMDS.

Tento počáteční scénář je velmi podobný tomu, který se používá pro nouzové úložiště, a používá:

- Skupina sdílení front IBM1, která obsahuje jednoho správce front CSQ3. Kromě struktury administrace definovala skupina sdílení front jednu strukturu aplikace SCEN2.
- Prostředek CF (Coupling Facility) CF01, ve kterém je struktura aplikace SCEN2 uložena jako struktura IBM1SCEN2. Tato struktura má maximální velikost 2 GB.
- Jedna sdílená fronta SCEN2.Q, která je konfigurována pro použití struktury aplikace.

Tato konfigurace je ilustrována v souboru [Obrázek 63](#) na stránce 195.



Obrázek 63. Základní konfigurace

Dále předpokládejme, že správce front CSQ3 je již jediným členem skupiny sdílení front IBM1.

Definici struktury IBM1SCEN2 musíte přidat do zásady správce prostředků prostředku Coupling Facility (CFRM). Pro zjednodušení je struktura definována tak, aby ji bylo možné vytvořit pouze v jednom prostředku Coupling Facility, CF01, zadáním hodnoty PREFLIST(CF01).

Ukázková zásada CFRM pro strukturu IBM1SCEN2:

```
STRUCTURE
NAME(IBM1SCEN2)
SIZE(2048M)
INITSIZE(2048M)
ALLOWAUTOALT(YES)
FULLTHRESHOLD(85)
PREFLIST(CF01)
ALLOWREALLOCATE(YES)
DUPLEX(DISABLED)
ENFORCEORDER(NO)
```

Klíčová slova **INITSIZE** i **SIZE** mají hodnotu 2048M, takže velikost struktury nelze změnit.

Postup

1. Aktualizujte zásadu CFRM pomocí následujícího příkazu:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME=IBM1SCEN2
```

2. Pomocí následujícího příkazu ověřte, zda byla struktura správně vytvořena:

```
D XCF,STR,STRNAME=IBM1SCEN2
```

Zadáním předchozího příkazu získáte následující výstup:

```
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
```

V tomto bodě nebyla vaše struktura, zobrazená řádkem STATUS , přidělena skupině sdílení front.

3. Nakonfigurujte produkt IBM MQ tak, aby používal strukturu definovanou v zásadě CFRM.
 - a. Pomocí příkazu DEFINE CFSTRUCT s názvem struktury SCEN2 vytvořte objekt IBM MQ CFSTRUCT.

```
DEFINE CFSTRUCT(SCEN2)
CFCONLOS(TOLERATE)
CFLEVEL(5)
DESCR('Structure for SCM scenario 2')
RECOVER(NO)
RECAUTO(YES)
OFFLOAD(DB2)
OFFFLD1SZ(64K) OFFFLD1TH(70)
OFFFLD2SZ(64K) OFFFLD2TH(80)
OFFFLD3SZ(64K) OFFFLD3TH(90)
```

- b. Zkontrolujte strukturu pomocí příkazu DISPLAY CFSTRUCT .
- c. Definujte sdílenou frontu SCEN2.Q pro použití struktury SCEN2 pomocí následujícího příkazu MQSC:

```
DEFINE QLOCAL(SCEN2.Q) QSGDISP(SHARED) CFSTRUCT(SCEN2) MAXDEPTH(999999999)
```

4. Pomocí programu IBM MQ Explorer vložte jednu zprávu do fronty SCEN2.Q a zprávu znovu vypněte.
5. Zadáním následujícího příkazu zkontrolujte, zda je struktura nyní přidělena:

```
D XCF,STR,STRNAME=IBM1SCEN2
```

Zkontrolujte výstup z příkazu, jehož část je zobrazena, a ujistěte se, že STATUS řádek zobrazuje 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
```

Dále si poznamenejte hodnoty polí v sekci SPACE USAGE:

- položek
- PRVKY
- Systém EMCS
- ZÁMKY

Následuje příklad hodnot:

SPACE USAGE	IN-USE	TOTAL	%
ENTRIES:	344686	345242	99
ELEMENTS:	6548455	6548467	99
EMCS:	2	780318	0
LOCKS:	1024		

Výsledky

Vytvořili jste základní konfiguraci. Nyní můžete získat představu o výkonu úrovně baseline vaší konfigurace pomocí libovolné metody, kterou vyberete.

Jak pokračovat dále

Měli byste otestovat základní scénář. Jako příklad můžete použít následující tři aplikace, spustit aplikace v uvedeném pořadí a spustit je souběžně.

1. Použít aplikaci PCF k vyžádání aktuální hloubky (**CURDEPTH**) pro hodnotu SCEN2 . Q každých pět sekund. Výstup lze použít k zakreslení hloubky fronty v průběhu času.
2. Jeden podproces, který opakovaně získává zprávy z produktu SCEN2 . Q, používá příkaz get s nekonečným čekáním. Chcete-li simulovat zpracování zpráv, které byly odebrány, aplikace provádějící získávání se pozastaví na čtyři milisekundy na každých deset zpráv, které odebrala.
3. Jedna vkládající aplikace s podporou podprocesů vkládá do produktu SCEN2 . Q celkem jeden milion dočasných zpráv o velikosti 4 kB. Tato aplikace není mezi vložení jednotlivých zpráv pozastavena, takže jsou zprávy SCEN2 . Q umístěny rychleji, než je může získat získávající aplikace.

V důsledku toho se při spuštění vkládající aplikace zvýší hloubka parametru SCEN2 . Q .

Je-li vyplněna struktura IBM1SCEN2 a vkládající aplikace obdrží kód příčiny MQRCS_STORAGE_MEDIUM_FULL, vkládající aplikace přejde do režimu spánku po dobu pěti sekund před pokusem o vložení další zprávy do fronty.

Výsledky aplikace CURDEPTH můžete zakreslit za určité časové období. Získáte nějakou formu výstupu řezné vlny jako uvedení aplikace pozastaví, aby se fronta částečně vyprázdnit.

Přejděte na část [“Přidání SCM do počáteční struktury”](#) na stránce 197.

Související pojmy

[“Použití paměti paměťové třídy se sdílenými frontami”](#) na stránce 182

Použití paměti paměťové třídy (SCM) může být výhodné při použití se sdílenými frontami IBM MQ for z/OS .

 [Přidání SCM do počáteční struktury](#)

Jak přidáte SCM pro zlepšení výkonu na systému IBM MQ.

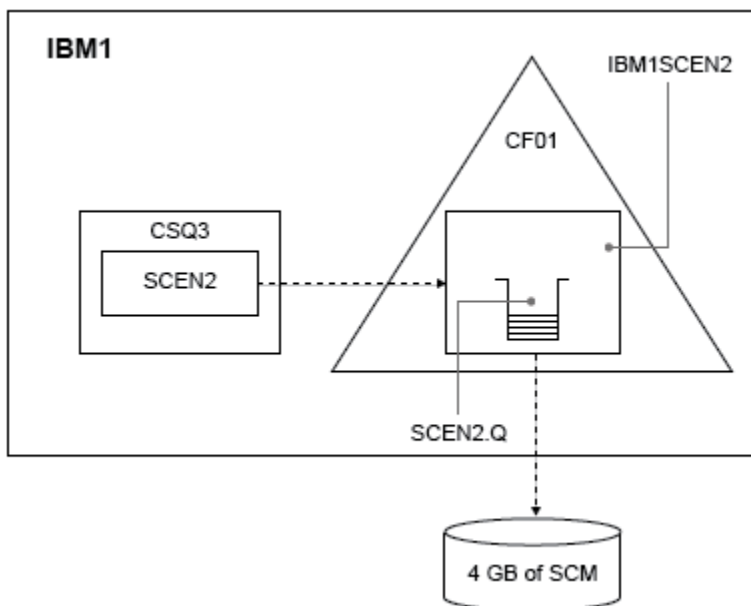
Informace o této úloze

Důležité: IBM z16 je plánováno jako poslední generace produktu IBM Z[®] pro podporu použití virtuální paměti Flash (známé také jako paměť paměťové třídy nebo SCM) pro obrazy prostředku Coupling Facility. Další informace viz: [IBM Z a IBM LinuxONE 4Q 2023 Příkazy pro směr](#).

Jako alternativu byste měli buď použít větší strukturu, nebo odlehčovat zprávy SMDS.

Tato část úlohy používá základní konfiguraci popsanou v části [“Vylepšený výkon-základní konfigurace”](#) na stránce 194. Scénář popisuje přidání SCM do počáteční struktury.

Tato konečná konfigurace je znázorněna v souboru [Obrázek 64](#) na stránce 198.



Obrázek 64. Konfigurace přidání SCM pro zlepšení výkonu

Postup

1. Přidejte 4 GB SCM do struktury IBM1SCEN2 provedením následujícího postupu:

a) Zkontrolujte, kolik SCM je nainstalováno a přiděleno pro CF01, zadáním následujícího příkazu:

```
D CF,CFNAME=CF01
```

b) Zkontrolujte obrázky STORAGE -CLASS MEMORY v sekci STORAGE CONFIGURATION zobrazeného výstupu, abyste viděli dostupné úložiště.

c) Aktualizujte zásadu CFRM pomocí klíčových slov SCMMAXSIZE a SCMALGORITHM, jak je uvedeno:

```
STRUCTURE
NAME(IBM1SCEN2)
SIZE(2048M)
INITSIZE(2048M)
ALLOWAUTOALT(YES)
FULLTHRESHOLD(85)
PREFLIST(CF01)
ALLOWREALLOCATE(YES)
DUPLEX(DISABLED)
ENFORCEORDER(NO)
SCMMAXSIZE(4G)
SCMALGORITHM(KEYPRIORITY1)
```

2. Aktivujte zásadu CFRM zadáním následujícího příkazu:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME=IBM1SCEN2
```

3. Znovu sestavte strukturu IBM1SCEN2 .

Tuto proceduru musíte provést, protože struktura byla přidělena, když jste provedli předchozí změny.

Chcete-li znovu sestavit strukturu, zadejte následující příkaz:

```
SETXCF START,REBUILD,STRNM=IBM1SCEN2
```

4. Zadáním následujícího příkazu potvrďte novou konfiguraci struktury:

```
D XCF,STR,STRNAME=IBM1SCEN2
```

Zkontrolujte výstup příkazu, jehož část je následující:

SPACE USAGE	IN-USE	TOTAL	%
ENTRIES:	33	342684	0
ELEMENTS:	48	6503697	0
EMCS:	2	575600	0
LOCKS:		1024	

Výsledky

Vypočítejte změnu v použití skutečného úložiště zvýšením řídicího úložiště potřebného pro použití SCM.

- Před přidáním SCM do struktury má struktura tyto součty, jak ukazuje [“Vylepšený výkon-základní konfigurace”](#) na stránce 194:
 - 345,242 záznamů
 - 6,548,467 prvků
 - 780,318 EMCS
- Po přidání SCM do struktury má struktura tyto součty:
 - 342,684 záznamů
 - 6,503,697 prvků
 - 575,600 EMCS

Při použití těchto číselných údajů se po přidání SCM struktura zmenší o:

- 2558 záznamů
- 44,770 prvků
- 204,718 EMCS

Množství úložiště struktury, které se používá ke správě SCM, je následující pro strukturu 2 GB s přidělenými 4 GB SCM:

```
(2558 + 44,770 + 204,718) * 256 = 61.5 MB
```

Všimněte si, že přidání více SCM pravděpodobně dosáhne pouze okrajového snížení velikosti struktury, protože množství řídicího úložiště použitého ke sledování SCM se zvyšuje, a to jak velikost struktury, tak množství přiděleného SCM se zvyšuje.

Jak pokračovat dále

Opakujte testy popsané v poslední části části [“Vylepšený výkon-základní konfigurace”](#) na stránce 194.

Výsledky revidované aplikace můžete zakreslit za určité časové období. Porovnáním pozemku s dříve získaným výstupem nyní získáte výstup bez pilové vlny, protože aplikace uvedení již nemusí čekat na částečné vyprázdnění fronty.

Další informace viz [MP16: WebSphere MQ for z/OS -Plánování a vyladění kapacity](#).

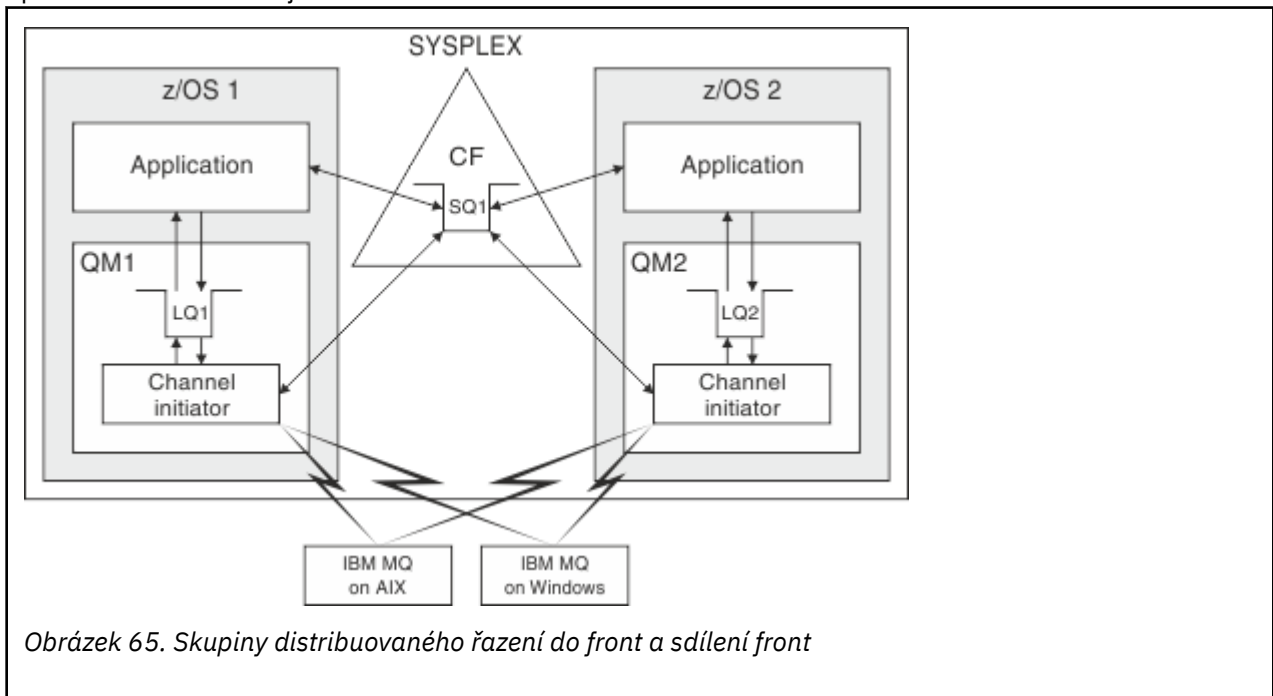
z/OS Skupiny distribuovaného řazení do front a sdílení front

Distribuované řazení do front a skupiny sdílení front jsou dvě techniky, které můžete použít ke zvýšení dostupnosti vašich aplikačních systémů. V tomto tématu naleznete další informace o těchto technikách.

K doplnění vysoké dostupnosti zpráv ve sdílených frontách má komponenta distribuované fronty produktu IBM MQ další funkce, které poskytují následující:

- Vyšší dostupnost pro síť.
- Zvýšená kapacita pro příchozí síťová připojení ke skupině sdílení front.

Obrázek 65 na stránce 200 ilustruje distribuované fronty a skupiny sdílení front. Zobrazuje dva správce front v prostředí sysplex, kteří patří do stejné skupiny sdílení front. Oba mohou přistupovat ke sdílené frontě SQ1. Správci front v síti (například v systémech AIX a Windows) mohou do této fronty vkládat zprávy prostřednictvím inicializátoru kanálu některého ze správců front. Naklonované aplikace v obou správcích front obsluhují frontu.



Obrázek 65. Skupiny distribuovaného řazení do front a sdílení front

Související pojmy

“Sdílené kanály” na stránce 200

Prostřednictvím tohoto tématu můžete porozumět konceptům sdílených kanálů a jejich použití s produktem IBM MQ for z/OS.

“Použití front v rámci skupiny” na stránce 205

Tento oddíl popisuje řazení do front v rámci skupiny, což je funkce IBM MQ for z/OS jedinečná pro platformu z/OS. Tato funkce je k dispozici pouze pro správce front definované pro skupinu sdílení front.

“Klastry a skupiny sdílení front” na stránce 202

V tomto tématu jsou uvedeny informace o způsobu použití skupin sdílení front s klastry.

z/OS Sdílené kanály

Prostřednictvím tohoto tématu můžete porozumět konceptům sdílených kanálů a jejich použití s produktem IBM MQ for z/OS.

Řada síťových produktů poskytuje mechanismus pro skrytí selhání serveru ze sítě nebo pro vyvážení příchozích síťových požadavků v rámci sady vhodných serverů. Síťové produkty zpřístupňují *generický port* pro příchozí požadavky na síťové připojení a příchozí požadavek lze uspokojit připojením k jednomu z vhodných serverů.

Tyto síťové produkty zahrnují:

- Generické prostředky VTAM
- Distributor SYSPLEX

Inicializátor kanálu využívá výhod těchto produktů k využití schopností sdílených front.

Existují dva typy sdílených kanálů: *sdílený příchozí kanál* a *sdílený odchozí kanál*.

- [Sdílené příchozí kanály](#)
- [Sdílené odchozí kanály](#)

Další informace o kanálech viz

- [Souhrn sdíleného kanálu](#)
- [Stav sdíleného kanálu](#)

Sdílené příchozí kanály

Každý inicializátor kanálu ve skupině sdílení front spustí další úlohu modulu listener pro naslouchání na *generickém portu*. Tento generický port je zpřístupněn síti jednou z podpůrných technologií (VTAM, TCP/IP). Příchozí síťové požadavky na připojení ke generickému portu jsou odbaveny síťovou technologií pro všechny listenery ve skupině sdílení front (QSG), které naslouchají na generickém portu.

Můžete spustit kanál na inicializátoru kanálu, ke kterému je příchozí připojení směřováno, pokud má inicializátor kanálu přístup k definici kanálu pro kanál s tímto názvem. Definici kanálu můžete definovat jako soukromou pro správce front nebo uloženou ve sdíleném úložišti, a tedy dostupnou kdekoli (globální definice). To znamená, že definici kanálu můžete zpřístupnit na libovolném inicializátoru kanálu ve skupině sdílení front tím, že ji definujete jako globální definici.

Při spouštění kanálu prostřednictvím generického portu existuje další rozdíl. Synchronizace kanálu se provádí se skupinou sdílení front, nikoli s individuálním správcem front. Zvažte například spuštění kanálu prostřednictvím generického portu vzdáleným správcem front. Při prvním spuštění může být kanál spuštěn ve správci front QM1 a v toku zpráv. Pokud se kanál zastaví a restartuje ve správci front QM2, informace o počtu zpráv, které protékaly, jsou stále správné, protože synchronizace probíhá se skupinou sdílení front.

Příchozí kanál spuštěný prostřednictvím generického portu můžete použít k vložení zpráv do libovolné fronty. Vzdálený správce front neví, zda je cílová fronta sdílená či nikoli. Je-li cílovou frontou sdílená fronta, připojí se vzdálený správce front prostřednictvím libovolného inicializátoru kanálu k dispozici s vyrovnáváním zátěže a zprávy budou vloženy do sdílené fronty.

Pokud je cílovou frontou soukromá fronta, jsou zprávy vkládány do soukromé fronty vlastněné správcem front, ke kterému je připojena aktuální instance kanálu. V tomto prostředí, známém jako *replikované lokální fronty*, musí mít každý správce front definovanou stejnou sadu soukromých front.

Konfigurace kanálů SVRCONN pro skupinu sdílení front

Optimální konfigurací pro kanály SVRCONN ve skupině sdílení front je nastavení soukromých modulů listener v každém parametru CHINIT, které používají jiné číslo portu z jednotlivých kanálů. Tyto porty modulu listener se pak použijí jako prostředky 'back-endového' pro nový mechanismus distribuce pracovní zátěže, jako je například Sysplex Distributor používající virtuální adresy IP (VIPA). Externí adresa VIPA se pak použije jako cílová adresa pro definice CLNTCONN v síti. Kanál SVRCONN lze definovat pomocí QSGDISP (GROUP), takže stejná definice je k dispozici všem správcům front v rámci skupiny QSG. Tato konfigurace se vyhýbá použití sdíleného modulu listener, a proto snižuje vliv výkonu skupiny sdílení front při udržování stavu sdíleného kanálu, který není pro kanály typu klient/server potřebný.

Sdílené odchozí kanály

Odchozí kanál je považován za sdílený kanál, pokud přebírá zprávy ze sdílené přenosové fronty. Pokud je sdílený, uchovává informace o synchronizaci na úrovni skupiny sdílení front. To znamená, že kanál lze

restartovat v jiném správci front a instanci inicializátoru kanálu v rámci skupiny sdílení front v případě, že dojde k selhání subsystému komunikací, inicializátoru kanálu nebo správce front. Restartování nezdařených kanálů tímto způsobem je funkcí sdílených kanálů nazývaných *obnova rovnocenných kanálů*.

Vyrovňávání pracovní zátěže pro sdílené odchozí kanály

Odchozí sdílený kanál je vhodný pro spuštění na libovolném inicializátoru kanálu v rámci skupiny sdílení front, pokud jste neurčili, že chcete, aby byl spuštěn na konkrétním inicializátoru kanálu. Inicializátor kanálu vybraný produktem IBM MQ je určen pomocí následujících kritérií:

- Je komunikační subsystém aktuálně dostupný pro inicializátor kanálu?
- Je připojení Db2 k dispozici pro inicializátor kanálu?
- Který inicializátor kanálu má nejnižší aktuální pracovní zátěž? Pracovní zátěž zahrnuje kanály, které jsou aktivní a opakují se.

Souhrn sdíleného kanálu

Sdílené kanály se liší od soukromých kanálů následujícími způsoby:

Soukromý kanál

Svázáno s jedním inicializátorem kanálu.

- Odchozí kanál používá lokální přenosovou frontu.
- Příchozí kanál byl spuštěn prostřednictvím lokálního portu.
- Informace o synchronizaci uchováváné v SYSTEM.CHANNEL.SYNCQ .

Sdílený kanál

Pracovní zátěž je vyvážena vysokou dostupností.

- Odchozí kanál používá sdílenou přenosovou frontu.
- Příchozí kanál byl spuštěn prostřednictvím generického portu.
- Informace o synchronizaci uchováváné v SYSTEM.QSG.CHANNEL.SYNCQ .

Pomocí voleb CHLDISP s příkazem START CHANNEL určíte, zda je kanál soukromý nebo sdílený. Sdílený kanál lze spustit spuštěním stejným způsobem jako soukromý kanál. Při spuštění sdíleného kanálu však produkt IBM MQ provádí vyrovňávání pracovní zátěže a spouští kanál v nevhodnějším inicializátoru kanálu v rámci skupiny sdílení front. (V případě potřeby můžete určit, že má být na konkrétním inicializátoru kanálu spuštěn sdílený kanál.)

Stav sdíleného kanálu

Iniciátory kanálu ve skupině sdílení front udržují tabulku sdíleného stavu kanálu v produktu Db2. Tento záznam zaznamenává, které kanály jsou aktivní na kterých inicializátorech kanálu. Tabulka stavu sdíleného kanálu se používá v případě selhání inicializátoru kanálu nebo komunikačního systému. Označuje, které kanály je třeba restartovat na jiném inicializátoru kanálu ve skupině sdílení front.

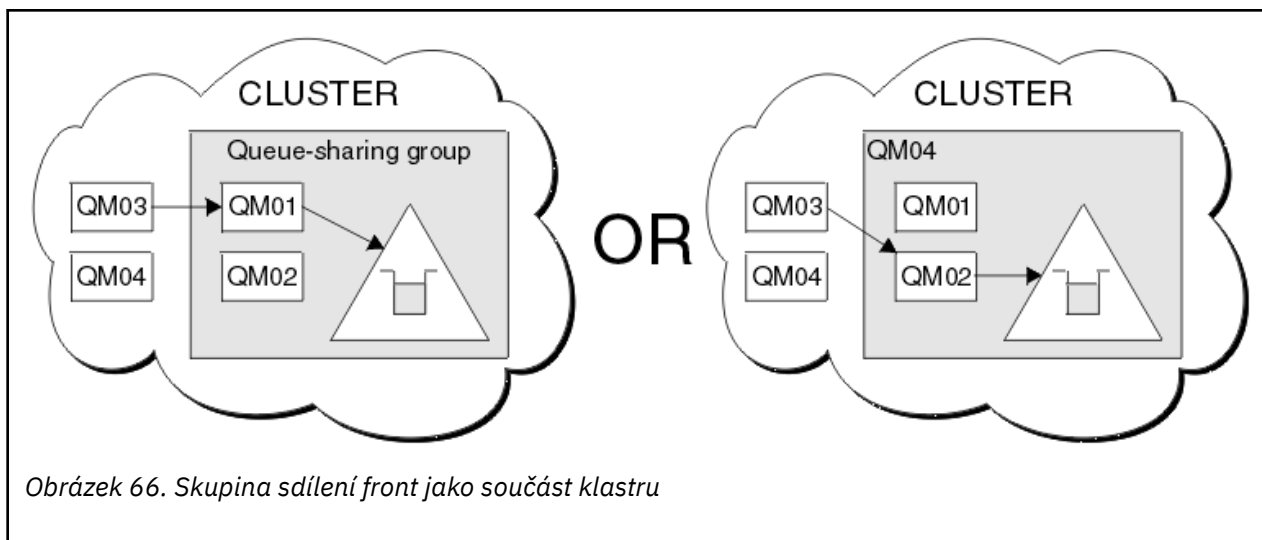
Klastry a skupiny sdílení front

V tomto tématu jsou uvedeny informace o způsobu použití skupin sdílení front s klastry.

Sdílené fronty můžete zpřístupnit klastru v jediné definici. Chcete-li tak učinit, zadejte při definování sdílené fronty název klastru.

Uživatelé v síti vidí sdílenou frontu jako hostovanou jednotlivými správci front v rámci skupiny sdílení front (sdílená fronta není propagována jako fronta hostovaná skupinou sdílení front). Klienti mohou spouštět relace s libovolnými členy skupiny sdílení front a vkládat zprávy do stejné sdílené fronty.

Obrázek 66 na stránce 203 ukazuje, jak členové klastru mohou přistupovat ke sdílené frontě prostřednictvím libovolného člena skupiny sdílení front.



► z/OS Ovlivňování distribuce pracovní zátěže pomocí sdílených front

Prostřednictvím tohoto tématu můžete porozumět faktorům, které ovlivňují distribuci pracovní zátěže se sdílenými frontami ve skupině sdílení front.

Produkt IBM MQ neposkytuje vyrovnávání pracovní zátěže pro sdílené fronty. Distribuce pracovní zátěže ve skupině sdílení front (QSG) však může být ovlivněna *způsobem založeným na stahování*. Volba, které správce front obsluhuje frontu (přijímá zprávu zapsanou do sdílené fronty), je ovlivněna dostupnou kapacitou zpracování jednotlivých správců front ve skupině sdílení front a cíli správy pracovní zátěže definovanými v prostředí sysplex.

Je však důležité ocenit, že správce front, který provádí operaci MQPUT zprávy, může mít také velký vliv na rozhodování o tom, který správce front zprávu získá.

Lokální správce front pravděpodobně provede operaci MQGET.

Pro aplikaci provádějící příkaz MQPUT se lokální správce front označuje jako správce front, ke kterému je aplikace připojena.

Právě služby správce front, které provádějí operaci MQPUT zprávy provedením operace MQGET jménem aplikace získávání, jsou ovlivněny následujícími aspekty.

Je-li zpráva vložena do prázdné sdílené fronty, je lokální správce front obvykle vystaven před oznámením všem ostatním správcům front ve skupině sdílení front. Je-li lokální správce front schopen zpracovat zprávu, obdrží oznámení o přechodu seznamu od prostředku Coupling Facility (CF) před jakýmkoli jiným správcem front v rámci skupiny sdílení front. (Oznámení o přechodu seznamu je oznámení, že sdílená fronta změnila stav z prázdného na neprázdný.)

Možné scénáře jsou v tomto případě následující:

1. MQPUT přechodné zprávy mimo synchronizační bod a *rychlé vložení do čekající metody getter*.

Pokud v lokálním správcu front pro frontu existuje aplikace s příkazem *MQGET with wait*, je požadavek MQPUT zprávy předán přímo do vyrovnávací paměti aplikace pro načítání a není zapsán do fronty. To platí pro sdílené i nesdílené fronty. Tato funkce se často nazývá *rychlé vložení do čekajícího mechanismu getter*. V případě sdílených front není upozorněn žádný jiný správce front v rámci skupiny sdílení front, protože nedochází k žádnému přechodu z prázdné fronty na neprázdnou frontu. To například znamená, že za předpokladu, že tento správce front může obsluhovat všechna vložení z této aplikace, a za předpokladu, že do fronty nevkládají zprávy žádné jiné aplikace, pak žádný jiný správce front ve skupině sdílení front při vyprazdňování této fronty nepomáhá. Pokud však neexistuje příkaz MQGET s čekáním na lokálního správce front a do sdílené fronty je vložena zpráva, prostředek CF oznámí ostatním správcům front ve skupině sdílení front podle svých pravidel pro oznamování přechodů seznamu.

2. MQPUT trvalé zprávy nebo zprávy v synchronizačním bodu.

Pokud v tomto případě existuje aplikace s příkazem *MQGET* s čekáním v lokálním správci front, bude zpráva vložena do sdílené fronty a prostředek CF oznámí ostatním správcům front ve skupině sdílení front podle svých pravidel pro oznámení o přechodech seznamu. Lokální správce front však nečeká na oznámení o přechodu z prostředku CF, ale nejprve dodržuje všechny lokální požadavky *MQGET* s volbou *wait* a obvykle provádí získávání této zprávy jménem aplikace, než bude moci kterýkoli jiný správce front ve skupině sdílení front odpovědět na oznámení prostředku CF. To závisí na tom, jak je lokální správce front zaneprázdněn. Jinak se každý správce front, na kterého byl prostředek CF upozorněn v důsledku přijetí zprávy v prázdné frontě, pokusí nejprve provést zpracování operace *get*. První správce front, který odpověděl, zpracuje novou zprávu.

3. Nakonec, pokud fronta není vyčerpána zprávami a prostředek CF odeslal oznámení o změně stavu z prázdné na neprázdnou pro frontu, všichni připojení správci front budou mít příležitost pomoci při zpracování fronty. V této události se pracovní zátěž označuje jako *založená na stažení*.

Tento návrh umožňuje lepší výkon nad čistě tažnou distribucí pracovní zátěže. Cílem je využít vysokou dostupnost, kterou nabízejí fronty zadržené v prostředku CF, a pokud je to možné, umožnit správci front provést operaci *MQGET*, aniž by bylo nutné odkazovat na prostředek CF, a zpracovat tak pracovní zátěž zpráv co nejefektivněji.

Alternativní přístupy mohou být přijaty tam, kde je důležitější důraz na vyváženost pracovní zátěže než dříve popsaná vylepšení výkonu. Například se ujistěte, že žádná z aplikací získávání není připojena ke stejnému správci front, ke kterému je aplikace vkládání připojena. Při použití tohoto návrhu jsou všechny zprávy vloženy do fronty a všichni správci front v rámci skupiny sdílení front jsou upozorněni, když se fronta přesune z prázdné na neprázdnou, v souladu s algoritmem CF pro zpracování těchto přechodů. Kromě toho mechanismus *fast put to waiting getter* nelze použít.

Kde najít další informace o sdílených frontách a skupinách sdílení front

V tabulce v tomto tématu naleznete další informace o tom, jak produkt IBM MQ for z/OS používá sdílené fronty a skupiny sdílení front.

Téma	Kam se podívat
Obnova skupiny sdílení front	“Obnova a restart na systému z/OS” na stránce 244
Zabezpečení skupiny sdílení front	“Koncepte zabezpečení v produktu IBM MQ for z/OS” na stránce 260
Soukromé a globální definice objektů Směrování příkazů do jiné fronty správci	Zdroje, ze kterých můžete zadávat příkazy v systému z/OS
Plánování prostředku Coupling Facility prostředí	Definování prostředků prostředku Coupling Facility
Plánování prostředí SMDS	Plánování prostředí datové sady sdílených zpráv (SMDS)
Plánování Prostředí produktu Db2	Plánování prostředí Db2
Nastavení sdílených front Parametry systému	“Sdílené fronty a skupiny sdílení front” na stránce 161

Tabulka 19. Kde najít další informace o sdílených frontách a skupinách sdílení front (pokračování)

Téma	Kam se podívat
Obslužné programy Migrate front	IBM MQ v z/OS odkazu
Zprávy konzoly	Zprávy pro IBM MQ for z/OS
Příkazy MQSC	Příkazy MQSC
IBM MQKlastry	Konfigurace klastru správců front
IBM MQ distribuované fronty Názvy kanálů	Úvod do správy distribuovaných front
Psaní aplikací	Přehled návrhu aplikace
Volání MQCONNX	MQCONNX

z/OS

Použití front v rámci skupiny

Tento oddíl popisuje řazení do front v rámci skupiny, což je funkce IBM MQ for z/OS jedinečná pro platformu z/OS. Tato funkce je k dispozici pouze pro správce front definované pro skupinu sdílení front.

Informace o skupinách sdílení front viz [“Sdílené fronty a skupiny sdílení front”](#) na stránce 161.

Koncepce řazení do front v rámci skupiny

Můžete provádět rychlý přenos zpráv mezi správci front ve skupině sdílení front bez definování kanálů. Používá systémovou frontu s názvem SYSTEM.QSG.TRANSMIT.QUEUE, což je sdílená přenosová fronta. Každý správce front ve skupině sdílení front spustí úlohu s názvem agent front v rámci skupiny, který čeká na doručení zpráv do této fronty, které jsou určeny pro příslušného správce front. Je-li taková zpráva zjištěna, je odebrána z fronty a umístěna do správné cílové fronty.

Používají se standardní pravidla pro překlad názvů, ale pokud je povoleno řazení do front v rámci skupiny (IGQ) a cílový správce front je ve skupině sdílení front, systém SYSTEM.QSG.TRANSMIT.QUEUE se používá k přenosu zprávy do správného cílového správce front namísto použití přenosové fronty a kanálu.

Řazení do front v rámci skupiny lze povolit prostřednictvím atributu správce front. Řazení do front v rámci skupiny přesouvá přechodné zprávy mimo synchronizační bod a trvalé zprávy v synchronizačním bodu. Pokud zjistí problém s doručení zpráv do cílové fronty, pokusí se je zařazení do fronty nedoručených zpráv zařadit do fronty nedoručených zpráv. Pokud je fronta nedoručených zpráv plná nebo nedefinovaná, přechodné zprávy jsou vyřazeny, ale trvalé zprávy jsou vráceny zpět a vráceny do SYSTEM.QSG.TRANSMIT.QUEUEa agent IGQ se pokusí doručit zprávy, dokud nebude úspěšný.

Příchozí sdílený kanál, který přijímá zprávu určenou pro frontu v jiném správci front ve skupině sdílení front, může použít řazení do fronty v rámci skupiny pro *přechod* zprávy do správného místa určení.

Může nastat situace, kdy chcete, aby lokální správce front vložil zprávu přímo do cílové fronty, pokud je cílovou frontou sdílená fronta, a nikoli zpráva, která byla nejprve přenesena do cílového správce front. K řízení této situace můžete použít atribut SQQMNAME správce front. Nastavíte-li hodnotu parametru SQQMNAME na USE, bude příkaz MQOPEN proveden ve správci front určeném parametrem **ObjectQMgrName**.

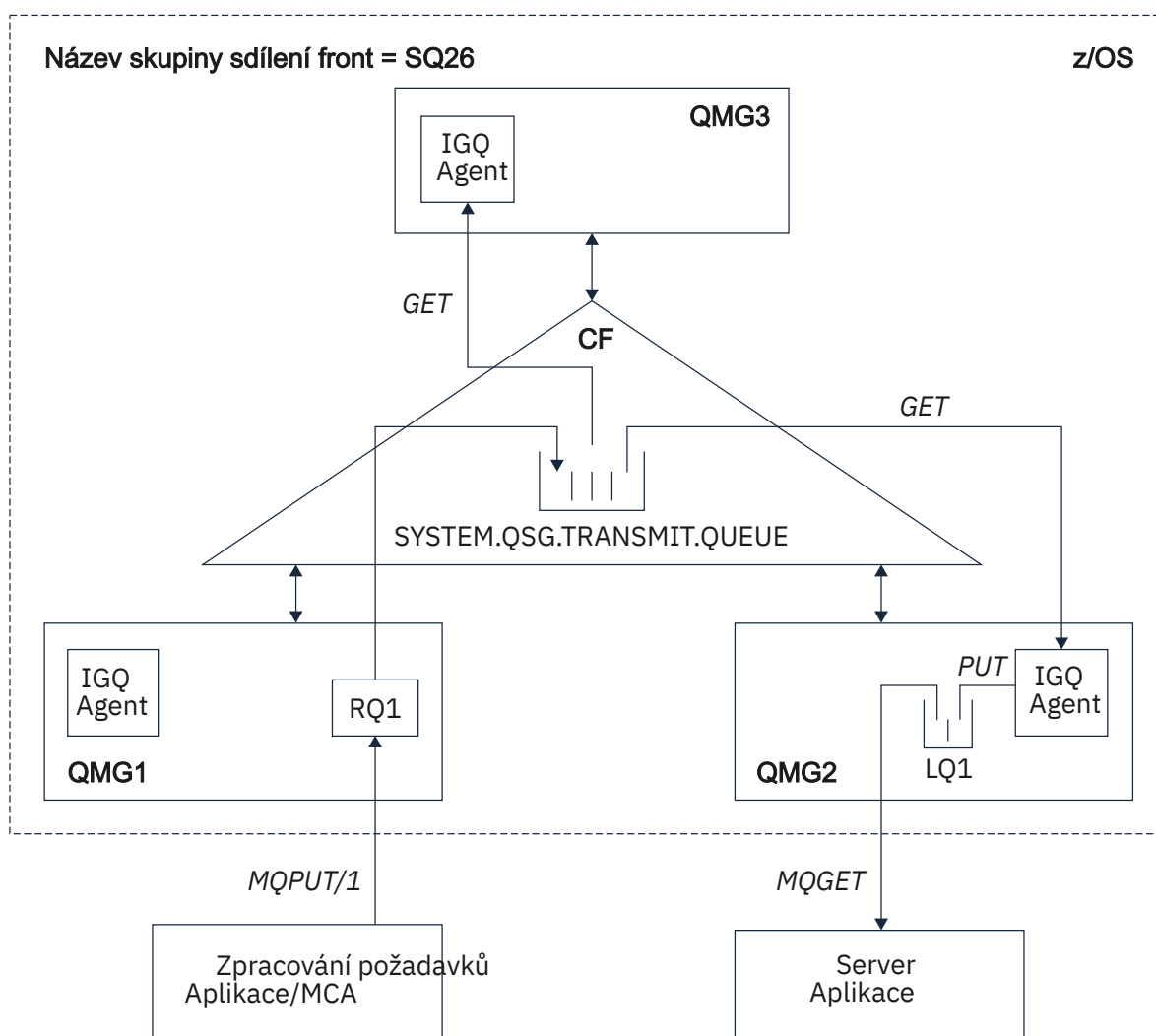
Pokud je však cílovou frontou sdílená fronta a nastavíte hodnotu SQQMNAME na hodnotu IGNORE a proměnná **ObjectQMgrName** je fronta jiného správce front ve skupině sdílení front, bude sdílená fronta

otevřena v lokálním správci front. Pokud lokální správce front nemůže otevřít cílovou frontu nebo vložit zprávu do fronty, bude zpráva přenesena do určeného **ObjectQMGrName** prostřednictvím kanálu IGQ nebo IBM MQ .

Řazení do front v rámci skupiny lze použít k efektivnějšímu doručování malých zpráv do front umístěných ve vzdálených správci front v rámci skupiny sdílení front. Řazení do front v rámci skupiny také podporuje velké zprávy, z nichž největší je 100 MB *minus* délka záhlaví přenosové fronty.

Poznámka: Používáte-li tuto funkci, musí mít uživatelé stejný přístup k frontám v každém správci front ve skupině sdílení front.

Následující diagram zobrazuje typický příklad řazení do front v rámci skupiny.



Obrázek 67. Příklad řazení do front v rámci skupiny

Diagram zobrazuje:

- Agenti IGQ spuštění ve třech správci front (QMG1, QMG2a QMG3), kteří jsou definováni pro skupinu sdílení front s názvem SQ26.
- Sdílená přenosová fronta SYSTEM.QSG.TRANSMIT.QUEUE , která je definována v prostředku Coupling Facility (CF).
- Definice vzdálené fronty, která je definována ve správci front QMG1.
- Lokální fronta, která je definována ve správci front QMG2.

- Požadující aplikace (tato aplikace může být MCA (Message Channel Agent)), která je připojena ke správci front QMG1.
- Serverová aplikace, která je připojena ke správci front QMG2.
- Zpráva požadavku, která se umísťuje do SYSTEM.QSG.TRANSMIT.QUEUE.

Řízení front v rámci skupiny a agent front v rámci skupiny

Agent IGQ je spuštěn během inicializace správce front. Když aplikace otevírají a vkládají zprávy do vzdálených front, lokální správce front určí, zda se pro přenos zpráv používá řazení do front v rámci skupiny. Má-li být použito řazení do front v rámci skupiny, lokální správce front umístí zprávu do systému SYSTEM.QSG.TRANSMIT.QUEUE. Agent IGQ v cílovém vzdáleném správci front načte zprávu a umístí ji do cílové fronty.

Terminologie řazení do front v rámci skupiny

Vysvětlení terminologie: řazení do front v rámci skupiny, sdílená přenosová fronta pro použití ve frontách v rámci skupiny a agent front v rámci skupiny.

Použití front v rámci skupiny

Řazení do front v rámci skupiny může mít vliv na potenciálně rychlý a méně nákladný přenos zpráv mezi správci front ve skupině sdílení front, aniž by bylo nutné definovat kanály.

Sdílená přenosová fronta pro použití ve frontách uvnitř skupiny

Každá skupina sdílení front má sdílenou přenosovou frontu s názvem SYSTEM.QSG.TRANSMIT.QUEUE pro použití ve frontách uvnitř skupiny. Je-li povoleno řazení do front v rámci skupiny, SYSTEM.QSG.TRANSMIT.QUEUE se objeví v cestě k rozlišení názvů při otevírání vzdálených front. Když aplikace (včetně agentů MCA (Message Channel Agent)) vkládají zprávy do vzdálené fronty, lokální správce front určí způsobilost zpráv pro rychlý přenos a umístí je do systému SYSTEM.QSG.TRANSMIT.QUEUE.

Agent řazení do front v rámci skupiny

Agent IGQ je úloha spuštěná při inicializaci správce front, která čeká na doručení vhodných zpráv do systému SYSTEM.QSG.TRANSMIT.QUEUE. Agent IGQ načte vhodné zprávy z této fronty a doručí je do cílových front.

Agent IGQ pro každého správce front je vždy spuštěn, protože řazení do front v rámci skupiny je používáno samotným správcem front pro vlastní interní zpracování.

z/OS Výhody řazení do fronty v rámci skupiny

Výhody řazení do front v rámci skupiny: snížené definice systému, snížená administrace systému, lepší výkon, podpora migrace a doručování zpráv při vícenásobném přeskokování mezi správci front ve skupině sdílení front.

Výhody zařazení do fronty v rámci skupiny jsou:

Snížené definice systému

Řazení do front v rámci skupiny odstraňuje potřebu definovat kanály mezi správci front ve skupině sdílení front.

Snížená administrace systému

Vzhledem k tomu, že mezi správci front ve skupině sdílení front nejsou definovány žádné kanály, není vyžadována administrace kanálů.

Zlepšený výkon

Vzhledem k tomu, že pro doručení zprávy do cílové fronty je zapotřebí pouze jeden agent IGQ (namísto dvou zprostředkujících agentů odesilatele a příjemce), může být doručení zpráv s použitím

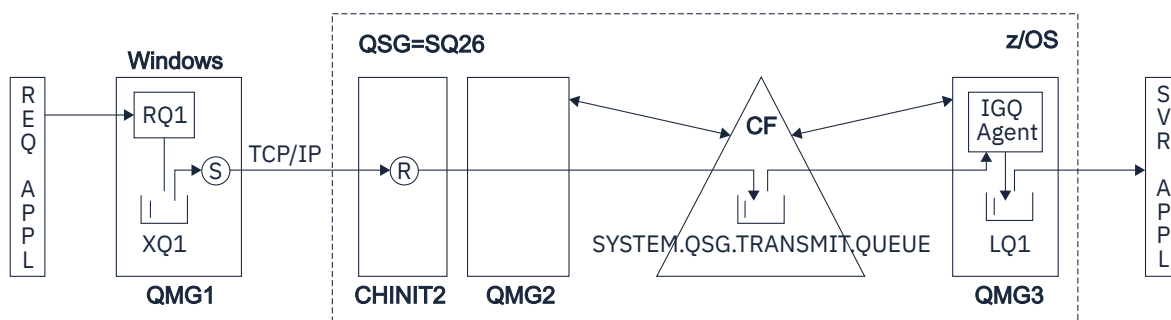
front v rámci skupiny méně nákladné než doručení zpráv s použitím kanálů. V rámci skupinového řazení do fronty existuje pouze přijímající komponenta, protože byla odstraněna potřeba odesílající komponenty. Toto uložení je způsobeno tím, že zpráva je pro agenta IGQ v cílovém správci front k dispozici pro doručení do cílové fronty po dokončení operace vložení v lokálním správci front a v případě zpráv vložených do oboru synchronizačního bodu je potvrzena.

Podporuje migraci

Externí aplikace pro skupinu sdílení front mohou doručovat zprávy do fronty umístěné v libovolném správci front ve skupině sdílení front, zatímco jsou připojeny pouze ke konkrétnímu správci front ve skupině sdílení front. Důvodem je skutečnost, že zprávy přicházející do přijímacího kanálu, určené pro frontu ve vzdáleném správci front, mohou být transparentně odesílány do cílové fronty s použitím front v rámci skupiny. Tento prostředek umožňuje implementaci aplikací mezi skupinou sdílení front bez nutnosti měnit systémy, které jsou vůči skupině sdílení front externí.

Typickou konfiguraci ilustruje následující diagram, ve kterém:

- Požadující aplikace připojená ke správci front QMG1 musí odeslat zprávu do lokální fronty ve správci front QMG3.
- Správce front QMG1 je připojen pouze ke správci front QMG2.
- Správci front QMG2 a QMG3, kteří byli dříve připojeni prostřednictvím kanálů, jsou nyní členy skupiny sdílení front SQ26.



Obrázek 68. Příklad podpory migrace

Tok operací je následující:

1. Požadující aplikace vloží zprávu určenou pro lokální frontu LQ1 ve vzdáleném správci front QMG3 do definice vzdálené fronty RQ1.
2. Správce front QMG1 spuštěný na pracovní stanici Windows NT umístí zprávu do přenosové fronty XQ1.
3. Odesílatel MCA (S) na systému QM1 odešle zprávu pomocí protokolu TCP/IP příjemci MCA (R) na inicializátoru kanálu CHINIT2.
4. Příjemce MCA (R) na inicializátoru kanálu CHINIT2 umístí zprávu do sdílené přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE.
5. Agent IGQ ve správci front QMG3 načte zprávu ze systému SYSTEM.QSG.TRANSMIT.QUEUE a umístí ji do cílové lokální fronty LQ1.
6. Aplikace serveru načte zprávu z cílové lokální fronty a zpracuje ji.

Doručování zpráv při vícenásobném přeskakování mezi správci front ve skupině sdílení front

Předchozí diagram v části Podporuje migraci také ilustruje doručení zpráv při vícenásobném přeskakování mezi správci front ve skupině sdílení front. Zprávy přicházející ve správci front v rámci skupiny sdílení front, ale určené pro frontu v jiném správci front ve skupině sdílení front, lze snadno přenést do cílové fronty ve správci cílových front pomocí front v rámci skupiny.

Omezení front v rámci skupiny

Omezení front v rámci skupiny jsou: zprávy vhodné pro přenos s použitím front v rámci skupiny, počet agentů front v rámci skupiny na jednoho správce front a spuštění a zastavení agenta front v rámci skupiny.

Toto téma popisuje omezení řazení do front v rámci skupiny.

Zprávy vhodné pro přenos s použitím front v rámci skupiny

Vzhledem k tomu, že řazení do front v rámci skupiny používá sdílenou přenosovou frontu, která je definována v prostředku CF (coupling facility), je řazení do front v rámci skupiny omezeno na doručování zpráv s maximální podporovanou délkou zpráv pro sdílené fronty minus délka záhlaví přenosové fronty (MQXQH).

Počet agentů řazení do front v rámci skupiny na jednoho správce front

Pro každého správce front ve skupině sdílení front je spuštěn pouze jeden agent IGQ.

Spuštění a zastavení agenta řazení do front v rámci skupiny

Agent IGQ je spuštěn během inicializace správce front a ukončen během vypnutí správce front. Je navržen tak, aby byl dlouhotrvající, sám zotavuje (v případě abnormálního ukončení), úkol. Pokud dojde k chybě s definicí SYSTEM.QSG.TRANSMIT.QUEUE (například pokud je tato fronta blokována Get), agent IGQ pokračuje v opakování. Pokud agent IGQ zjistí chybu, která má za následek normální ukončení agenta v době, kdy je správce front stále aktivní, lze jej restartovat zadáním příkazu ALTER QMGR IGQ (ENABLED). Tento příkaz se vyhýbá nutnosti recyklovat správce front.

Nastavení atributu správce front IGQ na hodnotu ENABLED nebo DISABLED

Je-li atribut správce front IGQ nastaven na hodnotu ENABLED nebo DISABLED, stávající manipulátory objektů mohou být zneplatněny s kódem příčiny MQRC_OBJECT_CHANGED. Další informace naleznete v tématu [Začínáme s řazením do front uvnitř skupiny](#).

Začínáme s řazením do front v rámci skupiny

Můžete povolit, zakázat a používat řazení do front v rámci skupiny, jak je popsáno v tomto tématu.

Povolení řazení do front v rámci skupiny

Chcete-li ve svých správcích front povolit řazení do front v rámci skupiny, je třeba provést následující akce:

- Definujte sdílenou přenosovou frontu s názvem SYSTEM.QSG.TRANSMIT.QUEUE. Definici této fronty lze nalézt v souboru thlqual.SCSQPROC(CSQ4INSS), ukázce CSQINP2 pro objekty SYSTEM pro skupiny sdílení front. Tato fronta musí být definována se správnými atributy, jak je uvedeno v souboru thlqual.SCSQPROC(CSQ4INSS), aby řazení do fronty v rámci skupiny fungovalo správně.
- Vzhledem k tomu, že agent IGQ je vždy spuštěn při inicializaci správce front, je pro zpracování příchozích zpráv vždy k dispozici řazení do fronty v rámci skupiny. Agent IGQ zpracovává všechny zprávy, které jsou umístěny v systému SYSTEM.QSG.TRANSMIT.QUEUE. Chcete-li však povolit řazení do front v rámci skupiny pro zpracování odchozích požadavků, musí být atribut IGQ správce front nastaven na hodnotu ENABLED.

Důležité: Je-li atribut správce front IGQ nastaven na hodnotu ENABLED, stávající manipulátory objektů mohou být zneplatněny s kódem příčiny MQRC_OBJECT_CHANGED. Další informace viz [“Specifické vlastnosti řazení do front v rámci skupiny”](#) na stránce 217. Jak je popsáno v části 'Odpověď programátora' pro tento kód příčiny, aplikace musí být kódovány, aby tuto situaci zvládly (další podrobnosti viz [2041 \(07F9\) \(RC2041\): MQRC_OBJECT_CHANGED](#)).

Vzhledem k tomu, že IGQ je navržen jako úloha s dlouhou dobou zpracování a samočinnou obnovou, která se spouští během inicializace a ukončuje se ukončením, další informace viz [“Omezení front v rámci skupiny”](#) na stránce 209.

Zakázání řazení do front v rámci skupiny

Chcete-li zakázat řazení do front v rámci skupiny pro přenos odchozích zpráv, nastavte atribut správce front IGQ na hodnotu DISABLED. Je-li pro konkrétního správce front zakázáno řazení do front v rámci skupiny, může agent IGQ v daném správci front i nadále zpracovávat příchozí zprávy umístěné

v systému SYSTEM.QSG.TRANSMIT.QUEUE od správce front, který má pro odchozí přenos povoleno řazení do front v rámci skupiny.

Důležité: Je-li atribut správce front IGQ nastaven na hodnotu ENABLED, stávající manipulátory objektů mohou být zneplatněny s kódem příčiny MQRC_OBJECT_CHANGED. Další informace viz [“Specifické vlastnosti řazení do front v rámci skupiny”](#) na stránce 217. Jak je popsáno v části 'Odpověď programátora' pro tento kód příčiny, aplikace musí být kódovány, aby tuto situaci zvládly (další podrobnosti viz 2041 (07F9) (RC2041): MQRC_OBJECT_CHANGED).

Vzhledem k tomu, že IGQ je navržen jako úloha s dlouhou dobou zpracování a samočinnou obnovou, která se spouští během inicializace a ukončuje se ukončením, další informace viz [“Omezení front v rámci skupiny”](#) na stránce 209 .

Použití řazení do front v rámci skupiny

Jakmile je povoleno řazení do front v rámci skupiny, je k dispozici pro použití a správce front jej používá, kdykoli je to možné. To znamená, že když aplikace vloží zprávu do definice vzdálené fronty, do plně kvalifikované vzdálené fronty nebo do fronty klastru, správce front určí, zda je zpráva vhodná k doručení pomocí front v rámci skupiny, a pokud ano, umístí zprávu do systému SYSTEM.QSG.TRANSMIT.QUEUE. Není třeba měnit uživatelské aplikace ani fronty aplikací, protože pro vhodné zprávy používá správce front systém SYSTEM.QSG.TRANSMIT.QUEUE, přednostně před jinou přenosovou frontou.

Konfigurace řazení do front v rámci skupiny

Kromě typické konfigurace řazení do front v rámci skupiny jsou možné i jiné konfigurace.

[“Koncepce řazení do front v rámci skupiny”](#) na stránce 205 popisuje typickou konfiguraci.

Související pojmy

[“Distribuované řazení do front s řazením do front v rámci skupiny \(více cest doručení\)”](#) na stránce 210
U aplikací, které zpracovávají krátké zprávy, může být možné konfigurovat řazení do front v rámci skupiny pouze pro doručování zpráv mezi správci front ve skupině sdílení front.

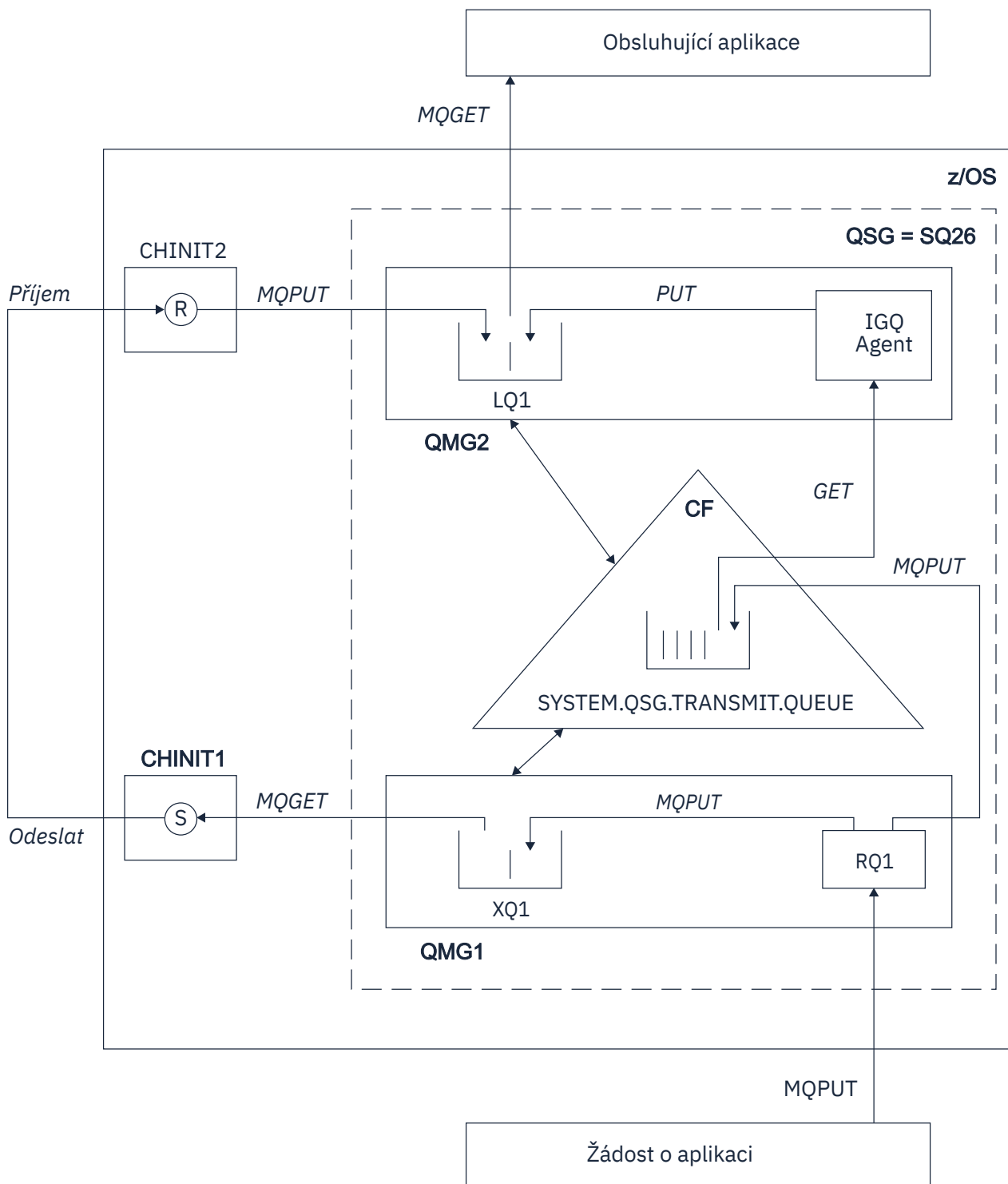
[“Klastrování s řazením do front v rámci skupiny \(více cest doručení\)”](#) na stránce 212
Je možné konfigurovat správce front tak, aby byli v klastru i ve skupině sdílení front.

[“Klastrování, řazení do front v rámci skupiny a distribuované řazení do front”](#) na stránce 214
Je možné konfigurovat správce front, který je členem klastru, stejně jako skupiny sdílení front a je připojen k distribuovanému správci front pomocí dvojice odesílací/přijímací kanál.

Distribuované řazení do front s řazením do front v rámci skupiny (více cest doručení)

U aplikací, které zpracovávají krátké zprávy, může být možné konfigurovat řazení do front v rámci skupiny pouze pro doručování zpráv mezi správci front ve skupině sdílení front.

Volba zařazení do fronty v rámci skupiny prostřednictvím komunikace kanálu může být řízena úrovní typu CFSTRUCT. (3 místo 4 nebo 5). Maximální délka zprávy nastavená na SYSTEM.QSQ.TRANSMIT.QUEUE.



Obrázek 69. Příklad konfigurace

Zpracování otevření/vložení

1. Je důležité si uvědomit, že když požadující aplikace otevře vzdálenou frontu RQ1, dojde k rozlišení názvu pro nesdílenou přenosovou frontu XQ1 a sdílenou přenosovou frontu SYSTEM. QSG. TRANSMIT. QUEUE.
2. Pokud požadující aplikace vloží zprávu do vzdálené fronty na základě toho, zda je povoleno použití front v rámci skupiny pro odchozí přenos ve správci front a na základě charakteristik zpráv, je zpráva vložena do přenosové fronty XQ1 nebo do přenosové fronty SYSTEM. QSG. TRANSMIT. QUEUE. Správce

front umístí všechny velké zprávy do přenosové fronty XQ1a všechny malé zprávy do přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE.

3. Pokud je přenosová fronta XQ1 plná nebo není k dispozici, požadavky na vložení velkých zpráv se nezdaří synchronně s vhodným návratovým kódem a kódem příčiny. Požadavky na vložení malých zpráv jsou však i nadále úspěšné a jsou umístěny do přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE.
4. Je-li přenosová fronta SYSTEM.QSG.TRANSMIT.QUEUE je plná nebo do ní nelze vložit požadavky na vložení malých zpráv, které selhávají synchronně s vhodným návratovým kódem a kódem příčiny. Požadavky na vložení velkých zpráv jsou však i nadále úspěšné a jsou umístěny do přenosové fronty XQ1. V tomto případě se neprovede žádný pokus o vložení malých zpráv do přenosové fronty.

Tok pro velké zprávy

1. Požadující aplikace vkládá velké zprávy do vzdálené fronty RQ1.
2. Správce front QMG1 vloží zprávy do přenosové fronty XQ1.
3. Odesílatel MCA (S) ve správci front QMG1 načte zprávy z přenosové fronty XQ1 a odešle je správci front QMG2.
4. Příjemce MCA (R) ve správci front QMG2 přijme zprávy a umístí je do cílové fronty LQ1.
5. Obsluhující aplikace načte a poté zpracuje zprávy z fronty LQ1.

Tok pro malé zprávy

1. Požadující aplikace vloží malé zprávy do vzdálené fronty RQ1.
2. Správce front QMG1 vloží zprávy do přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE.
3. Funkce IGQ ve správci front QMG2 načte zprávy a umístí je do cílové fronty LQ1.
4. Aplikace obsluhy načte zprávy z fronty LQ1.

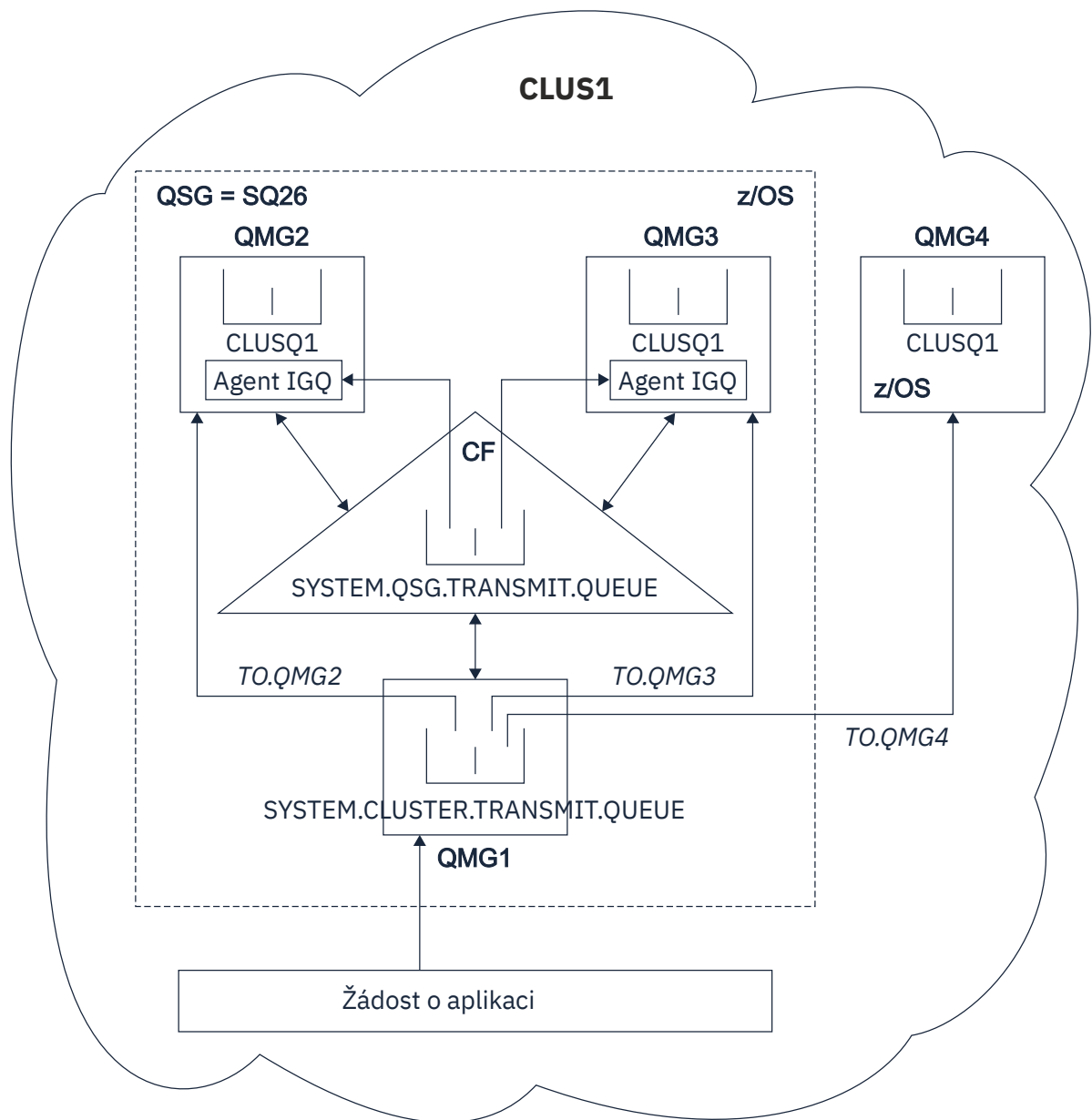
Body k poznámce

1. Žadající aplikace nemusí být informována o základním mechanismu používaném pro doručování zpráv.
2. Pro malé zprávy lze dosáhnout potenciálně rychlejšího mechanismu doručování zpráv.
3. Pro doručování zpráv je k dispozici více cest (tj. normální trasa kanálu a trasa řazení do front v rámci skupiny).
4. Trasa řazení do fronty v rámci skupiny, která je potenciálně rychlejší, je vybrána přednostně před normální kanálovou trasou. V závislosti na charakteristice zprávy může být doručení zprávy rozděleno mezi dvě cesty. Proto mohou být zprávy doručeny mimo pořadí (i když je toto doručení také možné, pokud jsou zprávy doručovány pouze pomocí normální trasy kanálu).
5. Po výběru přenosové cesty a umístění zpráv do přenosových front se pro doručení zpráv použije pouze vybraná přenosová cesta. Všechny nezpracované zprávy v systému SYSTEM.QSG.TRANSMIT.QUEUE nejsou přesměrovány do přenosové fronty XQ1.

z/OS Klastrování s řazením do front v rámci skupiny (více cest doručení)

Je možné konfigurovat správce front tak, aby byli v klastru i ve skupině sdílení front.

Jsou-li zprávy odesílány do fronty klastru a lokální a vzdálení správci cílových front jsou ve stejné skupině sdílení front, je pro doručování malých zpráv (prostřednictvím systému SYSTEM.QSG.TRANSMIT.QUEUE) a doručování velkých zpráv, pokud řazení do front v rámci skupiny podporuje velikost zprávy. Také systém SYSTEM.CLUSTER.TRANSMIT.QUEUE se používá pro doručení zpráv libovolnému správci front, který je v klastru, ale mimo skupinu sdílení front. Následující diagram ilustruje tuto konfiguraci (iniciátory kanálu nejsou zobrazeny).



Obrázek 70. Příklad klastrování s řazením do front v rámci skupiny

Diagram zobrazuje:

- Čtyři z/OS správci front QMG1, QMG2, QMG3 a QMG4 konfigurované v klastru CLUS1.
- Správci front QMG1, QMG2 a QMG3 konfigurovaní ve skupině sdílení front SQ26.
- Agenti IGQ spuštění ve správčích front QMG2 a QMG3.
- Lokální SYSTEM.CLUSTER.TRANSMIT.QUEUE definovaná v QMG1.

Poznámka: Pro přehlednost se jedná o systém SYSTEM.CLUSTER.TRANSMIT.QUEUE v ostatních správčích front, které nejsou zobrazeny.

- Sdílený systém SYSTEM.QSG.TRANSMIT.QUEUE definovaná v prostředí CF, který je ve struktuře IBM MQ nakonfigurované s atributem CFLEVEL (3) RECOVER (YES).
- Kanály klastru TO.QMG2 (připojení QMG1 k QMG2), TO.QMG3 (připojení QMG1 k QMG3) a TO.QMG4 (připojení QMG1 k QMG4).
- Fronta klastru CLUSQ1 je hostována na správčích front QMG2, QMG3 a QMG4.

Předpokládejme, že požadující aplikace otevře frontu klastru s volbou MQOO_BIND_NOT_FIXED, aby byl cílový správce front pro frontu klastru vybrán v době vložení.

Pokud je vybrán cílový správce front QMG2:

- Všechny velké zprávy vkládané žádající aplikací jsou:
 - Vložíme do SYSTEM.CLUSTER.TRANSMIT.QUEUE na QMG1, protože SYSTEM.QSG.TRANSMIT.QUEUE je ve struktuře CFLEVEL (3); proto podporuje zprávy pouze do velikosti 63 KB.
 - Přeneseno do fronty klastru CLUSQ1 na QMG2 pomocí kanálu klastru TO.QMG2
- Všechny malé zprávy vkládané žádající aplikací jsou
 - Vložte do sdílené přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE. Tato fronta je ve struktuře nakonfigurované s atributem RECOVER (YES), takže se používá pro trvalé i dočasné malé zprávy.
 - Načteno agentem IGQ na QMG2
 - Vložit do fronty klastru CLUSQ1 na QMG2

Je-li vybrán cílový správce front QMG4:

- Protože QMG4 není členem skupiny sdílení front SQ26, všechny zprávy vkládané požadující aplikací jsou
 - Vložíme do SYSTEM.CLUSTER.TRANSMIT.QUEUE na QMG1
 - Přeneseno do fronty klastru CLUSQ1 na QMG4 pomocí kanálu klastru TO.QMG4

Body k poznámce

- Žádající aplikace nemusí být informována o základním mechanismu používaném pro doručování zpráv.
- Potenciálně rychlejší mechanismus doručení je dosažen pro přenos malých přechodných zpráv mezi správci front ve skupině sdílení front (i když jsou stejní správci front v klastru).
- Pro doručení zpráv je k dispozici více cest (tj. trasa klastru i trasa řazení do front v rámci skupiny).
- Trasa řazení do front v rámci skupiny, která je potenciálně rychlejší, je vybrána přednostně před trasou klastru. V závislosti na charakteristice zprávy může být doručení zprávy rozděleno mezi dvě cesty. Proto mohou být zprávy doručeny mimo pořadí. Je důležité si uvědomit, že toto doručení je možné bez ohledu na volbu MQOO_BIND_* určenou aplikací. Řazení do front v rámci skupiny distribuuje zprávy stejným způsobem jako klastrování v závislosti na tom, zda je volba MQOO_BIND_NOT_FIXED, MQOO_BIND_ON_OPEN, MQOO_BIND_ON_GROUP nebo MQOO_BIND_AS_AS_Q_DEF zadána při otevření.
- Po výběru přenosové cesty a umístění zpráv do přenosových front se pro doručení zpráv použije pouze vybraná přenosová cesta. Všechny nezpracované zprávy v systému SYSTEM.QSG.TRANSMIT.QUEUE nejsou přesměrovány na SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Klastrování, řazení do front v rámci skupiny a distribuované řazení do front

Je možné konfigurovat správce front, který je členem klastru, stejně jako skupiny sdílení front a je připojen k distribuovanému správci front pomocí dvojice odesílací/přijímací kanál.

Tato konfigurace je kombinací distribuovaného řazení do front s řazením do front v rámci skupiny a klastrování s řazením do front v rámci skupiny.

Řazení do front v rámci skupiny je popsáno v tématu [“Distribuované řazení do front s řazením do front v rámci skupiny \(více cest doručení\)”](#) na stránce 210.

Klastrování s řazením do front v rámci skupiny je popsáno v tématu [“Klastrování s řazením do front v rámci skupiny \(více cest doručení\)”](#) na stránce 212.

Řazení zpráv do front v rámci skupiny

Tento oddíl popisuje zprávy vkládané do SYSTEM.QSG.TRANSMIT.QUEUE.

Struktura zprávy

Stejně jako všechny ostatní zprávy, které jsou vkládány do přenosových front, i zprávy, které jsou vkládány do SYSTEM.QSG.TRANSMIT.QUEUE mají předponu v záhlaví přenosové fronty (MQXQH).

Trvalost zpráv

V produktu IBM WebSphere MQ 5.3 a vyšším podporují sdílené fronty trvalé i přechodné zprávy.

Pokud se správce front ukončí v době, kdy agent IGQ zpracovává dočasné zprávy, nebo pokud se agent IGQ ukončí abnormálně během zpracování zpráv, mohou být zpracované přechodné zprávy ztraceny. Aplikace musí zajistit obnovu dočasných zpráv, pokud je jejich obnova požadována.

Pokud požadavek vložení pro dočasnou zprávu vydaný agentem IGQ neočekávaně selže, zpracovaná zpráva bude ztracena.

Doručování zpráv

Agent IGQ načítá a doručuje všechny přechodné zprávy mimo rozsah synchronizačního bodu a všechny trvalé zprávy v rozsahu synchronizačního bodu. V tomto případě agent IGQ vystupuje jako koordinátor synchronizačních bodů. Agent IGQ proto zpracovává přechodné zprávy, jako např. rychlé, přechodné zprávy, které jsou zpracovávány v kanálu zpráv. Viz [Rychlé, přechodné zprávy](#).

Dávkování zpráv

Agent IGQ používá pevnou velikost dávky 50 zpráv. Všechny trvalé zprávy načtené v rámci dávky jsou potvrzeny v intervalech 50 zpráv. Agent potvrdí dávku sestávající z trvalých zpráv, pokud nejsou k dispozici žádné další zprávy pro načtení v systému SYSTEM.QSG.TRANSMIT.QUEUE.

Velikost zprávy

Maximální velikost zprávy, kterou lze vložit do SYSTEM.QSG.TRANSMIT.QUEUE je maximální podporovaná délka zpráv pro sdílené fronty minus délka záhlaví přenosové fronty (MQXQH).

Výchozí perzistence zpráv a výchozí priorita zpráv

Pokud je SYSTEM.QSG.TRANSMIT.QUEUE je v cestě rozlišení názvu fronty vytvořené v době otevření. Pro zprávy, které jsou vloženy s výchozí perzistencí a výchozí prioritou (nebo s výchozí perzistencí nebo výchozí prioritou), jsou při výběru fronty, která má použitou výchozí prioritu a hodnoty perzistence, použita normální pravidla. (Další informace o pravidlech výběru fronty naleznete v části [IBM MQ zprávy](#).)

Související pojmy

[“Nedoručené/nezpracované zprávy” na stránce 215](#)

Toto téma popisuje, co se stane s nedoručenými a nezpracovanými zprávami v systému SYSTEM.QSG.TRANSMIT.QUEUE.

[“Zprávy sestavy-hromadné řazení do fronty” na stránce 216](#)

Toto téma popisuje zprávy sestavy: Potvrzení příjezdu, potvrzení doručení, zpráva o vypršení platnosti a zpráva o výjimce.

Nedoručené/nezpracované zprávy

Toto téma popisuje, co se stane s nedoručenými a nezpracovanými zprávami v systému SYSTEM.QSG.TRANSMIT.QUEUE.

Pokud agent IGQ nemůže doručit zprávu do cílové fronty, agent IGQ:

- Respektuje volbu sestavy MQRO_DISCARD_MSG (pokud pole Volby sestavy deskriptoru MQMD pro nedoručenou zprávu označuje, že musí) a vyřadí nedoručenou zprávu.
- Pokusí se umístit nedoručenou zprávu do fronty nedoručených zpráv pro cílového správce front, pokud tato zpráva dosud nebyla zahozena. Agent IGQ předchází zprávu záhlavím fronty nedoručených zpráv (MQDLH).

Není-li fronta nedoručených zpráv definována nebo nelze-li nedoručenou zprávu vložit do fronty nedoručených zpráv a je-li nedoručená zpráva:

- perzistentní, agent IGQ odvolá aktuální dávku trvalých zpráv, které zpracovává, a přejde do stavu opakování. Další informace viz téma [“Specifické vlastnosti řazení do front v rámci skupiny” na stránce 217](#).

- dočasný, agent IGQ zprávu vyřadí a pokračuje ve zpracování další zprávy.

Pokud je správce front ve skupině sdílení front ukončen dříve, než má přidružený agent IGQ čas na zpracování všech svých zpráv, nezpracované zprávy zůstanou v systému SYSTEM.QSG.TRANSMIT.QUEUE až do dalšího spuštění správce front. Agent IGQ poté načte a doručí zprávy do cílových front.

Dojde-li k selhání prostředku Coupling Facility před všemi zprávami v systému SYSTEM.QSG.TRANSMIT.QUEUE byla zpracována, všechny nezpracované přechodné zprávy jsou ztraceny.

Produkt IBM doporučuje, aby aplikace nevládaly zprávy přímo do přenosových front. Pokud aplikace vloží zprávy přímo do SYSTEM.QSG.TRANSMIT.QUEUE, agent IGQ nemusí být schopen zpracovat tyto zprávy a zůstanou v systému SYSTEM.QSG.TRANSMIT.QUEUE. Uživatelé pak musí použít své vlastní metody, aby se vypořádali s těmito nezpracovanými zprávami.

Zprávy sestavy-hromadné řazení do fronty

Toto téma popisuje zprávy sestavy: Potvrzení příjezdu, potvrzení doručení, zpráva o vypršení platnosti a zpráva o výjimce.

Potvrzení o příjezdu (COA)/potvrzení o doručení (COD) zpráva zprávy

Zprávy COA a COD jsou generovány správcem front při použití front v rámci skupiny.

Zprávy sestavy vypršení platnosti

Zprávy sestavy vypršení platnosti jsou generovány správcem front.

Zprávy sestavy výjimek

V závislosti na volbě sestavy MQRO_EXCEPTION_* zadané v poli *Volby sestavy* deskriptoru zprávy pro nedoručenou zprávu agent IGQ vygeneruje požadovanou sestavu výjimek a umístí ji do určené fronty pro odpověď. Řazení do fronty v rámci skupiny lze použít k doručení zprávy o výjimce do cílové fronty pro odpověď.

Perzistence zprávy sestavy je stejná jako perzistence nedoručené zprávy. Pokud se agentovi IGQ nepodaří interpretovat název cílové fronty pro odpověď nebo pokud se mu nepodaří vložit zprávu odpovědi do přenosové fronty (pro následný přenos do cílové fronty pro odpověď), pokusí se vložit zprávu výjimky do fronty nedoručených zpráv správce front, ve kterém je zpráva sestavy generována. Pokud to není možné, pak v případě, že nedoručená zpráva je:

- perzistentní, agent IGQ zruší sestavu výjimek, odvolá aktuální dávku zpráv a přejde do stavu opakování. Další informace viz téma [“Specifické vlastnosti řazení do front v rámci skupiny”](#) na stránce 217.
- Dočasný, agent IGQ zruší sestavu výjimek a pokračuje ve zpracování další zprávy v systému SYSTEM.QSG.TRANSMIT.QUEUE.

Zabezpečení pro řazení do front v rámci skupiny

Toto téma popisuje bezpečnostní opatření pro řazení do front v rámci skupiny.

Atributy správce front IGQAUT (oprávnění IGQ) a IGQUSER (ID uživatele agenta IGQ) lze nastavit tak, aby řídily úroveň kontroly zabezpečení, která se provádí při otevření cílových front agentem IGQ.

Oprávnění k řazení do fronty v rámci skupiny (IGQAUT)

Atribut IGQAUT lze nastavit tak, aby označoval typ kontrol zabezpečení, které se mají provést, a tím určoval ID uživatelů, která mají být použita agentem IGQ při vytváření oprávnění pro vložení zpráv do cílové fronty.

Atribut IGQAUT je analogický atributu PUTAUT, který je k dispozici v definicích kanálu.

Identifikátor uživatele řazení do front v rámci skupiny (IGQUSER)

Atribut IGQUSER lze použít k nominaci ID uživatele, které má agent IGQ použít, když vytváří oprávnění pro vložení zpráv do cílové fronty.

Atribut IGQUSER je analogický s atributem MCAUSER, který je k dispozici v definicích kanálu.

Tento oddíl popisuje specifické vlastnosti front v rámci skupiny, včetně zneplatnění manipulátorů objektů, schopnosti vlastního zotavení a opakovaného pokusu agenta front v rámci skupiny a agenta front v rámci skupiny a serializace.

Zneplatnění manipulátorů objektů (MQRC_OBJECT_CHANGED)

Pokud se po otevření objektu zjistí, že se změnila atributy objektu, správce front při příštím použití zruší platnost manipulátoru objektu s hodnotou MQRC_OBJECT_CHANGED.

Řazení do front v rámci skupiny zavádí následující pravidla pro zneplatnění popisovače objektu:

- Pokud je SYSTEM.QSG.TRANSMIT.QUEUE byla zahrnuta do cesty rozpoznání názvu během otevřeného zpracování, protože v době otevření bylo POVOLENO řazení do fronty v rámci skupiny, ale v době vložení bylo zjištěno, že zařazení do fronty v rámci skupiny je ZAKÁZÁNO, pak správce front zneplatní manipulátor objektu a požadavek vložení selže s MQRC_OBJECT_CHANGED.
- Pokud je SYSTEM.QSG.TRANSMIT.QUEUE nebyla zahrnuta do cesty rozpoznání názvu během otevřeného zpracování, protože v době otevření bylo řazení do fronty v rámci skupiny ZAKÁZÁNO, ale v době vložení bylo zjištěno, že je ve frontě v rámci skupiny POVOLENO, pak správce front ruší platnost popisovače objektu a požadavek vložení s MQRC_OBJECT_CHANGED selže.
- Pokud je SYSTEM.QSG.TRANSMIT.QUEUE byla zahrnuta do cesty pro překlad názvů během otevřeného zpracování, protože v době otevření bylo povoleno řazení do front v rámci skupiny, ale byl aktivován systém SYSTEM.QSG.TRANSMIT.QUEUE změnila podle času vložení, pak správce front zneplatní manipulátor objektu a požadavek vložení selže s hodnotou MQRC_OBJECT_CHANGED.

Vlastní obnova agenta řazení do front v rámci skupiny

Pokud se agent IGQ ukončí abnormálně, vydá se zpráva CSQM067E a agent IGQ se spustí znovu.

Schopnost opakování agenta řazení do fronty v rámci skupiny

Pokud agent IGQ narazí na problém s přístupem k systému SYSTEM.QSG.TRANSMIT.QUEUE (protože není definována, například, nebo je definována s nesprávnými atributy, nebo je blokována pro Gets, nebo z jiného důvodu), přejde agent IGQ do stavu opakování.

Agent IGQ sleduje počet krátkých a dlouhých opakování a intervaly. Hodnoty pro tyto počty a intervaly, které nelze změnit, jsou následující:

<i>Tabulka 20. Hodnoty počtu krátkých a dlouhých opakování a intervalů</i>	
Konstanta	Hodnota
Počet krátkých opakování	10
Interval krátkých opakování	60 sekund = 1 min
Počet dlouhých opakování	999,999,999
Interval dlouhých opakování	1200 sekund = 20 min

Agent řazení do front v rámci skupiny a serializace

Pokus agenta IGQ o serializaci přístupu ke sdíleným frontám v době, kdy stále probíhá obnova typu peer, může selhat.

Pokud dojde k selhání správce front ve skupině sdílení front v době, kdy agent IGQ pracuje s nepotvrzenými zprávami ve sdílené frontě nebo frontách, agent IGQ skončí a pro selhávajícího správce front proběhne zotavení typu peer se sdílenou frontou. Vzhledem k tomu, že zotavení typu peer sdílené fronty je asynchronní aktivita, ponechává možnost pro selhávajícího správce front a také agenta IGQ pro tohoto správce front restartovat před dokončením zotavení typu peer sdílené fronty. Což zase ponechává

možnost, aby všechny potvrzené zprávy byly zpracovány před a mimo pořadí se zprávami, které jsou stále obnovovány. Chcete-li se ujistit, že zprávy nejsou zpracovány mimo pořadí, agent IGQ serializuje přístup ke sdíleným frontám zadáním volání rozhraní API MQCONN.

Pokus agenta IGQ o serializaci přístupu ke sdíleným frontám v době, kdy stále probíhá obnova typu peer, může selhat. Je vydána chybová zpráva a agent IGQ je uveden do stavu opakování. Po dokončení obnovy rovnocenných uzlů správce front, například v době dalšího opakování, může být agent IGQ spuštěn.

z/OS Správa úložiště na systému z/OS

Produkt IBM MQ for z/OS vyžaduje trvalé a dočasné datové struktury a k uložení těchto dat používá sady stránek a vyrovnávací paměti. Tato témata poskytují další podrobnosti o tom, jak produkt IBM MQ používá tyto sady stránek a vyrovnávací paměti.

Související pojmy

[“Sady stránek pro IBM MQ for z/OS” na stránce 218](#)

Toto téma popisuje, jak produkt IBM MQ for z/OS používá sady stránek k ukládání zpráv.

[“Třídy úložiště pro IBM MQ for z/OS” na stránce 219](#)

Paměťová třída je koncept IBM MQ for z/OS, který umožňuje správci front mapovat fronty na sady stránek. Paměťové třídy můžete použít k řízení toho, které datové sady jsou používány jednotlivými frontami.

[“Vyrovnávací paměti a fondy vyrovnávacích pamětí pro IBM MQ for z/OS” na stránce 220](#)

Produkt IBM MQ for z/OS používá vyrovnávací paměti a fondy vyrovnávacích pamětí k dočasnému ukládání dat do mezipaměti. V tomto tématu se dozvíte, jak jsou vyrovnávací paměti uspořádány a používány.

Související odkazy

[“Kde najít další informace o správě úložiště pro IBM MQ for z/OS” na stránce 222](#)

Toto téma použijte jako odkaz na další informace o správě úložiště pro produkt IBM MQ for z/OS.

z/OS Sady stránek pro IBM MQ for z/OS

Toto téma popisuje, jak produkt IBM MQ for z/OS používá sady stránek k ukládání zpráv.

Sada stránek je lineární datová sada VSAM, která byla speciálně formátována pro použití produktem IBM MQ. Sady stránek se používají k ukládání většiny zpráv a definic objektů.

Výjimkou jsou globální definice, které jsou uloženy ve sdíleném úložišti v systému Db2, a zprávy ve sdílených frontách. Nejsou uloženy v sadách stránek správce front. Další informace o sdílených frontách viz [“Sdílené fronty a skupiny sdílení front” na stránce 161a](#) další informace o globálních definicích viz [Soukromé a globální definice](#).

Sady stránek IBM MQ mohou mít velikost až 64 GB. Každá sada stránek je identifikována identifikátorem sady stránek (PSID), celým číslem v rozsahu 00 až 99. Každý správce front musí mít své vlastní sady stránek.

Produkt IBM MQ používá sadu stránek nula (PSID=00) k ukládání definic objektů a dalších důležitých informací týkajících se správce front. Pro normální provoz produktu IBM MQ je nezbytné, aby sada stránek nula nebyla plná, takže ji nepoužívejte k ukládání zpráv.

Chcete-li zlepšit výkon systému, měli byste také oddělit krátkodobé zprávy od zpráv s dlouhou životností jejich umístěním na různé sady stránek.

Musíte formátovat sady stránek a produkt IBM MQ k tomu poskytuje obslužný program `FORMAT`; viz [Formátovací sady stránek \(FORMAT\)](#). Sady stránek musí být také definovány pro subsystém IBM MQ.

IBM MQ for z/OS lze nakonfigurovat tak, aby dynamicky rozbalovala sadu stránek, pokud se naplní. Produkt IBM MQ v případě potřeby pokračuje v rozbalování sady stránek, dokud nebude existovat 123 logických oblastí, pokud bude k dispozici dostatek místa na disku. Oblasti pro rozšíření mohou zahrnovat svazky, pokud je lineární datová sada definována tímto způsobem, avšak produkt IBM MQ nemůže rozšířit sady stránek nad 64 GB.

Nelze použít sady stránek z jednoho správce front IBM MQ v jiném správcí front IBM MQ ani změnit název správce front. Chcete-li přenést data z jednoho správce front do jiného, musíte uvolnit všechny objekty a zprávy z prvního správce front a znovu je načíst do jiného.

Nelze použít sady stránek větší než 4 GB ve správcí front se starší verzí než V6. Během období migrace, kdy je pravděpodobné, že se budete muset vrátit k předchozí verzi kódu:

- Neměňte sadu stránek 0 tak, aby byla větší než 4 GB.
- Ostatní sady stránek větší než 4 GB budou při restartování správce front s předchozí verzí ponechány v režimu offline.

Další informace o migraci existujících sad stránek schopných rozšíření nad 4 GB naleznete v tématu [Definování sady stránek větší než 4 GB](#).

Administrátor může dynamicky přidávat sady stránek do spuštěného správce front nebo odebírat sady stránek ze spuštěného správce front (s výjimkou sady stránek nula). Příkaz DEFINE PSID lze spustit po dokončení restartu správce front, pouze pokud příkaz obsahuje klíčové slovo DSN.

Třídy úložiště pro IBM MQ for z/OS

Paměťová třída je koncept IBM MQ for z/OS, který umožňuje správcí front mapovat fronty na sady stránek. Paměťové třídy můžete použít k řízení toho, které datové sady jsou používány jednotlivými frontami.

Zavedení paměťových tříd

Úložná třída mapuje jednu nebo více front na sadu stránek. To znamená, že zprávy pro tuto frontu jsou uloženy v dané sadě stránek.

Paměťové třídy vám umožňují řídit, kde jsou uložena nesdílená data zpráv pro účely administrace, správy prostoru datových sad a správy zátěže nebo izolace aplikací. Paměťové třídy můžete také použít k definování skupiny XCF a názvu člena oblasti IMS, používáte-li most IMS (viz [“IBM MQ a IMS” na stránce 275](#)).

Sdílené fronty nepoužívají paměťové třídy k získání mapování sady stránek, protože zprávy v nich nejsou uloženy v sadách stránek.

Jak fungují paměťové třídy

- Paměťovou třídu definujete pomocí příkazu DEFINE STGCLASS s uvedením identifikátoru sady stránek (PSID).
- Při definování fronty zadáváte paměťovou třídu v atributu STGCLASS.

V následujícím příkladu je lokální fronta QE5 mapována na sadu stránek 21 prostřednictvím paměťové třídy ARC2.

```
DEFINE STGCLASS(ARC2) PSID(21)
DEFINE QLOCAL(QE5) STGCLASS(ARC2)
```

To znamená, že zprávy, které jsou vloženy do fronty QE5, jsou uloženy v sadě stránek 21 (pokud zůstanou ve frontě dostatečně dlouho na to, aby mohly být zapsány do DASD).

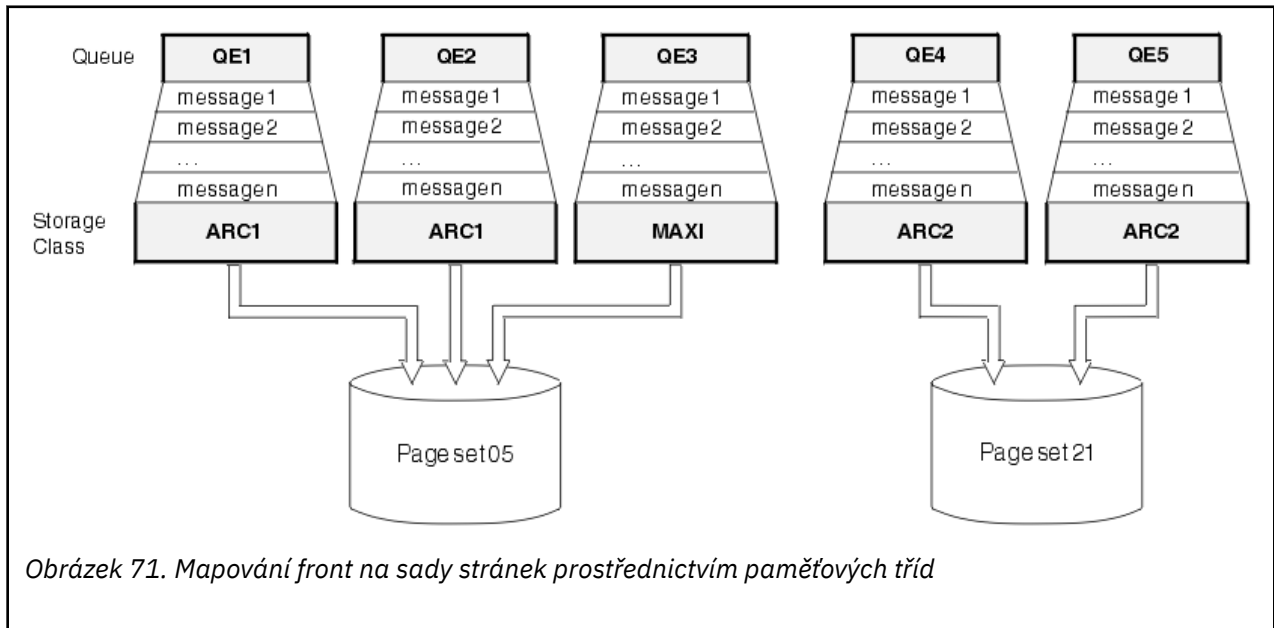
Více než jedna fronta může používat stejnou paměťovou třídu a můžete definovat libovolný počet paměťových tříd. Můžete například rozšířit předchozí příklad tak, aby obsahoval více definic paměťových tříd a front, a to následujícím způsobem:

```

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) ...

```

V produktu [Obrázek 71](#) na stránce 220 jsou obě úložné třídy ARC1 a MAXI přidruženy k sadě stránek 05. Proto jsou fronty QE1, QE2 a QE3 mapovány na sadu stránek 05. Podobně paměťová třída ARC2 přidružuje fronty QE4 a QE5 k sadě stránek 21.



Pokud definujete frontu bez uvedení paměťové třídy, produkt IBM MQ použije výchozí paměťovou třídu.

Pokud je zpráva vložena do fronty, která pojmenovává neexistující paměťovou třídu, aplikace obdrží chybu. Je třeba změnit definici fronty tak, aby byla pojmenována existující paměťová třída, nebo vytvořit paměťovou třídu pojmenovanou frontou.

Paměťovou třídu můžete změnit pouze v následujících případech:

- Všechny fronty, které používají tuto paměťovou třídu, jsou prázdné a nemají žádnou nepotvrzenou aktivitu.
- Všechny fronty, které používají tuto paměťovou třídu, jsou zavřeny.

z/OS **Vyrovnávací paměti a fondy vyrovnávacích pamětí pro IBM MQ for z/OS**

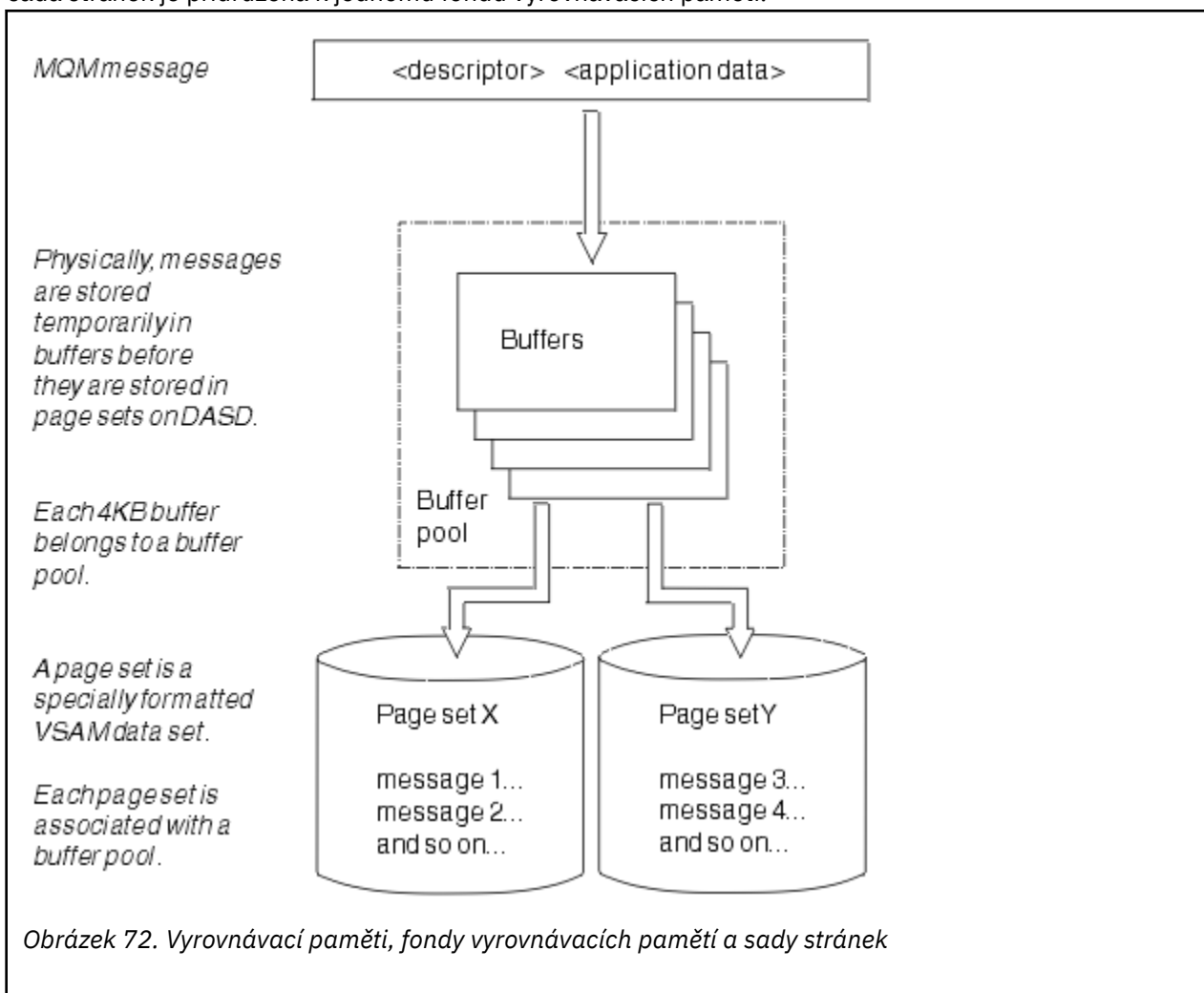
Produkt IBM MQ for z/OS používá vyrovnávací paměti a fondy vyrovnávacích pamětí k dočasnému ukládání dat do mezipaměti. V tomto tématu se dozvíte, jak jsou vyrovnávací paměti uspořádány a používány.

Z důvodu efektivity produkt IBM MQ používá formu ukládání do mezipaměti, kdy jsou zprávy (a definice objektů) dočasně uloženy ve vyrovnávacích pamětech, než jsou uloženy v sadách stránek na serveru DASD. Krátkodobé zprávy, tj. zprávy, které jsou načteny z fronty krátce po jejich přijetí, mohou být uloženy pouze ve vyrovnávacích pamětech. Tato aktivita ukládání do mezipaměti je řízena správcem vyrovnávací paměti, který je komponentou produktu IBM MQ.

Vyrovnávací paměti jsou uspořádány do *fondů vyrovnávacích pamětí*. Pro každého správce front můžete definovat až 100 fondů vyrovnávacích pamětí (0 až 99).

Doporučuje se použít minimální počet fondů vyrovnávacích pamětí, který je konzistentní s oddělením objektů a typů zpráv uvedeným v části **Obrázek 72** na stránce 221, a případné požadavky na izolaci dat, které vaše aplikace může mít. Každá vyrovnávací paměť je dlouhá 4 kB. Fondy vyrovnávacích pamětí standardně používají 31bitové úložiště, v tomto režimu je maximální počet vyrovnávacích pamětí určen množstvím 31bitového úložiště, které je k dispozici v adresním prostoru správce front; pro vyrovnávací paměti nepoužívejte více než 70%. Alternativně lze alokaci úložiště fondu vyrovnávacích pamětí provést z 64bitového úložiště (použijte atribut LOCATION příkazu **DEFINE BUFFPOOL**). Použití příkazu LOCATION (ABOVE) tak, aby bylo použito 64bitové úložiště, má dvě výhody. Za prvé je k dispozici mnohem více 64bitového úložiště, takže fondy vyrovnávacích pamětí mohou být mnohem větší, a za druhé, 31 bitové úložiště je k dispozici pro použití jinými funkcemi. Obvykle platí, že čím více vyrovnávacích pamětí máte, tím efektivnější je ukládání do vyrovnávací paměti a tím lepší je výkon systému IBM MQ.

Obrázek 72 na stránce 221 zobrazuje vztah mezi zprávami, vyrovnávacími pamětmi, fondy vyrovnávacích pamětí a sadami stránek. Fond vyrovnávacích pamětí je přidružen k jedné nebo více sadám stránek; každá sada stránek je přidružená k jednomu fondu vyrovnávacích pamětí.



Pomocí příkazu **ALTER BUFFPOOL** můžete dynamicky zadávat příkazy pro úpravu velikosti a umístění fondu vyrovnávacích pamětí. Sady stránek lze dynamicky přidávat pomocí příkazu **DEFINE PSID** nebo lze odstranit pomocí příkazu **DELETE PSID**.

Pokud je fond vyrovnávacích pamětí příliš malý, IBM MQ vydá zprávu **CSQP020E**. Poté můžete dynamicky přidávat další vyrovnávací paměti do ovlivněného fondu vyrovnávacích pamětí (všimněte si, že k tomu budete muset odebrat vyrovnávací paměti z jiných fondů vyrovnávacích pamětí).

Pomocí příkazu **DEFINE BUFFPOOL** můžete určit počet vyrovnávacích pamětí ve fondu a pomocí příkazu **ALTER BUFFPOOL** můžete dynamicky měnit velikost fondů vyrovnávacích pamětí. Aktuální počet vyrovnávacích pamětí ve fondu určíte dynamicky zobrazením sady stránek, která používá fond vyrovnávacích pamětí, pomocí příkazu **DISPLAY USAGE**.

Z důvodů výkonu nekládejte zprávy a definice objektů do stejného fondu vyrovnávacích pamětí. Použijte jeden fond vyrovnávacích pamětí (například číslo nula) výhradně pro sadu stránek nula, kde jsou uchovávány definice objektů. Podobně uchovávejte zprávy s krátkou životností a zprávy s dlouhou životností v různých fondech vyrovnávacích pamětí, a tedy v různých sadách stránek a v různých frontách.

Příkaz **DEFINE BUFFPOOL** nelze po restartu použít k vytvoření nového fondu vyrovnávacích pamětí. Místo toho, pokud příkaz **DEFINE PSID** používá klíčové slovo DSN, může explicitně identifikovat fond vyrovnávacích pamětí, který není momentálně definován. Poté bude vytvořen nový fond vyrovnávacích pamětí.

► z/OS Kde najít další informace o správě úložiště pro IBM MQ for z/OS

Toto téma použijte jako odkaz na další informace o správě úložiště pro produkt IBM MQ for z/OS.

Další informace o tématech v této sekci naleznete z následujících zdrojů:

Tabulka 21. Kde najít další informace o správě úložiště	
Téma	Kam se podívat
Kolik úložného prostoru potřebujete	Plánování požadavků na úložiště a výkon na systému z/OS
Jak velké, aby se vaše stránky sady a fondy vyrovnávacích pamětí	Naplánujte sady stránek a fondy vyrovnávacích pamětí
Správa sad stránek	Správa sad stránek
Příkazy MQSC	Příkazy MQSC

► z/OS přihlášení IBM MQ for z/OS

IBM MQ udržuje *protokoly* změn dat a významných událostí, jak k nim dochází. Tyto protokoly lze v případě potřeby použít k obnovení dat do předchozího stavu.

Datová sada samozavedení (BSDS) ukládá informace o datových sadách, které obsahují protokoly.

Protokol neobsahuje informace pro statistiku, trasování nebo vyhodnocení výkonu. Další podrobnosti o statistických a monitorovacích informacích, které produkt IBM MQ shromažďuje, viz [Monitorování a statistika](#).

Další informace o protokolování naleznete v následujících tématech:

- [“Soubory protokolu v adresáři IBM MQ for z/OS”](#) na stránce 223
- [“Jak je protokol strukturován”](#) na stránce 227
- [“Jak se zapisují protokoly IBM MQ for z/OS”](#) na stránce 227
- [“Větší relativní bajtová adresa protokolu”](#) na stránce 230
- [“Datová sada zaváděcího programu”](#) na stránce 231


Související úlohy

[Plánování prostředí protokolování](#)

[Nastavení protokolů pomocí modulu parametrů systému](#)

► z/OS [Správa serveru z/OS](#)

Související odkazy

 [Zprávy pro IBM MQ for z/OS](#)

Soubory protokolu v adresáři IBM MQ for z/OS

Soubory protokolu obsahují informace potřebné pro zotavení transakcí. Aktivní soubory protokolu lze archivovat, abyste mohli uchovávat data protokolu po dlouhou dobu.

Co je to soubor protokolu

Produkt IBM MQ zaznamenává všechny významné události, které se vyskytnou v *aktivním protokolu*. Protokol obsahuje informace potřebné k obnově:

- Trvalé zprávy
- objekty IBM MQ , například fronty
- Správce front IBM MQ

Aktivní protokol se skládá z kolekce datových sad (až 310), které se používají cyklicky.

Archivování protokolů můžete povolit tak, aby při zaplňování aktivního protokolu byla vytvořena kopie v datové sadě archivu. Použití archivace vám umožňuje uchovávat data protokolu po delší dobu. Pokud nepoužijete archivaci, dojde k přepsání zalamování protokolů a předchozích dat. Chcete-li obnovit sadu stránek nebo obnovit data ve struktuře prostředku CF, potřebujete data protokolu z doby, kdy byla provedena záloha sady stránek nebo struktury. Archivní protokol lze vytvořit na disku nebo na pásce.

Archivace

Vzhledem k tomu, že aktivní protokol má pevnou velikost, produkt IBM MQ pravidelně kopíruje obsah každé datové sady protokolu do *archivního protokolu*, což je obvykle datová sada na paměťovém zařízení s přímým přístupem (DASD) nebo na magnetické pásce. Pokud dojde k selhání subsystému nebo transakce, produkt IBM MQ použije k zotavení aktivní protokol a v případě potřeby protokol archivace.

Protokol archivace může obsahovat až 1000 sekvenčních datových sad. Každou datovou sadu můžete katalogizovat pomocí funkce ICF (integrated catalog facility) z/OS .

Archivace je nezbytnou součástí obnovy systému IBM MQ . Pokud je jednotka zotavení přerušitelná, záznamy protokolu v rámci této jednotky zotavení mohou být nalezeny v protokolu archivu. V tomto případě obnova vyžaduje data z protokolu archivu. Pokud je však archivace zakázána, aktivní protokol s novými záznamy protokolu se obtéká a přepisuje dřívější záznamy protokolu. To znamená, že produkt IBM MQ nemusí být schopen vrátit zpět jednotku zotavení a zprávy mohou být ztraceny. Správce front je poté nestandardně ukončen.

Proto v produkčním prostředí **nikdy nevypínejte archivaci**. Pokud tak učiníte, riskujete ztrátu dat po selhání systému nebo transakce. Pouze v případě, že pracujete v testovacím prostředí, můžete zvážit vypnutí archivace. Pokud to potřebujete provést, použijte makro CSQ6LOGP , které je popsáno v tématu [Použití CSQ6LOGP](#).

Produkt IBM MQ vydává zprávu (CSQJ160I nebo CSQJ161I), která pomáhá předcházet problémům s neplánovanými přerušitelnou jednotkou práce. pokud je během aktivního zpracování odlehčování protokolu zjištěna přerušitelná jednotka práce.

duální protokolování

V duálním protokolování se každý záznam protokolu zapisuje do dvou různých datových sad aktivního protokolu, aby se minimalizovala pravděpodobnost problémů se ztrátou dat během restartu.

Produkt IBM MQ můžete nakonfigurovat tak, aby se spouštěl s *jedním protokolováním* nebo *duálním protokolováním*. S jedním protokolováním se záznamy protokolu zapisují jednou do datové sady aktivního

protokolu. Každá datová sada aktivního protokolu je LDS (single-extent in-VSAM linear data set). S duálním protokolováním se každý záznam protokolu zapisuje do dvou různých datových sad aktivního protokolu. Duální protokolování minimalizuje pravděpodobnost problémů se ztrátou dat během restartu.

Posunování protokolu

Posunování protokolu způsobí, že se záznamy protokolu pro některé pracovní jednotky zapíší dále do protokolu. Tím se snižuje množství dat protokolu, která musí být načtena při restartu správce front nebo při odvolání, pro dlouhodobě spuštěné nebo dlouhodobě pochybné pracovní jednotky.

Když je jednotka práce považována za dlouhou, znázornění každého záznamu protokolu se zapíše dále do protokolu. Tato technika se nazývá *posunování*. Po zpracování celé jednotky práce je jednotka práce ve stavu *posunutá*. Jakákoli zpětná nebo restartovaná aktivita související s posunutou pracovní jednotkou může použít posunuté záznamy protokolu místo původních záznamů pracovního protokolu.

Detekce přerušitelné pracovní jednotky je funkcí procesu kontrolního bodu. V čase kontrolního bodu je každá aktivní jednotka práce zkontrolována, aby se zjistilo, zda je třeba ji posunovat. Pokud jednotka práce prošla dvěma předchozími kontrolními body od doby, kdy byla vytvořena, nebo od doby, kdy byla naposledy posunutá, je jednotka práce vhodná k posunu. To znamená, že jedna jednotka práce může být posunutá více než jednou. Toto je známé jako *víceposunutá* jednotka práce.

Jednotka práce je posunutá každé tři kontrolní body. Kontrolní bod se však provádí asynchronně k přepínači protokolu (nebo k zápisu záznamu protokolu, který způsobil překročení hodnoty LOGLOAD).

V daném okamžiku probíhá pouze jeden kontrolní bod, takže před dokončením kontrolního bodu může existovat více přepínačů protokolu.

To znamená, že pokud není dostatek aktivních protokolů, nebo pokud jsou příliš malé, nemusí být posunování velké pracovní jednotky dokončeno před vyplněním všech protokolů.

Zpráva [CSQR027I](#) se zobrazí v případě, že posunování nelze dokončit.

Je-li archivace protokolu vypnuta, dojde k chybě ABEND 5C6 s příčinou 00D1032A v případě, že dojde k pokusu o vrácení pracovní jednotky, pro kterou došlo k selhání posunu. Chcete-li se vyhnout tomuto problému, měli byste použít OFFLOAD=YES.

Posunování protokolu je vždy aktivní a spouští se bez ohledu na to, zda je povolena archivace protokolu.

Poznámka: Ačkoli jsou všechny záznamy protokolu pro jednotku práce posunuté, není celý obsah každého záznamu posunutý, pouze část, která je nezbytná pro vrácení zpět. To znamená, že množství zapsaných dat protokolu je udržováno na minimu a že posunuté záznamy nelze použít, pokud dojde k selhání sady stránek. Jednotka práce s dlouhou dobou zpracování je taková, která byla spuštěna pro více než tři kontrolní body správce front.

Další informace o posunu protokolu naleznete v tématu [Správa protokolů](#).

Kompresie protokolů

Produkt IBM MQ for z/OS můžete nakonfigurovat tak, aby komprimoval a dekomprimoval záznamy protokolu při jejich zápisu a čtení z datové sady protokolu.

Pomocí komprese protokolu lze snížit množství dat zapsaných do protokolu pro trvalé zprávy v soukromých frontách. Velikost dosažené komprese závisí na typu dat obsažených ve zprávách. Například funkce RLE (Run Length Encoding) pracuje s optimalizací opakovaných instancí bajtů, které mohou efektivně poskytovat dobré výsledky pro strukturovaná data nebo data orientovaná na záznamy.



Upozornění: Trvalé zprávy, které jsou vkládány do sdílené fronty, nepodléhají kompresi protokolu.

Můžete použít pole v sekci Správce protokolu záznamů SMF (System Management Facility 115), abyste monitorovali, kolik komprese dat je dosaženo. Další informace o nástroji SMF naleznete v tématu [Použití zařízení pro správu systému a Zprávy evidence a statistiky](#).

Kompresi protokolu zvyšuje využití procesoru systému. Kompresi byste měli zvážit pouze v případě, že je propustnost správce front omezena zápisem šířky pásma I/O do datových sad protokolu nebo je omezena diskovým úložištěm potřebným k uchování datových sad protokolu. Používáte-li sdílené fronty, lze omezení šířky pásma I/O odlehčovat přidáním dalších správců front do skupiny sdílení front a distribuováním pracovní zátěže mezi více správců front.

Volbu komprese protokolu lze povolit a zakázat podle potřeby bez nutnosti zastavit a restartovat správce front. Správce front může číst libovolné komprimované záznamy protokolu bez ohledu na aktuální nastavení komprese protokolu.

Správce front podporuje 3 nastavení pro kompresi protokolu.

ŽÁDNÉ

Nepoužívá se žádná komprese dat protokolu. Toto je výchozí hodnota.

RLE

Kompresi dat protokolu se provádí pomocí kódování RLE (run-length encoding).

ANY

Povolte správci front výběr algoritmu komprese, který poskytuje nejvyšší stupeň komprese záznamu protokolu. Výsledkem této volby je komprese RLE.

Kompresi záznamů protokolu můžete řídit pomocí jedné z následujících možností:

- Příkazy SET a DISPLAY LOG v prostředí MQSC; viz [SET LOG](#) a [DISPLAY LOG](#) .
- Funkce Nastavit protokol a dotazovat se na protokol v rozhraní PCF; viz [Nastavit protokol a dotazovat se na protokol](#) .
- Makro CSQ6LOGP v modulu parametrů systému; viz [Použití CSQ6LOGP](#) .

Kromě toho má obslužný program pro tisk protokolu CSQ1LOGP podporu pro rozšíření všech komprimovaných záznamů protokolu.

Data protokolu

Protokol může obsahovat až 18 milionů bajtů ($1.8 \cdot 10^{19}$) bajtů. Každý bajt může být adresován svým offsetem od začátku protokolu a tento offset je znám jako *relativní bajtová adresa* (RBA).

Na adresu RBA odkazuje 6bajtové nebo 8bajtové pole, které poskytuje celkový adresovatelný rozsah 2^{48} bajtů, nebo 2^{64} bajtů, v závislosti na tom, zda se používají 6bajtové nebo 8bajtové adaptéry RBA protokolu.

Pokud však produkt IBM MQ zjistí, že použitý rozsah je vyšší než F0000000000 (pokud se používají 6bajtové RBA) nebo FFFF80000000000 (pokud se používají 8bajtové protokoly RBA), zobrazí se zprávy CSQI045, CSQI046, CSQI047a CSQJ032 , které vás upozorní na resetování protokolu RBA.

Pokud hodnota RBA dosáhne hodnoty FFF800000000 (pokud se používají 6bajtové protokoly RBA) nebo FFFFFFFC000000000 (pokud se používají 8bajtové protokoly RBA), správce front se ukončí s kódem příčiny 00D10257.

Po vydání varovných zpráv o použitém rozsahu protokolu byste měli naplánovat výpadek správce front, během kterého lze správce front převést na použití 8bajtových protokolů RBA, nebo lze protokol resetovat. Postup resetování protokolu je zdokumentován v tématu [Resetování protokolu správce front](#).

Pokud váš správce front používá 6bajtové protokoly RBAs, zvažte převod správce front na 8bajtové protokoly RBA namísto resetování protokolu správce front podle postupu uvedeného v části [Implementace větší relativní bajtové adresy protokolu](#).

Protokol se skládá z *záznamů protokolu*, z nichž každá je sadou dat protokolu považovaných za jednu jednotku. Záznam protokolu je identifikován buď RBA prvního bajtu svého záhlaví, nebo jeho pořadovým číslem záznamu protokolu (LRSN). RBA nebo LRSN jedinečně identifikuje záznam, který začíná v určitém bodě protokolu.

Zda k identifikaci bodů protokolu použijete RBA nebo LRSN, závisí na tom, zda používáte skupiny sdílení front. V prostředí sdílení front nelze použít relativní bajtovou adresu k jedinečné identifikaci bodu

protokolu, protože více správců front může aktualizovat stejnou frontu současně a každý z nich má svůj vlastní protokol. Chcete-li to vyřešit, je pořadové číslo záznamu protokolu odvozeno od hodnoty časového razítka a nemusí nutně představovat fyzické posunutí záznamu protokolu v protokolu.

Každý záznam protokolu má záhlaví, které poskytuje jeho typ, dílčí komponentu IBM MQ, která vytvořila záznam, a pro záznamy obnovy jednotku identifikátoru obnovy.

Existují čtyři typy záznamů protokolu, které jsou popsány v následujících nadpisech:

- [Záznamy protokolu jednotky zotavení](#)
- [Záznamy kontrolního bodu](#)
- [Řídící záznamy sady stránek](#)
- [Záznamy zálohy struktury CF](#)

Záznamy protokolu jednotky zotavení

Většina záznamů protokolu popisuje změny ve frontách IBM MQ. Všechny tyto změny se provádějí v jednotkách obnovy.

Produkt IBM MQ používá speciální techniky protokolování zahrnující *undo/redo* a *compensating log records* ke zkrácení doby restartu a zlepšení dostupnosti systému.

Jedním z důsledků je, že čas restartu je omezen. Dojde-li během restartu k selhání, aby bylo nutné správce front znovu spustit podruhé, není nutné při druhém restartu znovu použít všechny aktivity zotavení, které byly dokončeny do bodu selhání v prvním restartu. To znamená, že dokončení následných restartů netrvá postupně déle.

Záznamy kontrolních bodů

Chcete-li zkrátit dobu restartu, produkt IBM MQ provádí pravidelné kontrolní body během normálního provozu. Ty se vyskytují takto:

- Když byl zapsán předdefinovaný počet záznamů protokolu. Toto číslo je definováno operandem frekvence kontrolních bodů s názvem LOGLOAD makra systémového parametru CSQ6SYSP, popsaným v tématu [Použití CSQ6SYSP](#).
- Na konci úspěšného restartu.
- Při normálním ukončení.
- Kdykoli se produkt IBM MQ přepne na další datovou sadu aktivního protokolu v cyklu.

V době provedení kontrolního bodu produkt IBM MQ vydá příkaz DISPLAY CONN (popsaný v části [DISPLAY CONN](#)). interně tak, aby byl do protokolu konzoly z/OS zapsán seznam aktuálně nejistých připojení.

Řídící záznamy sady stránek

Tyto záznamy registrují sady stránek a fondy vyrovnávacích pamětí známé správcí front IBM MQ v jednotlivých kontrolních bodech a zaznamenávají informace o rozsazích protokolů potřebných k provedení obnovy médií sady stránek v době kontrolního bodu.

Určité dynamické změny sad stránek a fondů vyrovnávacích pamětí se zapisují také jako řídicí záznamy sad stránek, aby bylo možné tyto změny obnovit a automaticky obnovit při příštím spuštění správce front.

Záznamy zálohy struktury prostředku CF

Tyto záznamy uchovávají data přečtená ze struktury seznamu prostředku Coupling Facility v reakci na příkaz BACKUP CFSTRUCT. V nepravděpodobném případě selhání struktury prostředku Coupling Facility

jsou tyto záznamy spolu se záznamy jednotky zotavení použity příkazem RECOVER CFSTRUCT k provedení obnovy médií struktury prostředku Coupling Facility do bodu selhání.

Související úlohy

[Implementace větší relativní bajtové adresy protokolu](#)

Jak je protokol strukturován

Použijte toto téma, abyste porozuměli terminologii použité k popisu záznamů protokolu.

Každá datová sada aktivního protokolu musí být lineární datová sada VSAM (LDS). Fyzická výstupní jednotka zapsaná do datové sady aktivního protokolu je 4kB řídicí interval (CI). Každá KP obsahuje jeden záznam VSAM.

Záznamy fyzických a logických transakčních protokolů

Jedna KP VSAM je *fyzický* záznam. Informace zaprotokolované v konkrétním čase tvoří *logický* záznam s délkou, která se liší nezávisle na prostoru dostupném v KP. Takže jeden fyzický záznam může obsahovat:

- Několik logických záznamů
- Jeden nebo více logických záznamů a část jiného logického záznamu
- Pouze část jednoho logického záznamu

Termín *záznam protokolu* odkazuje na *logický* záznam, bez ohledu na to, kolik *fyzických* záznamů je potřeba k jeho uložení.

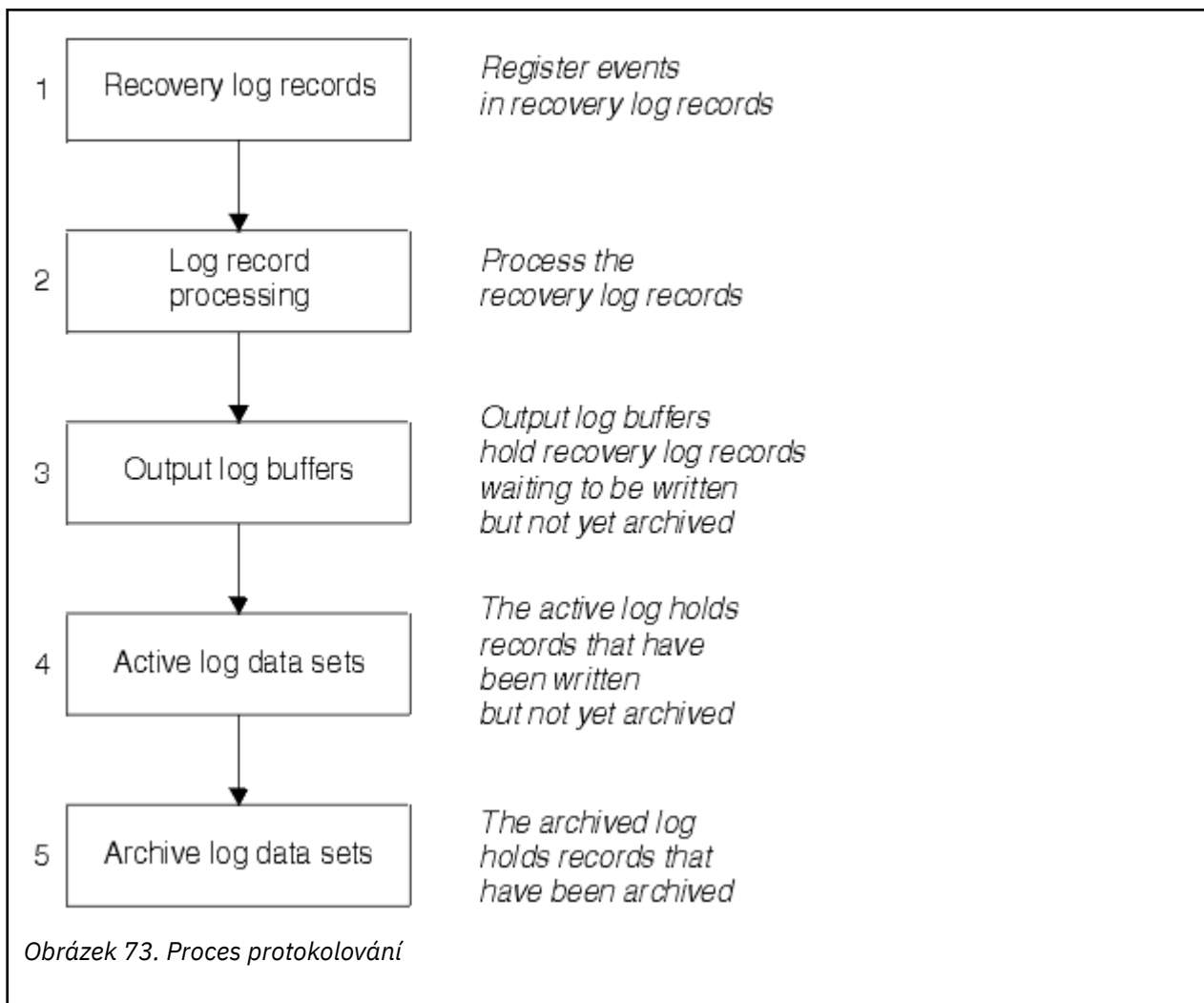
Jak se zapisují protokoly IBM MQ for z/OS

Toto téma popisuje, jak produkt IBM MQ zpracovává záznamy souboru protokolu.

Produkt IBM MQ zapíše každý záznam protokolu do datové sady DASD nazvané *aktivní protokol*. Když je aktivní protokol plný, produkt IBM MQ zkopíruje jeho obsah do datové sady DASD nebo pásky nazvané *protokol archivace*. Tento proces se nazývá *odlehčování*.

Obrázek 73 na stránce 228 ilustruje proces protokolování. Záznamy protokolu obvykle procházejí následujícím cyklem:

1. IBM MQ zaznamenává změny dat a významné události v záznamech protokolu pro zotavení.
2. Produkt IBM MQ zpracuje záznamy protokolu pro zotavení a v případě potřeby je rozdělí do segmentů.
3. Záznamy protokolu se umísťují sekvenčně do *výstupních vyrovnávacích pamětí protokolu*, které jsou formátovány jako VSAM Controls Intervaly (CI). Každý záznam protokolu je identifikován relativní bajtovou adresou v rozsahu 0 až $2^{64} - 1$.
4. KP se zapisují do sady předdefinovaných datových sad aktivního protokolu DASD, které se používají sekvenčně a recyklují.
5. Je-li archivace aktivní a každá datová sada aktivního protokolu se zaplní, dojde k automatickému přenesení jejího obsahu do nové datové sady archivního protokolu.



Při zápisu aktivního protokolu

Vyrovňovací paměti protokolu v úložišti se zapisují do datové sady aktivního protokolu vždy, když dojde k některé z následujících událostí:

- Vyrovňovací paměti protokolu se zaplní.
- Byla dosažena prahová hodnota zápisu (určená v makru CSQ6LOGP).
- Dojde k určitým významným událostem, například k bodu potvrzení nebo při zadání příkazu IBM MQ BACKUP CFSTRUCT.

Při inicializaci správce front jsou datové sady aktivního protokolu uvedené v BSDS dynamicky přidělovány pro výhradní použití správcem front a zůstávají přiděleny výhradně pro produkt IBM MQ, dokud není správce front ukončen.

Dynamické přidávání datových sad protokolu

Je možné dynamicky definovat nové datové sady aktivního protokolu za běhu správce front. Tato funkce zmírňuje problém uváznutí správce front, když archivace nemůže odlehčovat aktivní protokoly kvůli přechodnému problému. Další informace viz příkaz [DEFINE LOG](#).

Poznámka: Chcete-li předefinovat nebo odebrat aktivní protokoly, musíte ukončit a restartovat správce front.

IBM MQ a subsystém správy úložiště

Parametry systému IBM MQ vám umožňují určit úložné třídy subsystému správy úložiště (MVS/DFP SMS) při dynamickém přidělování datových sad protokolu archivu IBM MQ. Produkt IBM MQ zahájí archivaci datových sad protokolu, ale můžete použít SMS k provedení přidělení datové sady archivu.

Související odkazy

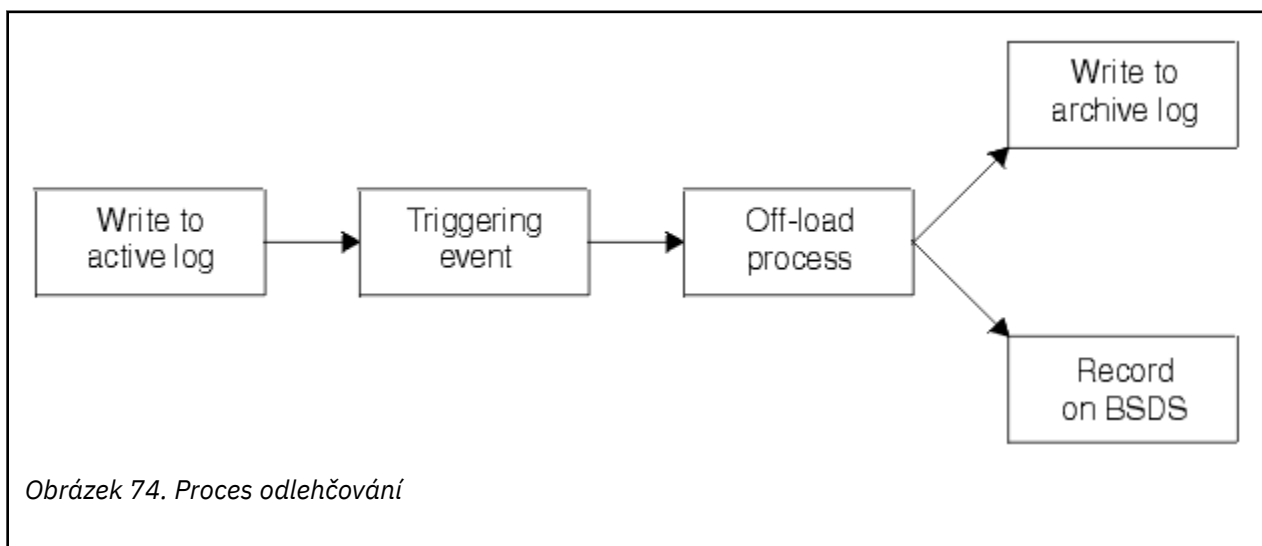
“Při zápisu protokolu archivace IBM MQ for z/OS” na stránce 229

V tomto tématu jsou uvedeny informace o procesu kopírování aktivních protokolů do archivních protokolů a o tom, kdy k tomuto procesu dojde.

Při zápisu protokolu archivace IBM MQ for z/OS

V tomto tématu jsou uvedeny informace o procesu kopírování aktivních protokolů do archivních protokolů a o tom, kdy k tomuto procesu dojde.

Proces kopírování aktivních protokolů do protokolů archivace se nazývá *odlehčování*. Vztah odlehčování k ostatním protokolovaným událostem je schematicky zobrazen v souboru [Obrázek 74 na stránce 229](#).



Spuštění procesu odlehčování

Proces odlehčování aktivního protokolu do protokolu archivace může být spuštěn několika událostmi. Příklad:

- Vyplňování datové sady aktivního protokolu.
- Použití příkazu MQSC ARCHIVE LOG.
- Došlo k chybě při zápisu do datové sady aktivního protokolu.

Datová sada je oříznuta před bodem selhání a záznam, který nebyl zapsán, se stane prvním záznamem nové datové sady. Odlehčování je spuštěno pro zkrácenou datovou sadu jako pro normální úplnou datovou sadu protokolu. Pokud existují duální aktivní protokoly, jsou obě kopie oříznuty, takže obě kopie zůstanou synchronizované.

Zpráva CSQJ110E se vydá, když je poslední dostupný aktivní protokol plný z 5% a poté se zvýší o 5%, přičemž se uvede využitá kapacita protokolu v procentech. Pokud se všechny aktivní protokoly zaplní, produkt IBM MQ zastaví zpracování, dokud nedojde k odlehčení, a vydá tuto zprávu:

```
CSQJ111A +CSQ1 OUT OF SPACE IN ACTIVE LOG DATA SETS
```

Proces odlehčování

Po zaplnění všech aktivních protokolů produkt IBM MQ spustí proces odlehčování a zastaví zpracování, dokud nebude proces odlehčování dokončen. Pokud zpracování odlehčování selže při zaplnění aktivních protokolů, IBM MQ se ukončí.

Je-li aktivní protokol připraven k odlehčení, odešle se operátorovi konzoly z/OS požadavek na připojení pásky nebo přípravu jednotky DASD. Hodnota volby protokolování ARCWTOR (další informace viz Použití CSQ6ARVP). určuje, zda je požadavek přijat. Pokud pro odlehčování používáte pásku, zadejte ARCWTOR=YES. Pokud je hodnota YES, požadavku předchází WTOR (číslo zprávy CSQJ008E) sdělování operátorovi, aby připravil datovou sadu protokolu archivace, která má být přidělena.

Operátor nemusí na tuto zprávu reagovat okamžitě. Zpoždění odezvy však zpožďuje proces odlehčování. Neovlivňuje výkon systému IBM MQ, pokud operátor nepozdrží odezvu tak dlouho, že IBM MQ vyčerpá aktivní protokoly.

Operátor může reagovat zrušením procesu odlehčování. V tomto případě, pokud je alokace určena pro první kopii duálních archivních datových sad, je proces odlehčování pouze odložen do doby, kdy se další datová sada aktivního protokolu zaplní. Pokud je alokace pro druhou kopii, proces archivace se přepne do režimu jedné kopie, ale pouze pro tuto datovou sadu.

Přerušení a chyby při odlehčování

Požadavek na zastavení správce front se neprojeví, dokud nebude zpracování odlehčování dokončeno. Dojde-li k selhání produktu IBM MQ během probíhajícího odlehčování, bude odlehčování zahájeno znovu po restartování správce front.

Zprávy během zpracování odlehčování

Odložené zprávy jsou odesílány do konzoly z/OS produktem IBM MQ a procesem odlehčování. Tyto zprávy můžete použít k vyhledání rozsahů RBA v různých datových sadách protokolu.

Větší relativní bajtová adresa protokolu

Tato funkce zlepšuje dostupnost správce front tím, že prodlužuje dobu potřebnou k resetování protokolu.


Data pro zotavení jsou zapsána do protokolu, aby byly při restartování správce front k dispozici trvalé zprávy. Termín log Relative Byte Address (log RBA) se používá k odkazování na umístění dat jako na odsazení od začátku protokolu.

Před IBM MQ 8.0 může 6bajtový protokol RBA adresovat až 256 terabajtů dat. Před zapsáním tohoto množství záznamů protokolu je nutné resetovat protokol správce front podle postupu popsaného v části [Resetování protokolu správce front](#).

Resetování protokolů správců front není rychlý proces a může vyžadovat rozšířený výpadek kvůli nutnosti resetovat sady stránek jako součást procesu. V případě správce front s vysokým využitím může být tato operace obvykle prováděna jednou za rok.

V systému IBM MQ 8.0 může být adresa RBA protokolu dlouhá 8 bajtů a správce front může nyní adresovat více než 64 000 krát více dat (16 exabajtů), než bude nutné resetovat adresu RBA protokolu. Dopad použití větší adresy RBA protokolu je ten, že velikost zapsaných dat protokolu se zvýší o několik bajtů.

Kdy je tato funkce povolena?

 Správci front vytvoření v produktu IBM MQ 9.3.0 nebo novějším již tuto funkci mají povolenou.

Pokud se aktuální protokol RBA blíží ke konci rozsahu RBA protokolu, zvažte převod správce front na použití 8bajtového protokolu RBA namísto resetování protokolu správce front. Převod správce front tak, aby používal 8bajtové protokoly RBAs, vyžaduje kratší výpadek, než je resetování protokolu, a výrazně prodlužuje dobu potřebnou k resetování protokolu.

Zpráva CSQJ034I, vydaná během inicializace správce front, označuje konec rozsahu RBA protokolu pro správce front, jak je konfigurován, a lze ji použít k určení, zda se používají 6bajtové nebo 8bajtové adaptéry RBA protokolu.

Jak je tato funkce povolena?

8bajtový protokol RBA je povolen spuštěním správce front s BSDS ve formátu verze 2. V souhrnu je toho dosaženo:

1. Zajištění, že všichni správci front ve skupině sdílení front splňují požadavky na povolení 8bajtového protokolu RBA
2. Čisté ukončení práce správce front
3. Spuštěním obslužného programu pro převod BSDS vytvoříte kopii BSDS ve formátu verze 2.
4. Restartování správce front s převedeným BSDS.

Jakmile byl správce front převeden na použití 8bajtového protokolu RBA, nemůže se vrátit k použití 6bajtového protokolu RBA.

Podrobný postup pro povolení 8bajtových protokolů RBA naleznete v tématu Implementace větší relativní bajtové adresy protokolu.

Související úlohy

Plánování zvýšení maximálního rozsahu adresovatelných protokolů

Související odkazy

Obslužný program pro převod BSDS (CSQJUCNV)

Datová sada zaváděcího programu

Datová sada samozavedení je vyžadována produktem IBM MQ jako mechanismus pro odkazování na datové sady protokolu a záznamy protokolu. Tyto informace jsou vyžadovány během normálního zpracování a obnovy po restartu.

Pro co je datová sada samozavedení

Datová sada zaváděcího programu (BSDS) je datová sada s posloupností klíčů VSAM (KSDS), která obsahuje informace potřebné pro IBM MQ. Obsahuje následující:

- Soupis všech aktivních a archivovaných datových sad protokolu známých produktu IBM MQ. Produkt IBM MQ používá tento inventář k:
 - Sledování aktivních a archivovaných datových sad protokolu
 - Vyhledat záznamy protokolu, aby mohly uspokojit požadavky na čtení protokolu během normálního zpracování
 - Vyhledat záznamy protokolu, aby mohl zpracovat zpracování restartu
- Produkt IBM MQ ukládá informace do inventáře pokaždé, když je definována datová sada protokolu archivace nebo když je znovu použita datová sada aktivního protokolu. Pro aktivní protokoly inventář zobrazuje, které jsou plné a které jsou k dispozici pro opětovné použití. Soupis obsahuje relativní bajtovou adresu (RBA) každé části protokolu uchovávaného v dané datové sadě.
- *Obalený* inventář všech nedávných IBM MQ aktivit. To je zapotřebí, pokud musíte restartovat správce front.

Služba BSDS je vyžadována v případě, že došlo k chybě správce front a je nutné ji restartovat. IBM MQ **musí** mít BSDS. Chcete-li minimalizovat pravděpodobnost problémů během restartu, můžete nakonfigurovat produkt IBM MQ pomocí duálních systémů BSDS, z nichž každý zaznamenává stejné informace. Použití duálních systémů BSDS je známé jako spuštění v *duálním režimu*. Je-li to možné, umístěte kopie na samostatné svazky. To snižuje riziko jejich ztráty, pokud je svazek poškozen nebo zničen. Použijte duální BSDS spíše než duální zápis do DASD.

BSDS je nastaven, když je produkt IBM MQ upraven, a můžete spravovat inventář pomocí obslužného programu inventáře protokolu změn (CSQJU003). Další informace o tomto obslužném programu viz [Administrace IBM MQ for z/OS](#). Odkazuje na něj příkaz DD v proceduře spuštění správce front.

Obvykle produkt IBM MQ uchovává duplicitní kopie BSDS. Pokud dojde k chybě I/O, dealokuje selhávající kopii a pokračuje s jedním BSDS. Můžete obnovit operaci v duálním režimu, to je popsáno v tématu [Administrace IBM MQ for z/OS](#).

Aktivní protokoly jsou nejprve registrovány v BSDS, když je nainstalován produkt IBM MQ . Aktivní protokoly nelze nahradit bez ukončení a restartování správce front.

Datové sady protokolu archivace jsou přiděleny dynamicky. Když je přidělena, je název datové sady registrován v BSDS. Seznam datových sad protokolu archivu se rozbálí při přidávání archivů a zalomí se po dosažení uživatelem určeného počtu položek. Maximální počet záznamů je 1000 pro jedno archivní protokolování a 2000 pro duální protokolování.

K odstranění datových sad protokolu archivace můžete použít systém správy pásek (produkt IBM MQ nemá automatizovanou metodu). Proto informace o datové sadě protokolu archivace mohou být v BSDS dlouho poté, co byla datová sada protokolu archivace odstraněna administrátorem systému.

Naopak maximální počet datových sad protokolu archivu mohl být překročen a data z BSDS zrušená dlouho před dosažením data vypršení platnosti datové sady.

Pomocí následujícího příkazu MQSC můžete určit rozsah protokolu a název datové sady aktivního nebo archivního protokolu, která obsahuje nejstarší protokol RBA, nezbytný pro různé typy zotavení média nebo správce front:

```
DISPLAY USAGE TYPE(DATASET)
```

Pokud modul systémových parametrů určuje, že datové sady protokolu archivace jsou při přidělení katalogizovány, odkazuje BSDS na katalog ICF (integrated catalog facility) pro informace potřebné pro pozdější alokace. Jinak položky BSDS pro každý svazek registrují sériové číslo svazku a informace o jednotce, které jsou potřebné pro pozdější přidělení.

Verze BSDS

Formát BSDS se liší podle jeho verze. Zvýšení verze BSDS umožňuje použití nových funkcí. Následující verze BSDS jsou podporovány produktem IBM MQ:

Verze 1

Podporováno všemi verzemi produktu IBM MQ. BSDS verze 1 podporuje 6bajtové hodnoty RBA protokolu.

verze 2

Podporováno produktem IBM MQ 8.0 a novějším. BSDS verze 2 povoluje 8bajtové hodnoty RBA protokolu a až 310 datových sad v každé aktivní kopii protokolu.

V 9.3.0 Standardně povoleno pro správce front vytvořené v produktu IBM MQ 9.3.0 a novějším.

verze 3

Podporováno produktem IBM MQ 8.0 a novějším. BSDS je automaticky převeden na verzi 3, z verze 2, když je do jedné z kopií aktivního protokolu přidáno více než 31 datových sad.

Verzi BSDS můžete určit spuštěním obslužného programu mapy tiskového protokolu (CSQJU004). Chcete-li převést BSDS z verze 1 na verzi 2, spusťte obslužný program pro převod BSDS (CSQJUCNV).

Další informace o 6bajtových a 8bajtových protokolech RBA naleznete v části [“Větší relativní bajtová adresa protokolu”](#) na stránce 230 .

Archivovat datové sady protokolu a kopie BSDS

Při každém vytvoření nové datové sady protokolu archivace se také vytvoří kopie BSDS. Pokud je protokol archivace na pásce, je BSDS první datovou sadou na prvním výstupním nosiči. Pokud je protokol archivace na serveru DASD, je BSDS oddělenou datovou sadou.

Názvy datových sad protokolu archivace a kopie BSDS jsou stejné, kromě toho, že kvalifikátor nejnižší úrovně názvu protokolu archivace začíná A a kopie BSDS začíná B, například:

Název protokolu archivace

CSQ.ARCHLOG1.E00186.T2336229. A 0000001

Název kopie BSDS

CSQ.ARCHLOG1.E00186.T2336229. B 0000001

Dojde-li při kopírování BSDS k chybě čtení, kopie se nevytvoří, vydá se zpráva [CSQJ125E](#) a odložení do nové datové sady protokolu archivu pokračuje bez kopie BSDS.

z/OS

Definice systému v systému z/OS

Produkt IBM MQ for z/OS používá mnoho výchozích definic objektů a poskytuje ukázkový soubor JCL k vytvoření těchto výchozích objektů. Toto téma slouží k pochopení těchto výchozích objektů a ukázkového kódu JCL.

Nastavení parametrů systému

V systému IBM MQ for z/OS modul systémových parametrů řídí prostředí protokolování, archivace, trasování a připojení, která produkt IBM MQ používá ve svém provozu. Parametry systému jsou určeny třemi makry assembleru:

CSQ6SYSP

Systémové parametry, včetně nastavení prostředí připojení a trasování.

CSQ6LOGP

Parametry protokolování.

CSQ6ARVP

Parametry archivace protokolu.

Výchozí moduly parametrů jsou dodávány s produktem IBM MQ for z/OS. Pokud tyto hodnoty neobsahují hodnoty, které chcete použít, můžete vytvořit vlastní moduly parametrů pomocí ukázky dodané s produktem IBM MQ. Ukázka je `thlqual.SCSQPROC (CSQ4ZPRM)`.

V době, kdy je spuštěn správce front, můžete změnit některé parametry systému. Viz příkazy SET SYSTEM, SET LOG a SET ARCHIVE v části [Příkazy MQSC](#).

Další informace o definování naleznete v následujících tématech:

- [“Definování systémových objektů pro IBM MQ for z/OS” na stránce 234](#)
- [“Vyladění správce front v systému IBM MQ for z/OS” na stránce 238](#)
- [“Ukázkové definice dodávané s produktem IBM MQ for z/OS” na stránce 239](#)

Související pojmy

[Upravit ukázkové vstupní datové sady inicializace](#)

[Zdroje, ze kterých můžete zadat příkazy MQSC a PCF na systému IBM MQ for z/OS](#)

Související úlohy

[Správa serveru z/OS](#)

[Konfigurace klastrů](#)

[Monitorování IBM MQ](#)

Produkt IBM MQ for z/OS vyžaduje další předdefinované objekty pro aplikace publikování/odběru, klastr, řízení kanálu a další funkce administrace systému.

Systémové objekty požadované produktem IBM MQ for z/OS lze rozdělit do následujících kategorií:

- [Publikovat/odebírat objekty](#)
- [Výchozí systémové objekty](#)
- [Objekty systémového příkazu](#)
- [Objekty administrace systému](#)
- [Fronty kanálů](#)
- [Fronty klastru](#)
- [Fronty skupin sdílení front](#)
- [Paměťové třídy](#)
- [Definování fronty nedoručených zpráv systémového objektu](#)
- [Výchozí přenosová fronta](#)
- [Interní fronty](#)
- [“Fronta ověření kanálu” na stránce 238](#)

Objekty publikování/odběru

Před použitím aplikací publikování/odběru s produktem IBM MQ for z/OS je třeba definovat několik systémových objektů. Ukázkové definice jsou dodávány s produktem IBM MQ, které vám pomohou tyto objekty definovat. Tyto ukázky jsou popsány v části [CSQ4INSG](#).

Chcete-li použít publikování/odběr, musíte definovat následující objekty:

- Lokální fronta s názvem SYSTEM.RETAINED.PUB.QUEUE, která se používá k uchování kopie každého zachovaného publikování ve správci front. Každý úplný název tématu může mít v této frontě uloženo až jedno zachované publikování. Pokud vaše aplikace budou využívat zachovaná publikování v mnoha různých tématech nebo pokud jsou zachované zprávy publikování velkými zprávami, měly by být požadavky na úložiště pro tuto frontu pečlivě naplánovány, včetně jejich přiřazení k vlastní sadě stránek, pokud jsou požadavky na úložiště pro tuto frontu velké. Chcete-li zlepšit výkon, měli byste definovat tuto frontu s typem indexu MSGID (jak je zobrazeno v dodané definici ukázkové fronty).
- Lokální fronta s názvem SYSTEM.DURABLE.SUBSCRIBER.QUEUE, která se používá k uchování trvalé kopie trvalých odběrů ve správci front. Chcete-li zlepšit výkon, měli byste tuto frontu definovat s typem indexu CORRELID (jak je zobrazeno v dodané definici ukázkové fronty).
- Lokální fronta s názvem SYSTEM.DURABLE.MODEL.QUEUE, která se používá jako model pro spravované trvalé odběry.
- Lokální fronta s názvem SYSTEM.NDURABLE.MODEL.QUEUE, která se používá jako model pro spravované dočasné odběry.
- Seznam názvů s názvem SYSTEM.QPUBSUB.QUEUE.NAMELIST, který obsahuje seznam názvů front monitorovaných rozhraním pro publikování/odběr ve frontě.
- Seznam názvů s názvem SYSTEM.QPUBSUB.SUBPOINT.NAMELIST, který obsahuje seznam objektů témat používaných rozhraním pro publikování/odběr ve frontě k porovnání objektů témat s body odběru.
- Téma s názvem SYSTEM.BASE.TOPIC, který se používá jako základní téma pro řešení atributů.
- Téma s názvem SYSTEM.BROKER.DEFAULT.STREAM, což je výchozí proud používaný rozhraním publikování/odběru zařazeným ve frontě.
- Téma s názvem SYSTEM.BROKER.DEFAULT.SUBPOINT, což je výchozí bod odběru RFH2 používaný rozhraním pro publikování/odběr ve frontě.

- Téma s názvem SYSTEM.BROKER.ADMIN.STREAM, což je administrativní proud používaný rozhraním publikování/odběru zařazeným ve frontě.
- Odběr s názvem SYSTEM.DEFAULT.SUB, což je výchozí objekt odběru používaný k zadání výchozích hodnot v příkazech DEFINE SUB.

Předvolené objekty systému

Předvolené systémové objekty se používají k poskytnutí předvolených atributů, když definujete objekt, a neuvádějí název jiného objektu, na kterém se má založit definice.

Názvy výchozích definic systémových objektů začínají znaky "SYSTEM.DEFAULT"nebo"SYSTEM.DEF. " Například výchozí lokální fronta systému má název SYSTEM.DEFAULT.LOCAL.QUEUE.

Tyto objekty definují předvolby systému pro atributy těchto objektů IBM MQ :

- Lokální fronty
- Modelové fronty
- Alias fronty
- Vzdálené fronty
- Procesy
- Seznamy názvů
- Kanály
- Paměťové třídy
- Ověřovací informace

Sdílené fronty jsou speciálním typem lokální fronty, takže když definujete sdílenou frontu, je definice založena na SYSTEM.DEFAULT.LOCAL.QUEUE. Musíte si pamatovat, že jste zadali hodnotu pro název struktury prostředku Coupling Facility, protože jeden z nich není uveden ve výchozí definici. Případně můžete definovat vlastní výchozí definici sdílené fronty, která se má použít jako základ pro sdílené fronty, aby všechny zdědily požadované atributy. Nezapomeňte, že je třeba definovat sdílenou frontu pouze v jednom správci front ve skupině sdílení front.

Objekty systémových příkazů

Názvy objektů systémových příkazů začínají znaky SYSTEM.COMMAND. Tyto objekty musíte definovat dříve, než budete moci použít operace IBM MQ a ovládací panely k vydání příkazů pro subsystém IBM MQ .

Existují dva objekty systémového příkazu:

1. Vstupní fronta systémových příkazů je lokální fronta, do které jsou příkazy vloženy před jejich zpracováním příkazovým procesorem IBM MQ . Musí mít název SYSTEM.COMMAND.INPUT. Alias s názvem SYSTEM.ADMIN.COMMAND.QUEUE by měla být také definována, pro kompatibilitu s IBM MQ for Multiplatformsa pro použití v IBM MQ Console a administrative REST API.
2. SYSTEM.COMMAND.REPLY.MODEL je modelová fronta, která definuje frontu pro odpověď systémového příkazu.

Existují dva další objekty pro použití v produktu IBM MQ Explorer:

- SYSTEM.MQEXPLORER.REPLY.MODEL
- SYSTEM.ADMIN.SVRCONN

SYSTEM.REST.REPLY.QUEUE je fronta odpovědi používaná produktem IBM MQ administrative REST API.

Příkazy se obvykle odesílají pomocí přechodných zpráv, takže oba objekty systémových příkazů by měly mít atribut DEFPSIST (NO), aby aplikace, které je používají (včetně dodaných aplikací, jako je obslužný program a operace a ovládací panely), standardně získávaly přechodné zprávy. Máte-li aplikaci, která pro

příkazy používá trvalé zprávy, nastavte atribut DEFTYPE (PERMDYN) pro frontu odpovědí, protože zprávy odpovědí pro tyto příkazy jsou trvalé.

Objekty administrace systému

Názvy objektů administrace systému začínají znaky SYSTEM.ADMIN.

Existuje sedm objektů administrace systému:

- Systém SYSTEM.ADMIN.CHANNEL.EVENT fronta
- Systém SYSTEM.ADMIN.COMMAND.EVENT fronta
- Systém SYSTEM.ADMIN.CONFIG.EVENT fronta
- Systém SYSTEM.ADMIN.PERFM.EVENT fronta
- Systém SYSTEM.ADMIN.QMGR.EVENT fronta
- Systém SYSTEM.ADMIN.TRACE.ROUTE.QUEUE fronta
- Systém SYSTEM.ADMIN.ACTIVITY.QUEUE fronta

Fronty kanálů

Chcete-li používat distribuované řazení do front, je třeba definovat následující objekty:

- Lokální fronta s názvem SYSTEM.CHANNEL.SYNCQ, který se používá k udržování pořadových čísel a logických identifikátorů pracovních jednotek (LUWID) kanálů. Chcete-li zlepšit výkon kanálu, měli byste definovat tuto frontu s typem indexu MSGID (jak je zobrazeno v dodané definici ukázkové fronty).
- Lokální fronta s názvem SYSTEM.CHANNEL.INITQ, který se používá pro příkazy kanálu.

Tyto fronty nelze definovat jako sdílené fronty.

Fronty klastru

Chcete-li používat klastry IBM MQ, musíte definovat následující objekty:

- Lokální fronta s názvem SYSTEM.CLUSTER.COMMAND.QUEUE, která se používá ke komunikaci změn úložiště mezi správci front. Zprávy zapsané do této fronty obsahují aktualizace dat úložiště, které mají být použity na lokální kopii úložiště, nebo požadavky na data úložiště.
- Lokální fronta s názvem SYSTEM.CLUSTER.REPOSITORY.QUEUE, která se používá k uchování trvalé kopie úložiště.
- Lokální fronta s názvem SYSTEM.CLUSTER.TRANSMIT.QUEUE, což je přenosová fronta pro všechny cíle v klastru. Z důvodů výkonu byste měli tuto frontu definovat s typem indexu CORRELID (jak je uvedeno v definici ukázkové fronty).

Tyto fronty obvykle obsahují velký počet zpráv.

Tyto fronty nelze definovat jako sdílené fronty.

Fronty skupin sdílení front

Chcete-li používat sdílené kanály a řazení do front v rámci skupiny, je třeba definovat následující objekty:

- Sdílená fronta s názvem SYSTEM.QSG.CHANNEL.SYNCQ, který se používá k uchování informací o synchronizaci pro sdílené kanály.
- Sdílená fronta s názvem SYSTEM.QSG.TRANSMIT.QUEUE, která se používá jako přenosová fronta pro řazení do front v rámci skupiny. Pokud pracujete ve skupině sdílení front, musíte tuto frontu definovat, a to i v případě, že nepoužíváte řazení do front v rámci skupiny.

Paměťové třídy

Doporučuje se definovat následujících šest paměťových tříd. Musíte definovat čtyři z nich, protože jsou vyžadovány produktem IBM MQ. Ostatní definice paměťových tříd se doporučují, protože se používají v definicích ukázkové fronty.

VÝCHOZÍ (povinné)

Tato paměťová třída se používá pro všechny fronty zpráv, které nejsou kritické pro výkon a nevejdou se do žádné jiné paměťové třídy. Jedná se také o dodanou výchozí paměťovou třídu, pokud ji neuvedete při definování fronty.

NODEFINE (povinné)

Tato paměťová třída se používá, pokud paměťová třída uvedená při definování fronty není definována.

REMOTE (vyžadováno)

Tato paměťová třída se používá primárně pro přenosové fronty, tj. pro fronty související se systémem s krátkodobými zprávami kritickými pro výkon.

SYSLNGLV

Tato paměťová třída se používá pro zprávy s dlouhou životností a kritickým výkonem.

SYSTEM (vyžadováno)

Tato paměťová třída se používá pro kritické fronty zpráv související se systémem, například SYSTEM.CHANNEL.SYNQ a SYSTEM.CLUSTER.* fronty.

SYSVOLAT

Tato paměťová třída se používá pro krátkodobé zprávy kritické z hlediska výkonu.

Podle potřeby můžete upravit jejich atributy a přidat další definice paměťových tříd.

Definování fronty nedoručených zpráv systémového objektu

Fronta nedoručených zpráv se použije, pokud není místo určení zprávy platné. Produkt IBM MQ tyto zprávy vkládá do lokální fronty s názvem fronta nedoručených zpráv. Ačkoli není fronta nedoručených zpráv povinná, měli byste ji považovat za nezbytnou, zejména pokud používáte distribuované fronty nebo jeden z mostů IBM MQ .

Nedefinujte frontu nedoručených zpráv jako sdílenou frontu. Vložení do lokální fronty v jednom správci front může být vložena do fronty nedoručených zpráv. Pokud byla fronta nedoručených zpráv sdílenou frontou, mohla by obslužná rutina fronty nedoručených zpráv v jiném systému zpracovat zprávu a vložit ji do fronty se stejným názvem, ale protože se nachází v jiném správci front, byla by to nesprávná fronta nebo by měla jiný profil zabezpečení. Pokud by fronta neexistovala, nepodařilo by se ji znovu zpracovat.

Pokud se rozhodnete definovat frontu nedoručených zpráv, musíte také sdělit správci front její název. K tomu použijte příkaz ALTER QMGR DEADQ (*název_fronty*). Další informace naleznete v tématu [Zobrazení a změna atributů správce front](#).

Výchozí přenosová fronta

Výchozí přenosová fronta se používá v případě, že pro odesílání zpráv jinému správci front není k dispozici žádná jiná vhodná přenosová fronta. Pokud definujete výchozí přenosovou frontu, je nutné definovat také kanál, který bude tato fronta využívat. Pokud tak neučiníte, zprávy umístěné do výchozí přenosové fronty nebudou přeneseny do vzdáleného správce front a zůstanou ve frontě.

Pokud se rozhodnete definovat výchozí přenosovou frontu, musíte také sdělit správci front její název. Chcete-li to provést, použijte příkaz ALTER QMGR.

Vnitřní fronty

- **Nevyřízená datová fronta**

- Fronta definovaná pro interní použití, SYSTEM.PENDING.DATA.QUEUE, podporuje použití trvalých odběrů v prostředí JMS publikování/odběru.
- **JMS 2.0 fronta fázování prodlevy doručení**
 - Je-li použita funkčnost prodlevy doručení poskytovaná produktem JMS 2.0, je použita interní fronta fázování SYSTEM.DDELAY.LOCAL.QUEUE, musí být definována. Tuto frontu používá správce front k dočasnému uložení zpráv odeslaných s nenulovou prodlevou doručení až do dokončení prodlevy doručení a vložení zprávy do cílového místa určení. V části CSQ4INSG je k dispozici ukázková definice pro tuto frontu, která je označena jako komentář.
 - Při definování SYSTEM.DDELAY.LOCAL.QUEUE, musíte nastavit atributy STGCLASS, MAXMSGL a MAXDEPTH pro očekávaný počet zpráv, které budou odeslány s prodlevou doručení. Dále při definování SYSTEM.DDELAY.LOCAL.QUEUE zajišťuje, že do této fronty mohou vkládat zprávy pouze správce front. Je třeba dbát na to, aby žádný identifikátor uživatele neměl oprávnění vkládat zprávy do této fronty.

Fronta ověření kanálu

Pro interní použití ověření kanálu použijte SYSTEM.CHLAUTH.DATA.QUEUE. Ukázkové definice jsou dodávány s produktem IBM MQ, které vám pomohou tyto objekty definovat. Tato ukázka je popsána v části CSQ4INSA, která také definuje některá výchozí pravidla.

Vyladění správce front v systému IBM MQ for z/OS

Existuje několik jednoduchých kroků, které můžete provést, abyste se ujistili, že je váš správce front vyladěn, abyste se vyhnuli základním problémům s výkonem.

Existuje řada způsobů, jak zlepšit výkon správce front, které jsou řízeny atributy správce front nastavenými příkazem ALTER QMGR. Tato část obsahuje informace o tom, jak to lze provést nastavením maximálního počtu zpráv povolených ve správci front nebo provedením 'úklidu' ve správci front. IBM MQ SupportPac [MP16 - IBM MQ pro z/OS Plánování kapacity & ladění](#) poskytuje další informace o výkonu a ladění.

Synchronizační body

Jednou z rolí správce front je řízení synchronizačního bodu v rámci aplikace. Aplikace vytvoří pracovní jednotku obsahující libovolný počet volání MQPUT nebo MQGET ukončených voláním MQCMIT.

S rostoucím počtem volání MQPUT nebo MQGET v rozsahu jednoho MQCMIT se výrazně zvyšují náklady na výkon potvrzení. Aplikace by obecně měly být navrženy tak, aby v jednom synchronizačním bodu nezobrazovaly MQPUT/MQGET velký počet zpráv.

Pomocí atributu správce front MAXUMSGS můžete administrativně omezit počet zpráv v rámci jednoho synchronizačního bodu. Pokud aplikace tento limit překročí, obdrží ve volání MQPUT, MQPUT1 nebo MQGET hodnotu MQRC_SYNCPOINT_LIMIT_DOSAŽENÉ, která tento limit překračuje. Aplikace by pak měla podle potřeby vydat příkaz MQCMIT nebo MQBACK.

Výchozí hodnota MAXUMSGS je 10000. Tuto hodnotu lze snížit, chcete-li vynutit dolní limit, což může také pomoci chránit před cyklováním aplikací. Před snížením hodnoty MAXUMSGS se ujistěte, že rozumíte existujícím aplikacím, abyste se ujistili, že nepřekračují limit, nebo že můžete tolerovat návratový kód MQRC_SYNCPOINT_LIMIT_DOSAŽENÉ

Vypršelé zprávy

Zprávy, jejichž platnost vypršela, jsou zahozeny dalším příslušným voláním MQGET. Pokud však nedojde k žádnému takovému volání, nebudou zprávy s vypršenou platností vyřazeny a pro některé fronty,

zejména pro ty, kde načítání zpráv provádí MessageId, CorrelId nebo GroupId a fronta je indexována pro výkon, se může hromadit mnoho zpráv s vypršenou platností. Správce front může pravidelně skenovat zprávy s vypršenou platností, které jsou poté odstraněny. Můžete si vybrat, jak často toto skenování probíhá, pokud vůbec. Existují dva způsoby, jak to udělat:

Explicitní požadavek

Můžete řídit, které fronty jsou skenovány a kdy. Zadejte příkaz REFRESH QMGR TYPE (EXPIRAČNÍ) s uvedením fronty nebo front, které chcete skenovat.

Periodické skenování

Pomocí atributu EXPRYINT můžete určit interval vypršení platnosti v objektu správce front. Správce front udržuje informace o zprávách s vypršenou platností v jednotlivých frontách a ví, v jakém čase má skenování zpráv s vypršenou platností smysl. Při každém dosažení intervalu EXPRYINT hledá správce front kandidátské fronty, které stojí za to skenovat zprávy s vypršenou platností, a skenuje pouze ty fronty, které považuje za užitečné. Neskenuje všechny fronty. Tím se vyhnete plýtvání časem procesoru při zbytečných skenováních.

Sdílené fronty jsou skenovány pouze jedním správcem front ve skupině sdílení front. Obecně platí, že první správce front, který se restartuje, nebo první, který má sadu EXPRYINT, provede skenování.

Poznámka: Je třeba nastavit stejnou hodnotu EXPRYINT pro všechny správce front v rámci skupiny sdílení front.

Ukázkové definice dodávané s produktem IBM MQ for z/OS

Toto téma použijte jako odkaz pro ukázkou JCL a kód dodaný s produktem IBM MQ for z/OS.

Následující ukázkové definice jsou dodávány s produktem IBM MQ v knihovně thlqual.SCSQPROC. Můžete je použít k definování systémových objektů a k přizpůsobení vlastních objektů. Některé z nich můžete zahrnout do vstupních datových sad inicializace (popsáno v části [Příkazy inicializace](#)).

<i>Tabulka 22. IBM MQ ukázkové definice pro systémové objekty</i>	
vstupní datová sada inicializace	Vzorový název
CSQINP1	CSQ4INP1 CSQ4INPR
CSQINP2	CSQ4INSA CSQ4INYS ¹ CSQ4INSX CSQ4INSG CSQ4INSR CSQ4INSS CSQ4INSJ CSQ4INSM CSQ4INYG CSQ4INYR CSQ4INYC CSQ4INYD CSQ4INSC
CSQINPT	CSQ4INST CSQ4INYT

Tabulka 22. IBM MQ ukázkové definice pro systémové objekty (pokračování)

vstupní datová sada inicializace	Vzorový název
Jiná	CSQ4DISP CSQ4INPX CSQ4IVPQ CSQ4IVPG CSQ4MSTR CSQ4MSRR CSQ4QMIN

Poznámka:

1. Pořadí těchto ukázkových definic je důležité: pokud jsou INYS, INSX a INSG seřazeny nesprávně, dojde k chybě.

Ukázky CSQINP1

Použijte ukázkovou datovou sadu CSQINP1 `thlqual.SCSQPROC(CSQ4INP1)`, když používáte jednu sadu stránek pro každou třídu zprávy, nebo `thlqual.SCSQPROC(CSQ4INPR)`, když používáte více sad stránek pro hlavní třídy zpráv. Obsahuje definice fondů vyrovnávacích pamětí, sadu stránek pro přidružení fondů vyrovnávacích pamětí a příkaz ALTER SECURITY. Zahrňte ukázkou do zřetězení CSQINP1 vaší procedury spuštěné úlohy správce front.

Ukázky CSQINP2

Ukázka systémového objektu CSQ4INSG

Ukázková datová sada CSQINP2 `thlqual.SCSQPROC(CSQ4INSG)` obsahuje definice pro následující systémové objekty pro obecné použití:

- Předvolené objekty systému
- Objekty systémových příkazů
- Objekty administrace systému
- Ostatní objekty pro použití v systému

Musíte definovat objekty v této ukázce, ale musíte to provést pouze jednou, když je subsystém poprvé spuštěn. Nejlepším způsobem, jak to provést, je zahrnout definice do datové sady CSQINP2. Jsou udržovány při ukončení práce a restartování správce front. Názvy objektů nesmíte měnit, ale v případě potřeby můžete změnit jejich atributy.

Jsou-li splněny následující podmínky, je do systému SYSTEM.DURABLE.SUBSCRIBER.QUEUE (i v případě, že odběr publikování není aktivní):

- Atribut PSMODE správce front je nastaven na hodnotu DISABLED.
- Je přítomen ukázkový objekt CSQ4INST příkaz `DEFINE SUB('SYSTEM.DEFAULT.SUB')`.

Chcete-li se tomu vyhnout, odstraňte příkaz `DEFINE SUB('SYSTEM.DEFAULT.SUB')` nebo jej označte jako komentář.

Fronta fázování prodlevy doručení JMS 2.0, SYSTEM.DDELAY.LOCAL.QUEUE je třeba definovat pouze v případě, že je použita prodleva doručení JMS 2.0. Standardně je definice fronty označena jako komentář, což v případě potřeby můžete zrušit komentář.

CSQ4INSA systémový objekt a ukázka ověření

Ukázková datová sada CSQINP2 thlqual.SCSQPROC(CSQ4INSA) obsahuje definici fronty systému ověřování kanálu. Tato fronta uchovává záznamy ověřování kanálu. Obsahuje také výchozí pravidla ověřování kanálu.

Musíte definovat objekty v této ukázce, pokud má CHLAUTH ve správci front hodnotu ENABLED a chcete spustit kanály, nebo chcete-li nastavit nebo zobrazit záznam CHLAUTH. Je třeba je definovat pouze jednou při prvním spuštění subsystému. Nejlepším způsobem, jak to provést, je zahrnout definice do datové sady CSQINP2 . Jsou udržovány v rámci ukončení a restartování správce front. Název fronty nelze změnit.

Ukázka systémového objektu CSQ4INSS

Pokud používáte skupiny sdílení front, můžete definovat další systémové objekty.

Vzorová datová sada thlqual.SCSQPROC(CSQ4INSS) obsahuje vzorové příkazy pro použití se strukturami prostředku CF a sadu definic pro systémové objekty vyžadované pro sdílené kanály a řazení do front v rámci skupiny.

Tuto ukázku nelze použít tak, jak je. Před použitím ji musíte upravit. Poté můžete tohoto člena zahrnout do zřetězení definic dat CSQINP2 spouštěcí procedury správce front nebo jej můžete použít jako vstup do funkce COMMAND obslužného programu CSQUTIL k zadání požadovaných příkazů.

Při definování skupin nebo sdílených objektů je třeba je zahrnout do zřetězení definic dat CSQINP2 pouze pro jednoho správce front ve skupině sdílení front.

Ukázka systémového objektu CSQ4INSX

Pokud používáte distribuované řazení do front a klastrování, musíte definovat další systémové objekty.

Ukázková datová sada thlqual.SCSQPROC(CSQ4INSX) obsahuje požadované definice front. Tohoto člena můžete zahrnout do zřetězení definic dat CSQINP2 spouštěcí procedury správce front, nebo jej můžete použít jako vstup pro funkci COMMAND v obslužném programu CSQUTIL a zadat požadované příkazy DEFINE.

Existují dva typy definic objektů:

- SYSTEM.CHANNEL.xx, potřebný pro všechny distribuované fronty
- SYSTEM.CLUSTER.xx, potřebný pro klastrování

Ukázka objektu CSQ4INSJ system JMS

Definuje fronty používané v doméně publikování/odběru JMS .

Ukázka systémového objektu CSQ4INSM

Používáte-li rozšířené zabezpečení zpráv, musíte definovat další systémové objekty. Ukázková datová sada thlqual.SCSQPROC(CSQ4INSM) obsahuje požadované definice front.

Ukázka objektu CSQ4INSR

Definuje fronty používané produktem WebSphere Application Server a zprostředkovateli.

Ukázka objektu CSQ4INYD

Pokud používáte distribuované fronty a potřebujete nastavit vlastní fronty, procesy a kanály.

Ukázková datová sada thlqual.SCSQPROC(CSQ4INYD) obsahuje ukázkové definice, které můžete použít pro přizpůsobení distribuovaných objektů front. Zahrnuje:

- Sada definic pro odesílající konec
- Sada definic pro přijímací konec
- Sada definic pro použití klientů

Tuto ukázkou nemůžete použít tak, jak je-musíte ji před použitím upravit. Poté můžete tohoto člena zahrnout do zřetězení definic dat CSQINP2 spouštěcí procedury správce front nebo jej můžete použít jako vstup do funkce COMMAND obslužného programu CSQUTIL a zadat požadované příkazy DEFINE. (To je vhodnější, protože to znamená, že při každém restartování správce front nemusíte tyto objekty předefinovat).

Ukázka objektu CSQ4INYC

Používáte-li klastrování, jsou definice ekvivalentní definicím kanálů a definicím vzdálených front distribuovaných front vytvářeny automaticky v případě potřeby. Některé ruční definice kanálů jsou však potřebné-přijímací kanál klastru pro klastr a odesílací definice klastru pro alespoň jednoho správce front úložiště klastru.

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INYC) obsahuje následující ukázkové definice, které můžete použít pro přizpůsobení objektů klastrování:

- Definice pro správce front
- Definice pro přijímací kanál
- Definice pro odesílací kanál
- Definice pro fronty klastru
- Definice pro seznamy klastrů

Tuto ukázkou nemůžete použít tak, jak je-musíte ji před použitím upravit. Poté můžete tohoto člena zahrnout do zřetězení definic dat CSQINP2 spouštěcí procedury správce front nebo jej můžete použít jako vstup do funkce COMMAND obslužného programu CSQUTIL a zadat požadované příkazy DEFINE. To je vhodnější, protože to znamená, že tyto objekty nemusíte předefinovat pokaždé, když restartujete produkt IBM MQ.

Ukázka objektu CSQ4INYG

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INYG) obsahuje následující ukázkové definice, které můžete použít pro přizpůsobení vlastních objektů pro obecné použití:

- Fronta nedoručených zpráv
- Výchozí přenosová fronta
- Objekty adaptéru CICS

Tuto ukázkou nemůžete použít tak, jak je-musíte ji před použitím upravit. Poté můžete tohoto člena zahrnout do zřetězení definic dat CSQINP2 spouštěcí procedury správce front nebo jej můžete použít jako vstup do funkce COMMAND obslužného programu CSQUTIL a zadat požadované příkazy DEFINE. To je vhodnější, protože to znamená, že tyto objekty nemusíte předefinovat pokaždé, když restartujete produkt IBM MQ.

Kromě ukázkových definic zde můžete použít definice systémových objektů jako základ pro své vlastní definice prostředků. Můžete například vytvořit pracovní kopii produktu SYSTEM.DEFAULT.LOCAL.QUEUE a pojmenujte ji MY.DEFAULT.LOCAL.QUEUE. Poté můžete podle potřeby změnit kterýkoli z parametrů v této kopii. Poté můžete zadat příkaz DEFINE jakoukoli metodou, kterou zvolíte, za předpokladu, že máte oprávnění k vytvoření prostředků tohoto typu.

Výchozí přenosová fronta

Než se rozhodnete, zda chcete definovat výchozí přenosovou frontu, přečtěte si popis [Výchozí přenosová fronta](#).

- Pokud se rozhodnete, že chcete definovat výchozí přenosovou frontu, nezapomeňte, že musíte také definovat kanál, který ji bude obsluhovat.
- Pokud se rozhodnete, že jej nechcete definovat, nezapomeňte odebrat příkaz DEFXMLTQ z příkazu ALTER QMGR v ukázce.

Objekty adaptéru CICS

Ukázka definuje inicializační frontu s názvem CICS01.INITQ. Tuto frontu používá transakce CKTI dodávaná společností IBM MQ . Název této fronty můžete změnit; musí se však shodovat s názvem uvedeným v tabulce inicializace systému CICS (SIT) nebo v přepisu SYSIN v příkazu INITPARM.

Ukázky objektů CSQ4INYS/CSQ4INYR

Definice paměťových tříd pro použití:

- jedna sada stránek pro každou třídu zpráv
- více sad stránek pro hlavní třídy zpráv

Například SYSTEM.COMMAND.INPUT používá STGCLASS ('SYSVOLAT') a SYSTEM.CLUSTER.TRANSMIT.QUEUE používá STGCLASS ('REMOTE'). V produktu CSQ4INYSobě tyto paměťové třídy používají stejnou sadu stránek. V produktu CSQ4INYRtyto paměťové třídy používají různé sady stránek, aby se snížil dopad plnění přenosové fronty.

Ukázky CSQINPT

CSQ4INST

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INST) obsahuje definici pro systémový výchozí odběr.

Musíte definovat objekt v této ukázce, ale musíte to provést pouze jednou při prvním spuštění stroje publikování/odběru. Nejlepší způsob, jak to provést, je zahrnout definici do datové sady CSQINPT. Je udržován při ukončení a restartování správce front. Název objektu nesmíte změnit, ale v případě potřeby můžete změnit jeho atributy.

CSQ4INYT

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INYT) obsahuje sadu příkazů, které můžete chtít spustit při spuštění stroje publikování/odběru. Tato ukázka zobrazuje informace o tématu a odběru.

Jiný

Ukázka zobrazení CSQ4DISP

Ukázková datová sada: thlqual.SCSQPROC(CSQ4DISP) obsahuje sadu generických příkazů DISPLAY, které zobrazují všechny definované prostředky ve vašem správci front. To zahrnuje definice pro všechny objekty IBM MQ a definice, jako jsou paměťové třídy a trasování. Tyto příkazy mohou generovat velké množství výstupu. Tuto ukázku můžete použít v datové sadě CSQINP2 nebo jako vstup do funkce COMMAND obslužného programu CSQUTIL.

Ukázka CSQ4INPX

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INPX) obsahuje sadu příkazů, které můžete chtít provést při každém spuštění inicializátoru kanálu. Tuto ukázku musíte před použitím upravit; poté ji můžete zahrnout do datové sady CSQINPX pro inicializátor kanálu.

Ukázky CSQ4IVPQ a CSQ4IVPG

Ukázkové datové sady: thlqual.SCSQPROC(CSQ4IVPQ) a thlqual.SCSQPROC(CSQ4IVPG) obsahují sady příkazů DEFINE, které jsou nezbytné pro spuštění programů IVP (installation verification programs).

Tyto ukázky můžete zahrnout do datové sady CSQINP2 . Pokud jste úspěšně spustili IVP, nemusíte je spouštět znovu při každém restartování správce front. Proto nemusíte tyto ukázky uchovávat ve zřetězení CSQINP2 trvale.

Ukázky CSQ4MSTR a CSQ4MSRR

Toto jsou ukázkové procedury spuštěných úloh pro správce front: thlqual.SCSQPROC(CSQ4MSTR) a thlqual.SCSQPROC(CSQ4MSRR).

CSQ4MSRR používá CSQ4INYR ve zřetězení CSQINP2 , aby se důležité fronty rozložily mezi různé sady stránek.

Komentáře můžete odebrat, abyste mohli v případě potřeby použít kartu CSQMINI pro nově vytvořené správce front.

Ukázka CSQ4QMIN

Ukázková datová sada QMINI, thlqual.SCSQPROC(CSQ4QMIN).

Podrobnosti o datové sadě QMINI a sekci **TransportSecurity** viz [Datová sada QMINI](#) .

z/OS Obnova a restart na systému z/OS

Pomocí odkazů v tomto tématu zjistíte informace o funkcích produktu IBM MQ for z/OS pro restart a obnovu.

Produkt IBM MQ for z/OS má robustní funkce pro restart a obnovu. Informace o způsobu zotavení správce front po jeho zastavení a o tom, co se stane při jeho restartování, naleznete v následujících dílčích tématech:

- [“Jak jsou prováděny změny dat v produktu IBM MQ for z/OS” na stránce 244](#)
- [“Jak je udržována konzistence v produktu IBM MQ for z/OS” na stránce 246](#)
- [“Co se stane během ukončení v IBM MQ for z/OS” na stránce 248](#)
- [“Co se stane během restartu a obnovy v IBM MQ for z/OS” na stránce 249](#)
- [“Jak jsou vyřešeny sporné jednotky zotavení” na stránce 251](#)
- [“Obnova sdílené fronty” na stránce 254](#)

Související pojmy

z/OS [IBM MQ for z/OS akce obnovy](#)

Související úlohy

[Plánování zálohování a obnovy](#)

z/OS [Správa serveru z/OS](#)

Související odkazy

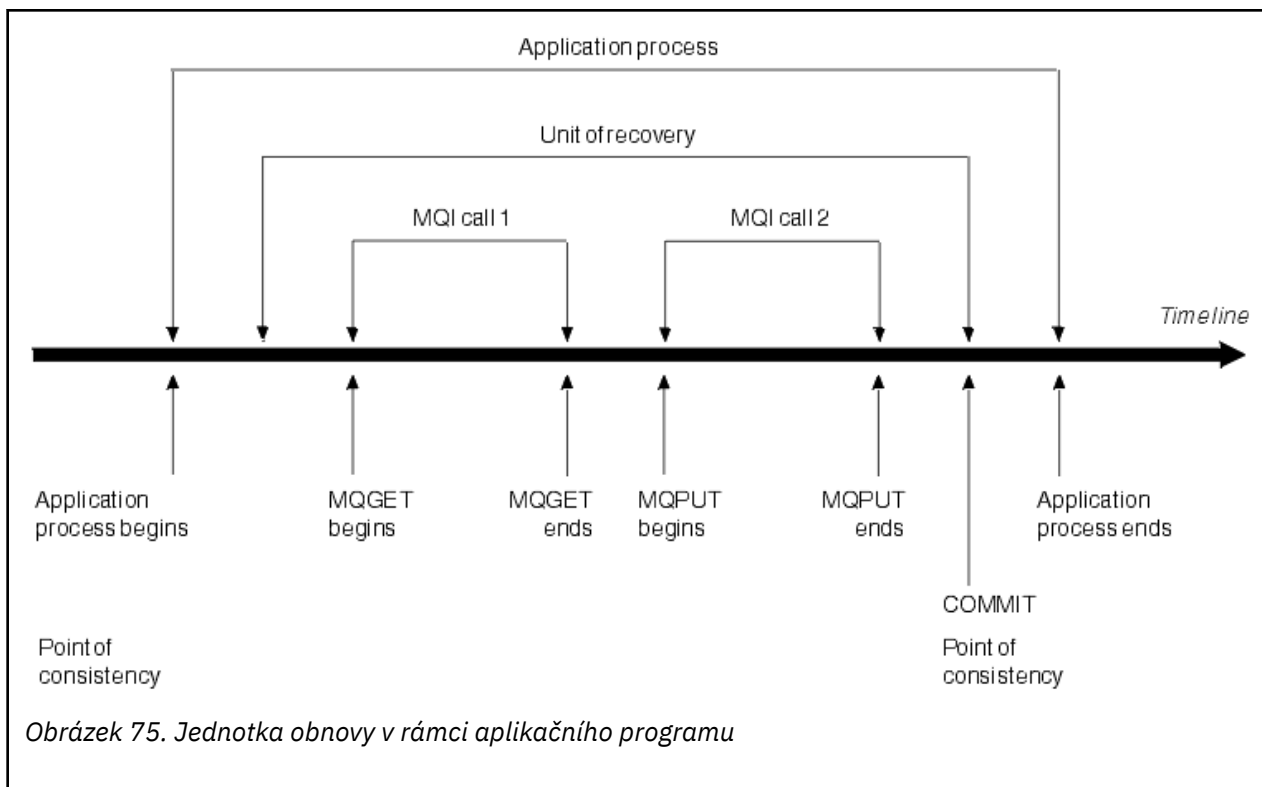
z/OS [Zprávy pro IBM MQ for z/OS](#)

z/OS Jak jsou prováděny změny dat v produktu IBM MQ for z/OS

Produkt IBM MQ for z/OS musí interaktivně spolupracovat s ostatními subsystemy, aby byla všechna data konzistentní. Toto téma obsahuje informace o *jednotkách zotavení*, o tom, co jsou a jak se používají v *zpětných* informacích.

Jednotky využití

Jednotka zotavení je zpracování prováděné jedním správcem front pro aplikační program, které mění data produktu IBM MQ z jednoho bodu konzistence na jiný. *Bod konzistence* -nazývaný také *synchronizační bod* nebo *bod potvrzení* -je bod v čase, kdy jsou všechna obnovitelná data, ke kterým přistupuje aplikační program, konzistentní.



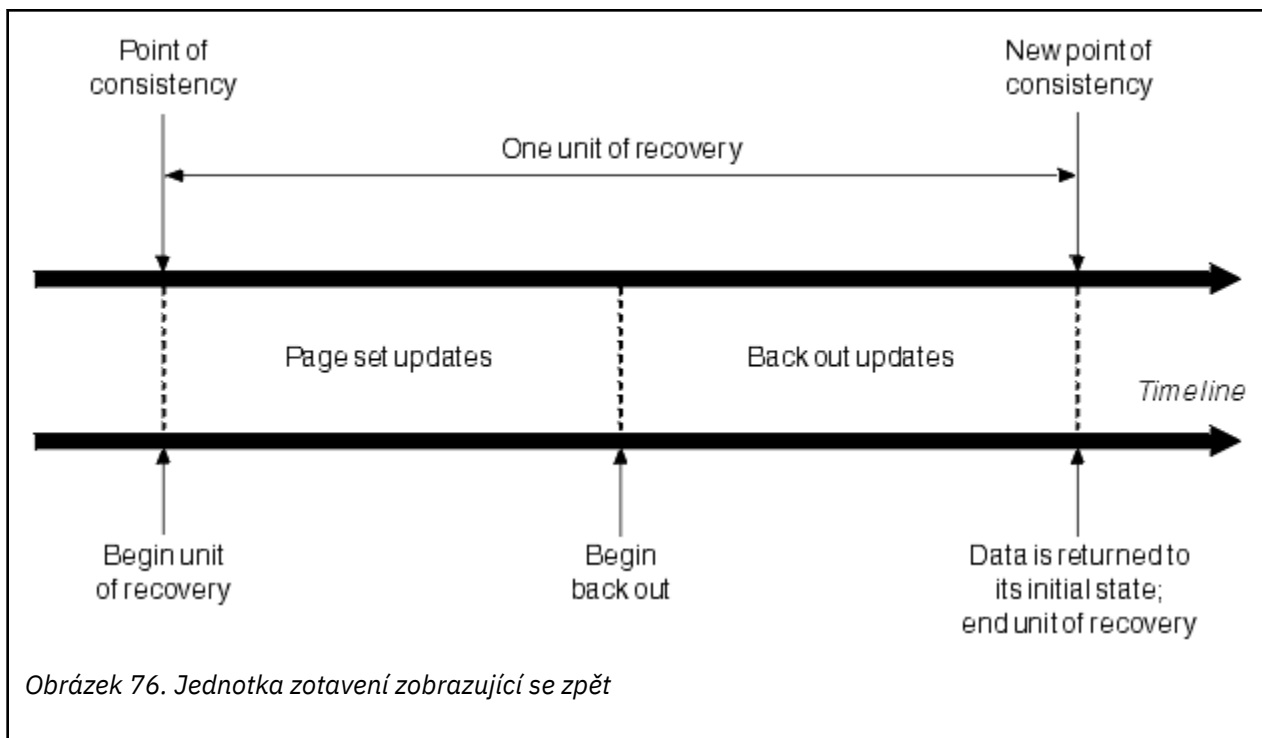
Jednotka obnovy začíná první změnou dat po začátku programu nebo po předchozím bodu konzistence; končí pozdějším bodem konzistence. Obrázek 75 na stránce 245 zobrazuje vztah mezi jednotkami obnovy, bodem konzistence a aplikačním programem. V tomto příkladu provádí aplikační program změny front prostřednictvím volání MQI 1 a 2. Aplikační program může obsahovat více než jednu jednotku obnovy nebo pouze jednu. Avšak jakákoli úplná jednotka obnovy končí v bodu potvrzení.

Například bankovní transakce převádí prostředky z jednoho účtu na druhý. Za prvé, program odečte částku od prvního účtu, účtu A. Pak přičte částku k druhému účtu, B. Po odečtení částky od A jsou dva účty nekonzistentní a IBM MQ nemůže potvrdit. Stanou se konzistentními, když se částka přidá na účet B. Po dokončení obou kroků může program oznámit bod konzistence prostřednictvím potvrzení, čímž se změny zviditelní pro ostatní aplikační programy.

Normální ukončení aplikačního programu automaticky způsobí bod konzistence. Některé požadavky na programy v programech CICS a IMS také způsobují bod konzistence, například EXEC CICS SYNCPOINT.

Vycouvání práce

Dojde-li k chybě v rámci obnovovací jednotky, produkt IBM MQ odebere veškeré změny dat a vrátí data do stavu na začátku obnovovací jednotky; to znamená, že produkt IBM MQ práci vrátí zpět. Události jsou zobrazeny v souboru Obrázek 76 na stránce 246.



z/OS Jak je udržována konzistence v produktu IBM MQ for z/OS

Data v souboru IBM MQ for z/OS musí být konzistentní s dávkovým, CICS, IMS nebo TSO. Jakákoli data změněná v jednom se musí shodovat se změnou v druhém.

Než jeden systém potvrdí změněná data, musí vědět, že druhý systém může provést odpovídající změnu. Takže systémy musí komunikovat.

Během *dvoufázového potvrzování* (například v části CICS) koordinuje proces jeden subsystém. Tento subsystém se nazývá *koordinátor*; druhým je *účastník*. CICS nebo IMS je vždy koordinátorem v interakcích s produktem IBM MQ a IBM MQ je vždy účastníkem. V dávkovém prostředí nebo v prostředí TSO se může produkt IBM MQ účastnit protokolů dvoufázového potvrzování koordinovaných produktem z/OS RRS.

Během *jednofázového potvrzování* (například v rámci TSO nebo dávky) je IBM MQ vždy koordinátorem v interakcích a zcela řídí proces potvrzování.

V prostředí WebSphere Application Server sémantika objektu relace JMS určuje, zda je použita koordinace jednofázového nebo dvoufázového potvrzování.

Konzistence s CICS nebo IMS

Připojení mezi produktem IBM MQ a produktem CICS nebo IMS podporuje následující protokoly synchronizačního bodu:

- Dvoufázové potvrzování-pro transakce, které aktualizují prostředky vlastněné více než jedním správcem prostředků.

Jedná se o standardní protokol distribuovaného synchronizačního bodu. Zahrnuje více protokolů a toků zpráv než jednofázové potvrzování.

- Jednofázové potvrzování-pro transakce, které aktualizují prostředky vlastněné jedním správcem prostředků (IBM MQ).

Tento protokol je optimalizován pro protokolování a toky zpráv.

- Vynechání synchronizačního bodu-pro transakce, které zahrnují IBM MQ, ale neprovádějí nic ve správcí front, který vyžaduje synchronizační bod (například procházení fronty).

V každém případě se CICS nebo IMS chová jako správce synchronizačního bodu.

Fáze dvoufázového potvrzování, které produkt IBM MQ používá ke komunikaci s produktem CICS nebo IMS, jsou následující:

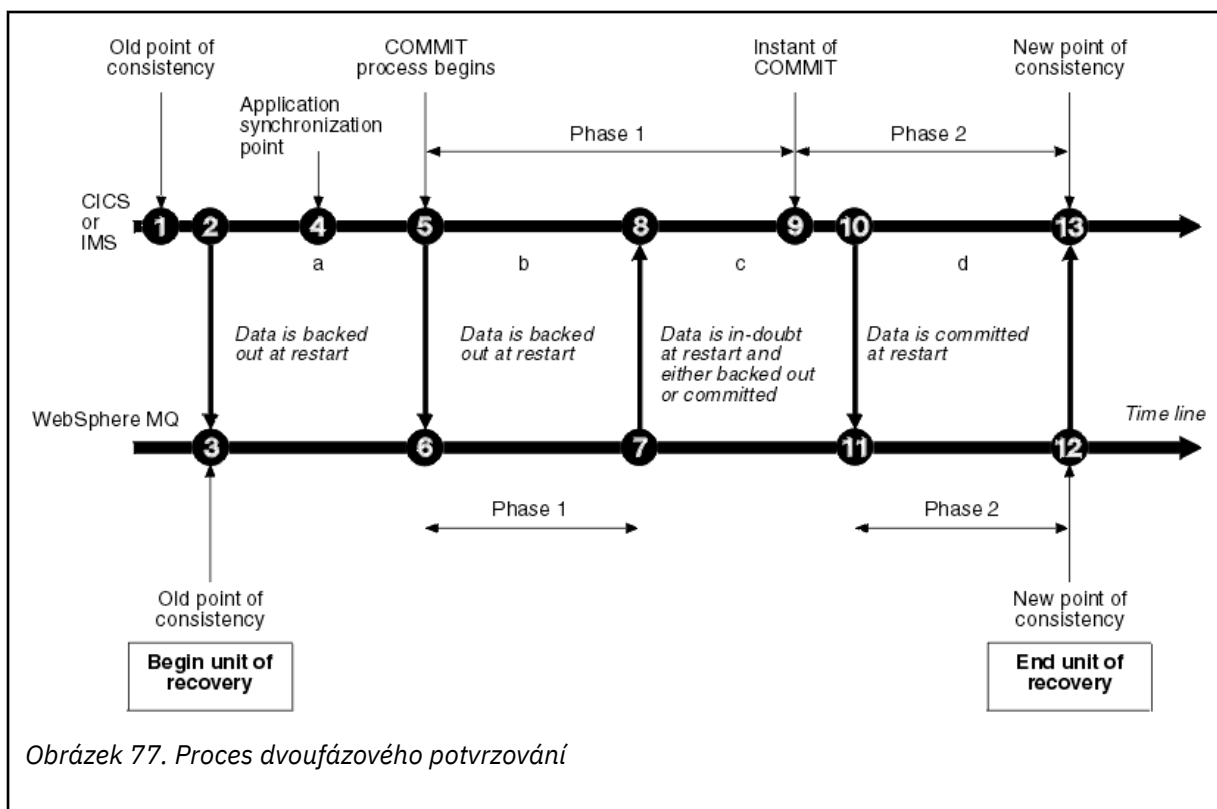
1. Ve fázi 1 každý systém nezávisle určí, zda má ve svém protokolu zaznamenáno dostatek informací o zotavení, a může potvrdit svou práci.

Na konci fáze systémy komunikují. Pokud s tím souhlasí, každá z nich zahájí další fázi.

2. Ve fázi 2 jsou změny trvalé. Pokud se jeden ze systémů během fáze 2 neukončí, operace se dokončí procesem obnovy během restartu.

Ilustrace procesu dvoufázového potvrzování

Obrázek 77 na stránce 247 ilustruje proces dvoufázového potvrzování. Události v koordinátoru CICS nebo IMS jsou zobrazeny na horním řádku, události v IBM MQ na dolním řádku.



Obrázek 77. Proces dvoufázového potvrzování

Číslo v následující části jsou spojena s čísly uvedenými na obrázku.

1. Data v koordinátorovi jsou v bodě konzistence.
2. Aplikační program v koordinátorovi volá IBM MQ, aby aktualizoval frontu přidáním zprávy.
3. Tím se spustí jednotka obnovy v souboru IBM MQ.
4. Zpracování v koordinátorovi pokračuje, dokud není dosaženo synchronizačního bodu aplikace.
5. Koordinátor poté zahájí zpracování potvrzení. Programy CICS používají příkaz SYNCPOINT nebo normální ukončení aplikace ke spuštění potvrzení. Programy IMS mohou spustit potvrzení pomocí volání CHPK, volání SYNC, volání GET UNIQUE na IOPCB nebo normálního ukončení aplikace. Začíná fáze 1 zpracování potvrzení.
6. Vzhledem k tomu, že koordinátor zahajuje zpracování fáze 1, pracuje také s produktem IBM MQ.
7. Produkt IBM MQ úspěšně dokončí fázi 1, zapíše tuto skutečnost do svého protokolu a oznámí to koordinátorovi.
8. Koordinátor obdrží oznámení.

9. Koordinátor úspěšně dokončil zpracování fáze 1. Nyní oba subsystémy souhlasí s potvrzením změn dat, protože obě dokončily fázi 1 a mohly by se zotavit z jakýchkoli chyb. Koordinátor zaznamenává ve svém protokolu okamžik potvrzení-neodvolatelné rozhodnutí dvou subsystémů provést změny.

Koordinátor nyní zahájí fázi 2 zpracování-skutečný závazek.

10. Koordinátor oznámí produktu IBM MQ zahájení fáze 2.
11. Produkt IBM MQ protokoluje začátek fáze 2.
12. Fáze 2 byla úspěšně dokončena a nyní se jedná o nový bod konzistence pro IBM MQ. Produkt IBM MQ poté oznámí koordinátorovi, že dokončil zpracování fáze 2.
13. Koordinátor dokončí zpracování fáze 2. Data řízená oběma subsystémy jsou nyní konzistentní a dostupná pro jiné aplikace.

Jak je zachována konzistence po abnormálním ukončení

Je-li správce front restartován po nestandardním ukončení, musí určit, zda mají být potvrzeny nebo ukončeny jednotky zotavení, které byly v době ukončení aktivní. Pro některé jednotky zotavení má produkt IBM MQ dostatek informací k rozhodnutí. Pro ostatní, to není, a musí získat informace od koordinátora, když je připojení znovu.

Obrázek 77 na stránce 247 zobrazuje čtyři tečky v rámci dvou fází: a, b, c a d. Stav jednotky obnovy závisí na období, ve kterém došlo k ukončení. Stav může mít jednu z následujících hodnot:

Za letu

Správce front skončil před dokončením fáze 1 (období a nebo b); během restartu produkt IBM MQ provede odvolání aktualizací.

Nejisté

Správce front skončil po dokončení fáze 1 a před spuštěním fáze 2 (období c); pouze koordinátor ví, zda k chybě došlo před potvrzením nebo po něm (bod 9). Pokud k tomu došlo dříve, IBM MQ musí své změny vrátit zpět; pokud k nim došlo později, IBM MQ musí provést své změny a potvrdit je. Při restartu produkt IBM MQ čeká na informace od koordinátora před zpracováním této jednotky zotavení.

V potvrzení

Správce front skončil po zahájení vlastního zpracování fáze 2 (období d); provádí potvrzené změny.

V zpětném vrácení

Správce front skončil poté, co byla jednotka zotavení vrácena zpět, ale před dokončením procesu (nikoli na obrázku) během restartu produkt IBM MQ pokračuje v vrácení změn zpět.

Co se stane během ukončení v IBM MQ for z/OS

Správce front se obvykle ukončuje v reakci na příkaz STOP QMGR. Pokud se správce front zastaví z jiného důvodu, je ukončení nestandardní.

Všimněte si, že během ukončení správce front produkt IBM MQ interně zadá příkaz

```
DISPLAY CONN(*) TYPE(CONN) ALL WHERE (APPLTYPE NE SYSTEMAL)
```

takže jste si vědomi toho, které podprocesy mohou bránit správci front v dokončení ukončení práce systému.

SYSTEMAL odpovídá APPLTYPES buď SYSTEM, nebo CHINIT, takže příkaz DISPLAY CONN filtrující typy aplikací, které neodpovídají SYSTEMAL, se vrátí do informací protokolu úlohy o podprocesech, které by mohly bránit normálnímu ukončení práce systému.

Normální ukončení

Při normálním ukončení produkt IBM MQ řádným způsobem zastaví všechny aktivity. Produkt IBM MQ můžete zastavit pomocí režimu uvedení do klidového stavu, vynucení nebo restartování. Účinky jsou uvedeny v části [Tabulka 23 na stránce 249](#).

<i>Tabulka 23. Ukončení pomocí QUIESCE, FORCE a RESTART</i>			
Typ podprocesu	QUIESCE	Vynutit	RESTART
Aktivní podprocesy	Spustit do dokončení	Vrátit zpět	Vrátit zpět
Nové podprocesy	Může začít	Nepovoleno	Nepovoleno
Nová připojení	Nepovoleno	Nepovoleno	Nepovoleno

Dávkové aplikace jsou upozorněny, pokud dojde k ukončení v době, kdy je aplikace stále připojena.

S produktem CICSse aktuální podproces spustí pouze do konce jednotky zotavení. V případě produktu CICSzastaví zastavení správce front v klidovém režimu adaptér CICS , a pokud tedy aktivní úloha obsahuje více než jednu jednotku zotavení, nemusí být tato úloha nutně dokončena.

Pokud zastavíte správce front v režimu vynucení nebo restartování, nebudou přiděleny žádné nové podprocesy a práce na připojených podprocesech bude odvolána. Pomocí těchto režimů lze vytvořit neověřené jednotky zotavení pro podprocesy, které jsou mezi fázemi zpracování potvrzení. Jsou rozpoznány, když je produkt IBM MQ znovu připojen k řídicími subsystemy CICS, IMSnebo RRS.

Při zastavení správce front v libovolném režimu se jedná o následující kroky:

1. Připojení jsou ukončena.
2. Produkt IBM MQ přestane přijímat příkazy.
3. Produkt IBM MQ zajistí, že všechny nevyřízené aktualizace sad stránek budou dokončeny.
4. Příkaz DISPLAY USAGE je vydán interně produktem IBM MQ , takže je restartovaný adaptér RBA zaznamenán v protokolu konzoly z/OS .
5. Kontrolní bod vypnutí je převzat a BSDS je aktualizován.

Ukončení, která určují režim uvedení do klidového stavu, neovlivňují neověřené jednotky zotavení. Každá jednotka, která je na pochybách, zůstává na pochybách.

Nestandardní ukončení

Nestandardní ukončení může zanechat data v nekonzistentním stavu, například:

- Jednotka zotavení byla přerušena před dosažením bodu konzistence.
- Potvrzená data nebyla zapsána do sad stránek.
- Nepotvrzená data byla zapsána do sad stránek.
- Aplikační program byl přerušen mezi fází 1 a fází 2 procesu potvrzení, takže jednotka zotavení byla nejistá.

Produkt IBM MQ řeší veškeré nekonzistence dat vyplývající z nestandardního ukončení během restartu a obnovy.

Co se stane během restartu a obnovy v IBM MQ for z/OS

Produkt IBM MQ používá protokol pro zotavení a datovou sadu zaváděcího programu (BSDS) k určení, co se má obnovit při restartování. BSDS identifikuje datové sady aktivního a archivního protokolu a umístění nejnovějšího kontrolního bodu IBM MQ v protokolu.

Úvod do restartování a obnovy

Po inicializaci produktu IBM MQ probíhá proces restartování správce front následujícím způsobem:

- Inicializace protokolu

- Znovusestavení aktuálního stavu
- Dopředné zotavení žurnálu
- Zpětná obnova protokolu
- Znovusestavení indexu fronty

Po dokončení obnovy:

- Potvrzené změny se projeví v datech.
- Nejistá aktivita se odráží v datech. Data jsou však uzamčena a nelze je použít, dokud produkt IBM MQ nerozpozná a neprovede pochybné rozhodnutí.
- Z front byly odebrány přerušené probíhající a předčasně ukončené změny. Zprávy jsou konzistentní a lze je použít.
- Byl vytvořen nový kontrolní bod.
- Pro indexované fronty obsahující trvalé zprávy byly sestaveny nové indexy (viz [“Znovusestavení indexů fronty”](#) na stránce 251).

Pokud se používají duální BSDS, produkt IBM MQ zkontroluje konzistenci časových razítek v BSDS:

- Pokud jsou obě kopie BSDS aktuální, produkt IBM MQ testuje, zda jsou obě časová razítka stejná. Pokud nejsou, IBM MQ vydá zprávu CSQJ120E a ukončí se. K tomu může dojít, když jsou dvě kopie BSDS udržovány na samostatných svazcích DASD a jeden ze svazků byl obnoven, když byl správce front zastaven. Produkt IBM MQ zjistí situaci při restartu.
- Pokud byla jedna kopie BSDS dealokována a protokolování pokračovalo s jedním BSDS, může dojít k problému. Pokud jsou obě kopie BSDS udržovány na jednom svazku a svazek byl obnoven, nebo pokud byly obě kopie BSDS obnoveny odděleně, produkt IBM MQ nemusí zjistit obnovu. V takovém případě by záznamy protokolu, které nejsou uvedeny v BSDS, byly pro systém neznámé.

Dávkové aplikace nejsou upozorněny, když dojde k restartu *poté*, co aplikace požádala o připojení.

Základní informace o rozsahu protokolů vyžadovaném pro zotavení

Během restartu závisí rozsah dat protokolu, která musí být přečtena, na mnoha faktorech:

- V době nestandardního ukončení je v systému obvykle mnoho neúplných pracovních jednotek. Jak bylo popsáno dříve, zpracování restartu uvede systém do stavu konzistence, což může zahrnovat zálohování jednotek zpracování nebo obnovu zámek na neověřených jednotkách práce. Obnova pracovní jednotky vyžaduje, aby byly k dispozici všechny záznamy pracovního protokolu pro průlom, neověřené a neověřené jednotky práce. IBM MQ bude 'zkrat' staré jednotky práce, aby bylo možné provést obnovu jednotky práce s použitím mnohem menšího rozsahu dat protokolu.
- V době nestandardního ukončení existuje obvykle mnoho trvalých aktualizací, které jsou uloženy pouze v mezipaměti fondu vyrovnávacích pamětí. Dosud nebyly zapsány na disk. Tyto změny musí být načteny z protokolu a znovu použity na data uložená v sadách stránek. Adaptéry RBA pro zotavení sady stránek v kontrolním bodu popisují nejnižší protokol RBA nezbytný pro aktualizaci sad stránek do konzistentního stavu.
- Pokud byly do systému zavedeny staré sady stránek, například byla zavedena záloha sady stránek pro obnovu po selhání média, všechny změny musí být načteny z protokolu od okamžiku, kdy byla záloha provedena. Tyto změny se znovu použijí na data zadržená v obnovované sadě stránek. Oblasti RBA pro zotavení sady stránek, které jsou uloženy na stránce 0 sady stránek, popisují nejnižší protokol RBA nezbytný pro zotavení sady stránek z médií.
- Používáte-li trvalé zprávy ve sdílených frontách, je k obnově struktury CFstruktura, která zadržuje trvalé zprávy, vyžadována řada dat protokolu. Nejstarší data protokolu, která by byla požadována k provedení nápravy CFSTRUCT, jsou přibližně od doby, kdy byl starý CFSTRUCT BACKUP.

Během normálního spuštění lze použít příkaz DISPLAY USAGE TYPE (DATASET) k zobrazení rozsahu protokolu pro zotavení přidruženého k těmto faktorům (samozřejmě nemůže poskytnout informace kvůli opětovnému zavedení starých sad stránek). Chcete-li se vyhnout problémům, které by mohly prodloužit

restart správce front v případě nestandardního ukončení, pravidelně monitorujte výstup hodnot z DISPLAY USAGE TYPE (DATASET).

Kromě toho správce front vydává informační zprávy týkající se těchto faktorů:

- CSQJ160I a CSQJ161I varují před přerušitými pracovními jednotkami.
- CSQR026I a CSQR027I poskytují informace o tom, zda tyto dlouho běžící jednotky práce byly úspěšně posunovány.
- CSQE040I a CSQE041E varují, že zálohy struktury stárnou, a proto by operace RECOVER CFSTRUCT trvala dlouhou dobu.

Určení, která aplikace má dlouhotrvající pracovní jednotku

Je možné určit aplikaci s dlouhotrvající jednotkou práce. K tomu použijte příkaz DISPLAY CONN.

Příkaz DISPLAY CONN vrací informace o připojení pro všechny aplikace připojené ke správci front spolu s dalšími informacemi, které vám pomohou určit, které aplikace mají aktuálně dlouhotrvající pracovní jednotku. Informace vrácené příkazem DISPLAY CONN jsou podobné informacím vráceným příkazem DISPLAY QSTATUS, ale hlavní rozdíl je v tom, že příkaz DISPLAY CONN zobrazuje informace o objektech a informace o transakcích pro konkrétní připojení, a nikoli podrobnosti o tom, která připojení jsou přidružena k určitému objektu.

Pro každou připojenou aplikaci příkaz DISPLAY CONN vrací následující informace:

- Základní informace včetně ID připojení a PID.
- Transakční informace pro toto připojení, včetně času a data vytvoření transakce (tj. kdy byla provedena první operace MQGET/PUT pod synchronizačním bodem) a kdy transakce poprvé zapsala do protokolu.
- Informace o čase protokolu označující, která aplikace má stále dlouhotrvající jednotku práce.
- Seznam všech objektů, které má připojení aktuálně otevřené. Podrobnosti pro každý objekt jsou vráceny jako samostatná zpráva s ID připojení použitým jako klíč. Vzhledem k tomu, že existují různé typy objektů, jako jsou fronty a správci front, jsou informace zobrazené s objektem specifické pro příslušný typ objektu.

Znovusestavení indexů fronty

Chcete-li zvýšit rychlost operací MQGET ve frontě, ve které nejsou zprávy načítány sekvenčně, můžete určit, že má produkt IBM MQ udržovat index identifikátorů zprávy nebo korelace nebo ID skupiny pro všechny zprávy v dané frontě.

Po restartování správce front jsou tyto indexy znovu sestaveny pro každou frontu. Toto platí pouze pro trvalé zprávy; přechodné zprávy jsou při restartu odstraněny. Pokud indexované fronty obsahují velký počet trvalých zpráv, zvýší se doba potřebná k restartování správce front.

Pomocí parametru QINDXBLD makra CSQ6SYSP můžete zvolit opětovné asynchronní sestavení indexů pro spuštění správce front. Pokud nastavíte QINDXBLD=NOWAIT, IBM MQ se restartuje bez čekání na znovusestavení indexů.

Jak jsou vyřešeny sporné jednotky zotavení

Pokud produkt IBM MQ ztratí připojení k jinému správci prostředků, obvykle se při restartu pokusí obnovit všechny nekonzistentní objekty.

Pokud produkt IBM MQ ztratí připojení k serveru CICS, IMS nebo RRS, obvykle se pokusí při restartu obnovit všechny nekonzistentní objekty. Informace potřebné k vyřešení pochybných jednotek obnovy musí pocházet z koordinačního systému. Další části popisují proces řešení pro různá prostředí.

- [Způsob řešení nejistých jednotek zotavení z CICS](#)
- [Způsob řešení nejistých jednotek zotavení z IMS](#)
- [Způsob řešení nejistých jednotek zotavení z RRS](#)

- Způsob vyřešení nejistých jednotek zotavení s dispozicí obnovy GROUP

Jak jsou vyhodnoceny sporné jednotky zotavení z CICS

Za určitých okolností produkt CICS nemůže spustit proces IBM MQ , který by vyřešil neověřené jednotky zotavení. Pokud k tomu dojde, produkt IBM MQ odešle jednu z následujících zpráv:

- CSQC404E
- CSQC405E
- CSQC406E
- CSQC407E

následovaná zprávou CSQC408I.

Podrobnosti o tom, co tyto zprávy znamenají, naleznete v příručce IBM MQ for z/OS zprávy, dokončení, a kódy příčiny .

Vyřešení nejistých jednotek neovlivní prostředky CICS . Produkt CICS řídí koordinaci obnovy a po restartování automaticky potvrdí nebo odvolá každou jednotku v závislosti na tom, zda existoval záznam protokolu označující začátek potvrzení. Existence nejistých objektů nezamyká CICS prostředky, zatímco IBM MQ se znovu připojuje.

Jednou z funkcí adaptéru CICS je udržovat data synchronizovaná mezi CICS a IBM MQ. Pokud dojde k nestandardnímu ukončení správce front v době připojení k produktu CICS, je možné, aby produkt CICS potvrdil nebo ukončil práci, aniž by o tom IBM MQ byl informován. Po restartování správce front je tato práce označována jako *nejistá*.

Produkt IBM MQ nemůže vyřešit tyto neověřené jednotky zotavení (tj. potvrdit nebo vrátit změny provedené v prostředcích IBM MQ), dokud nebude připojení k produktu CICS restartováno nebo znovu připojeno.

Proces vyřešení nejistých jednotek zotavení je zahájen během spouštění adaptéru CICS . Proces se spustí, když adaptér požádá o seznam neověřených jednotek zotavení. Poté:

- Adaptér obdrží seznam nejistých jednotek zotavení pro toto ID připojení z produktu IBM MQa předá je produktu CICS k vyřešení.
- Produkt CICS porovnává položky z tohoto seznamu s položkami ve vlastním protokolu. CICS určí ze svého vlastního seznamu, jakou akci provede pro každou neověřenou jednotku obnovy.

Pro všechny vyřešené jednotky produkt IBM MQ aktualizuje fronty podle potřeby a uvolní odpovídající zámky. Nevyřešené jednotky mohou zůstat po restartu. Vyřešte je metodami popsány v souboru Administrace IBM MQ for z/OS.

Jak jsou vyhodnoceny sporné jednotky zotavení z IMS

Vyřešení nejistých jednotek obnovy v produktu IMSneovlivní DL/I prostředky. Produkt IMS řídí koordinaci obnovy a po restartování automaticky potvrdí nebo odvolá nedokončenou práci DL/I. Rozhodnutí o potvrzení nebo odvolání pro oblasti online (nezkrácené cesty) je na přítomnosti nebo nepřítomnosti typů záznamů protokolu IMS X'3730 'a X'3801'. Existence nejistých jednotek obnovy neznamená, že záznamy DL/I jsou uzamčeny, dokud se produkt IBM MQ nepřipojí.

Během restartu správce front produkt IBM MQ vytvoří seznam nejistých jednotek zotavení. Produkt IMS sestaví svůj vlastní seznam zbytkových položek obnovy (RREs). Položky RREs jsou protokolovány v kontrolních bodech IMS , dokud nejsou vyřešeny všechny položky.

Během opětovného připojení IMS oblasti k IBM MQ IMS označuje IBM MQ , zda potvrdit nebo vrátit zpět jednotky práce označené v IBM MQ jako pochybné.

Po vyřešení nejistých jednotek:

1. Pokud produkt IBM MQ rozpozná, že označil položku pro potvrzení a IMS označil, že má být vrácena zpět, IBM MQ vydá zprávu CSQQ010E. Produkt IBM MQ vydá tuto zprávu pro všechny nekonzistence tohoto typu mezi IBM MQ a IMS.

2. Pokud má produkt IBM MQ nějaké zbývající neověřené jednotky, adaptér vydá zprávu CSQQ008I.

Pro všechny vyřešené jednotky produkt IBM MQ aktualizuje fronty podle potřeby a uvolní odpovídající zámky.

Produkt IBM MQ udržuje zámky na pochybné práci, která nebyla vyřešena. To může způsobit nevyřízené požadavky v systému, pokud jsou drženy důležité zámky. Připojení zůstává aktivní, takže můžete vyřešit IMS RREs. Obnovte neověřené podprocesy metodami popsány v souboru [Administrace IBM MQ for z/OS](#).

Veškerá pochybná práce by měla být vyřešena, pokud se nevyřeší softwarové nebo provozní problémy, například při studeném startu systému IMS . Pochybné řešení ze strany kontrolní oblasti IMS se odehrává za dvou okolností:

1. Na začátku připojení k produktu IBM MQ, během kterého se rozlišení provádí synchronně.
2. Když se program neukončí, během kterého se rozlišení provádí asynchronně.

Jak jsou z RRS vyřešeny sporné jednotky zotavení

Jednou z funkcí adaptéru RRS je udržovat data synchronizovaná mezi produktem IBM MQ a ostatními správci prostředků, kteří se účastní RRS. Dojde-li k selhání v okamžiku, kdy produkt IBM MQ dokončí jednu fázi potvrzení a čeká na rozhodnutí z RRS (koordinátor potvrzení), přejde jednotka zotavení do stavu nejistoty.

Když je komunikace mezi RRS a IBM MQ znovu zavedena, RRS automaticky potvrdí nebo odvolá každou jednotku zotavení v závislosti na tom, zda existoval záznam protokolu označující začátek potvrzení. Produkt IBM MQ nemůže vyřešit tyto neověřené jednotky zotavení (tj. potvrdit nebo vrátit změny provedené v prostředcích IBM MQ), dokud nebude znovu navázáno připojení k RRS.

Za určitých okolností nemůže služba RRS vyřešit pochybné jednotky zotavení. Když k tomu dojde, produkt IBM MQ odešle jednu z následujících zpráv na konzolu z/OS :

- CSQ3011I
- CSQ3013I
- CSQ3014I
- CSQ3016I

Podrobnosti o tom, co tyto zprávy znamenají, naleznete v příručce [IBM MQ for z/OS zprávy, dokončení, a kódy příčiny](#) .

Pro všechny vyřešené jednotky obnovy produkt IBM MQ aktualizuje fronty podle potřeby a uvolní odpovídající zámky. Nevyřešené jednotky obnovy mohou zůstat po restartu. Vyřešte je pomocí metody popsané v části [Administrace IBM MQ for z/OS](#).

Jak jsou vyřešeny neověřené jednotky zotavení s dispozicí skupinové jednotky zotavení

Neověřené transakce, které mají jednotku zotavení GROUP, mohou být vyřešeny koordinátorem transakcí libovolným správcem front ve skupině sdílení front (QSG), kde je povolen atribut správce front GROUPUR. Kdykoli koordinátor transakcí znovu připojí, obvykle si vyžádá seznam všech nevyřízených neověřených transakcí a poté je vyřeší pomocí informací ze svých protokolů.

Pokud koordinátor transakcí, který je připojen k dispoziční jednotce zotavení GROUP, požádá o seznam neověřených transakcí, bude vrácený seznam tvořen všemi neověřenými transakcemi s dispozicí zotavení skupiny, která existuje v rámci skupiny sdílení front. Tento seznam není závislý na tom, na kterém správci front byly spuštěny neověřené transakce. Správce front, který zpracovává takový požadavek,

zkompile seznam prostřednictvím komunikace se všemi ostatními aktivními správci front ve skupině sdílení front pomocí volby `SYSTEM.QSG.UR.RESOLUTION.QUEUE`. Správce front poté načte protokoly všech neaktivních správců front z jejich posledního kontrolního bodu, aby identifikoval případné další neověřené transakce, které by ohlásili, kdyby byli aktivní.

Když koordinátor transakcí požádá o vyřešení sporné transakce, správce front, ke kterému je připojen, identifikuje, zda transakce pochází sama od sebe, a pokud ano, řeší ji stejným způsobem jako transakce s dispozicí jednotky zotavení QMGR. Pokud transakce pochází z jiného aktivního správce front v rámci skupiny sdílení front, bude požadavek na dokončení řešení přesměrován na tohoto správce front pomocí volby `SYSTEM.QSG.UR.RESOLUTION.QUEUE`. V případě, že transakce pochází z neaktivního správce front v rámci skupiny sdílení front, je veškerá práce ve sdílené frontě okamžitě vyřešena a požadavek na vyřešení všech zbývajících soukromých úloh ve frontě je umístěn do systému `SYSTEM.QSG.UR.RESOLUTION.QUEUE`. Neaktivní správce front zpracuje tento požadavek při spuštění před přijetím nové práce. V tomto scénáři protokoly původního správce front stále odrážejí, že jednotka zotavení je nejistá, dokud nebude restartována a nezpracuje požadavek.

Obnova sdílené fronty

V tomto tématu jsou uvedeny informace o zotavení produktu IBM MQ a odolnosti různých komponent v prostředí skupiny sdílení front.

- [“Zotavení transakcí” na stránce 254](#)
- [“Obnova typu peer” na stránce 254](#)
- [“Definice sdílené fronty” na stránce 255](#)
- [“Protokolování” na stránce 255](#)
- [“Selhání prostředku Coupling Facility a struktury” na stránce 255](#)
- [“Scénáře selhání struktury” na stránce 256](#)
- [“Odolnost vůči selháním konektivity prostředku Coupling Facility” na stránce 257](#)
- [“Správa odolnosti vůči selháním konektivity prostředku Coupling Facility” na stránce 258](#)
- [“Provozní chování” na stránce 260](#)

Zotavení transakcí

Pokud aplikace vydá volání MQBACK nebo dojde k jeho nestandardnímu ukončení (například kvůli příkazu EXEC CICS ROLLBACK nebo IMS abend), informace na úrovni podprocesu uložené ve správci front zajistí odvolání probíhající pracovní jednotky. Operace MQPUT a MQGET v rámci synchronizačního bodu ve sdílených frontách jsou odvolány stejným způsobem jako aktualizace nesdílených front.

Obnova typu peer

Dojde-li k selhání správce front, dojde k nestandardnímu odpojení od struktury prostředku Coupling Facility, ke kterým je aktuálně připojen. Dojde-li k selhání připojení mezi instancí z/OS a prostředkem CF (například selhání fyzického spoje nebo vypnutí prostředku CF nebo oblasti), bude také zjištěno nestandardní ukončení připojení mezi správcem front a dotčenými strukturami prostředku CF. Ostatní správci front ve stejné skupině sdílení front, kteří zůstávají připojeni k této struktuře, zjistí nestandardní odpojení a všichni se pokusí zahájit *zotavení typu peer* pro nezdařeného správce front v této struktuře. Pouze jeden z těchto správců front úspěšně iniciuje zotavení typu peer, ale všichni ostatní správci front spolupracují při obnově pracovních jednotek, které byly vlastněny správcem front, který selhal.

Pokud dojde k selhání správce front v případě, že ke struktuře nejsou připojeni žádní rovnocenní správci, dojde k zotavení při připojení jiného správce front k této struktuře nebo při restartování správce front, u kterého došlo k selhání.

Zotavení typu peer, často nazývané PLR (Peer Level Recovery), se provádí na základě struktury a je možné, aby se jeden správce front podílel na zotavení více než jedné struktury současně. Sada rovnocenných

správců spolupracujících při obnově různých struktur se však může lišit v závislosti na tom, kteří správci front byli v době selhání připojeni k různým strukturám.

Po restartování správce front, který se nezdařil, se znovu připojí ke strukturám, ke kterým byl připojen v době selhání, a obnoví všechny zbývající nevyřešené jednotky práce, které nebyly obnoveny pomocí obnovy typu peer.

Obnova typu peer je vícefázový proces. Během první fáze jsou obnoveny jednotky práce, které pokročily mimo probíhající fázi; to může zahrnovat potvrzování zpráv pro jednotky práce, které jsou v potvrzování, a zamykání zpráv pro jednotky práce, které jsou v nejistém stavu. Během druhé fáze jsou kontrolovány fronty, pro které jsou v selhávajícím správci front aktivní podprocesy, odvolány nepotvrzené zprávy týkající se probíhajících pracovních jednotek a informace o aktivních manipulacích ve sdílených frontách ve správci front, pro který došlo k selhání, jsou resetovány. To znamená, že produkt IBM MQ resetuje všechny indikátory, že selhávající správce front měl sdílenou frontu otevřenou pro vstup výhradní, což umožňuje ostatním aktivním správčům front otevřít frontu pro vstup.

Definice sdílené fronty

Objekty front, které představují atributy sdílené fronty, jsou uloženy ve sdíleném úložišti Db2 používaném skupinou sdílení front. Ujistěte se, že jsou zavedeny odpovídající procedury pro zálohování a obnovu tabulek Db2 používaných k zadržení objektů IBM MQ. Pomocí obslužného programu IBM MQ CSQUTIL můžete také vytvořit příkazy MQSC pro přehrání ve správci front za účelem předefinování objektů IBM MQ, včetně definic sdílených front a skupin uložených v produktu Db2.

Protokolování

Skupiny sdílení front mohou podporovat trvalé zprávy, protože zprávy ve sdílených frontách lze protokolovat v protokolech správce front.

Selhání prostředku Coupling Facility a struktury

Existují dva typy selhání, které lze ohlásit pro strukturu prostředku Coupling Facility (CF): selhání struktury a ztráta konektivity. Služby prostředí sysplex pro sdílení dat (XES) informují produkt IBM MQ o selhání struktury prostředku CF nebo o selhání prostředku CF s událostí selhání struktury. Pokud systém XES vytvoří událost ztráty konektivity, nemusí to nutně znamenat, že došlo k problému se strukturou, může se stát, že pro komunikaci se strukturou není k dispozici žádné připojení. Je možné, že ne všichni správci front obdrží pro strukturu událost ztráty konektivity. Záleží na konfiguraci připojení k prostředku CF. Událost ztráty konektivity lze také přijmout kvůli příkazům operátora, například VARY PATH OFFLINE nebo CONFIG CHP OFFLINE.

Struktury prostředku CF používané produktem IBM MQ lze konfigurovat tak, aby používaly duplexování spravované systémem. To znamená, že pokud dojde k jedinému selhání, zpracování překonání selhání spravované systémem skryje selhání struktury nebo ztrátu konektivity a správce front nebude o tomto selhání informován. Dojde-li k selhání obou instancí duplexované struktury nebo připojení, správce front obdrží příslušnou událost a zpracuje ji stejným způsobem jako událost selhání pro jednoduchou strukturu. Podrobnosti o tom, jak správce front zpracovává události, jsou popsány v části [Scénáře](#).

V nepravděpodobném případě selhání prostředku CF nebo struktury dojde ke ztrátě všech dočasných zpráv uložených v ovlivněných strukturách aplikace. Trvalé zprávy můžete obnovit pomocí příkazu RECOVER CFSTRUCT. Pokud se obnovitelná struktura aplikace nezdařila, je zabráněno jakékoli další aktivitě aplikace pro tuto strukturu, dokud nebude struktura obnovena.

Chcete-li zajistit, že budete moci obnovit strukturu prostředku CF v přiměřeném časovém období, provádějte časté zálohy pomocí příkazu BACKUP CFSTRUCT. Můžete se rozhodnout provést zálohy pro libovolné správce front ve skupině sdílení front nebo vyhradit jednoho správce front pro provedení všech záloh. Automatizujte proces vytváření záloh, abyste se ujistili, že jsou vytvářeny pravidelně.

Každá záloha je zapsána do datové sady aktivního protokolu správce front, který zálohu provádí. Úložiště sdílené fronty Db2 zaznamenává název zálohované struktury prostředku CF, název správce front, který zálohu provádí, rozsah RBA pro tuto zálohu v protokolu daného správce front a dobu zálohy.

Administrativní struktura obsahuje informace o nedokončených pracovních jednotkách ve sdílených frontách v době selhání struktury aplikace, takže administrativní struktura musí být k dispozici během zpracování příkazu RECOVER CFSTRUCT. Pokud se struktura administrace nezdařila, musí všichni správci front ve skupině sdílení front znovu sestavit své položky struktury administrace, než budete moci zadat příkaz RECOVER CFSTRUCT.

Správci front znovu sestavují své položky administrativní struktury automaticky a bez ukončení. Není-li správce front v době selhání spuštěn, mohou být jeho položky struktury administrace znovu sestaveny jiným správcem front ve skupině sdílení front, která je spuštěna na stejné nebo vyšší úrovni.

Chcete-li obnovit strukturu aplikace, zadejte pro správce front příkaz RECOVER CFSTRUCT, který má provést zotavení. Můžete obnovit jednu strukturu prostředku CF nebo můžete obnovit několik struktur prostředku CF současně. Můžete provést obnovu pomocí libovolného správce front ve skupině sdílení front, nemusí to být ten, který provedl zálohu, nebo ten, který byl dříve připojen k nezdařené struktuře.

Příkaz RECOVER CFSTRUCT používá zálohu, která je umístěna prostřednictvím informací o úložišti Db2 (Db2 proto musí být k dispozici ve správci front, kde se provádí obnova), a obnoví ji do bodu selhání.

Příkaz RECOVER CFSTRUCT to provede použitím záznamů protokolu z každého správce front ve skupině sdílení front, která provedla operaci MQPUT nebo MQGET mezi spuštěním zálohy a časem selhání, na libovolnou sdílenou frontu, která se mapuje na strukturu prostředku CF. Výsledné sloučení protokolů může vyžadovat načtení značného množství dat protokolu, protože všechna data protokolu zapsaná zúčastněnými správci front od doby, kdy byla záloha načtena. Důrazně se doporučuje provádět časté (například hodinové) zálohy, zejména pokud jsou v záloze velké zprávy.

Scénáře selhání struktury

Scénáře

Je-li pro strukturu prostředku CF ohlášeno selhání, závisí akce prováděná připojenými správci front na následujících okolnostech:

- Typ selhání nahlášeného komponentou XES z/OS do IBM MQ.
- Typ struktury (aplikace nebo administrace)
- Úroveň správce front
- Úroveň CFLEVEL objektu IBM MQ CFSTRUCT (2, 3, 4 nebo 5). Toto není úroveň CFLEVEL mikrokódu CFCC)
- Atribut RECAUTO objektu IBM MQ CFSTRUCT na úrovni CFLEVEL (5)

Následující scénáře popisují, co se stane, když je hlášena porucha pro administrativní strukturu:

- Je-li pro administrativní strukturu přijata událost selhání struktury, dojde k opětovnému přidělení struktury a k automatickému opětovnému sestavení bez ukončení správce front. Nová instance struktury je přidělena XES, když se k ní správce front pokusí připojit.

Po připojení správce front k nové instanci struktury zapíše správce front položky do struktury. Toto zpracování provádí správce front a není součástí zpracování znovusestavení XES.

Pokud nebyl správce front v době selhání spuštěn nebo byl ukončen před dokončením obnovy jeho části administrativní struktury, mohou být jeho položky administrativní struktury znovu sestaveny jiným správcem front ve skupině sdílení front.

Položky administrační struktury správce front mohou být znovu sestaveny pouze jiným správcem front spuštěným na stejné nebo vyšší úrovni. Pokud položky administrační struktury správce front nemohou být znovu sestaveny jiným správcem front ve skupině sdílení front, restartujte správce front, aby mohl dokončit znovusestavení své části struktury.

Určité akce jsou pozastaveny, dokud nebudou znovu sestaveny položky administrativní struktury pro všechny správce front. Mezi pozastavené akce patří následující:

- Otevírání a zavírání sdílených front.
- Potvrzení nebo zálohování jednotek zotavení.
- Serializované aplikace, které se připojují ke správci front nebo se od něj odpojují.
- Zálohování nebo obnova struktury aplikace.

Všechny serializované aplikace, které jsou již připojeny ke správci front, mohou pokračovat ve zpracování. Všechny serializované aplikace, které se pokusí připojit pomocí parametrů MQCNO_SERIALIZE_CONN_TAG_QSG nebo MQCNO_RESTRICT_CONN_TAG_QSG, obdrží návratový kód MQRC_CONN_TAG_NOT_USABLE.

Po opětovném sestavení položek administrativní struktury pro správce front jsou pozastavené akce obnoveny.

Následující scénáře popisují, co se stane, když je pro strukturu aplikace nahlášeno selhání:

- Pokud je pro strukturu aplikace přijata událost selhání struktury a úroveň CFLEVEL je 1 nebo 2, správce front se ukončí. Restartujte správce front. První správce front, který se znovu pokusí připojit ke struktuře, způsobí, že XES přidělí novou instanci struktury.
- Pokud je pro strukturu aplikace přijata událost selhání struktury a úroveň CFLEVEL je 3, 4 nebo 5, správci front připojení ke struktuře budou pokračovat. Aplikace, které nepoužívají fronty ve struktuře, která selhala, mohou pokračovat v normálním zpracování.

Avšak aplikace, které se pokoušejí o operace ve frontách ve struktuře, která selhala, obdrží chybu MQRC_CF_STRUC_FAILED, dokud nebude struktura úspěšně znovu sestavena, což znamená, že aplikace může znovu otevřít fronty.

Znovusestavení struktury se spustí automaticky pro struktury aplikace CFLEVEL (5) definované s volbou RECAUTO (YES). Jinak bude struktura znovu sestavena při zadání příkazu RECOVER CFSTRUCT.

Odolnost vůči selháním konektivity prostředku Coupling Facility

Jaká je odolnost vůči selháním konektivity prostředku Coupling Facility?

Odolnost vůči selháním konektivity prostředku Coupling Facility odkazuje na schopnost správců front ve skupině sdílení front tolerovat ztrátu konektivity ke struktuře prostředku Coupling Facility bez ukončení. Tato funkce se také pokusí znovu sestavit strukturu v jiném prostředku Coupling Facility s lepší konektivitou, aby se co nejdříve získal přístup ke sdíleným frontám.

Co je částečná ztráta konektivity?

IBM MQ definuje částečnou ztrátu konektivity jako situaci, kdy jeden nebo více systémů v prostředí sysplex ztratí konektivitu k prostředku Coupling Facility, kde je přidělena struktura, ke které systém přistupuje, ale alespoň jeden systém v prostředí sysplex udržuje konektivitu ke stejnému prostředku Coupling Facility.

Co je to naprostá ztráta konektivity?

IBM MQ definuje celkovou ztrátu konektivity jako situaci, kdy žádné systémy v prostředí sysplex nemají konektivitu k prostředku Coupling Facility a strukturu v něm přidělenou.

Proč byste tuto funkci povolovali?

Odolnost vůči selháním konektivity prostředku Coupling Facility zlepšuje dostupnost produktu IBM MQa umožňuje, aby nesdílené fronty zůstaly k dispozici i po ztrátě konektivity správce front k jedné nebo více strukturám prostředku Coupling Facility. Správci front, kteří ztratí konektivitu ke struktuře prostředku Coupling Facility, se navíc automaticky pokusí znovu sestavit strukturu v jiném dostupném prostředku Coupling Facility, čímžlepší dostupnost sdílených front v rámci skupiny sdílení front.

Aspekty při povolování této funkce

Správce front, který toleruje ztrátu připojitelnosti ke strukturám prostředku Coupling Facility bez ukončení, nemusí být schopen se po určitou dobu znovu připojit ke struktuře prostředku Coupling Facility, pokud není k dispozici alternativní prostředek Coupling Facility. Sdílené fronty definované ve

strukturu, u které došlo ke ztrátě konektivity, zůstávají nedostupné, dokud se neobnoví konektivita ke struktuře. V této situaci mohou aplikace, které se připojují ke členům skupiny sdílení front za účelem provádění práce se sdílenými frontami, zjistit, že sdílené fronty, ke kterým potřebují přístup, nejsou k dispozici. Chcete-li se této situaci vyhnout, doporučuje se, aby správci front byli konfigurováni tak, aby se ukončili v případě, že dojde ke ztrátě konektivity ke struktuře prostředku Coupling Facility. Toto ukončení nutí aplikace připojit se k jinému členu skupiny sdílení front, který má konektivitu ke strukturám prostředku Coupling Facility, kde jsou definovány sdílené fronty, které aplikace vyžaduje.

Správa odolnosti vůči selhání konektivity prostředku Coupling Facility

Jak mohu tuto funkci povolit?

Chcete-li umožnit odolnost vůči konektivě prostředku Coupling Facility, je třeba provést následující kroky.

1. Ujistěte se, že datová sada dvojice CFRM byla formátována tak, aby podporovala znovusestavení spravované systémem. To umožňuje správcům front zahájit znovusestavení spravované systémem za účelem opětovného vytvoření struktury v dostupném prostředku Coupling Facility. Pomocí příkazu **DISPLAY XCF, COUPLE, TYPE=CFRM** určete formát datové sady páru CFRM. Chcete-li podporovat znovusestavení spravované systémem, měla by být datová sada páru CFRM formátována zadáním:

```
"ITEM NAME(SMREBLD) NUMBER(1) "
```

Další informace o formátování datové sady páru CFRM naleznete v dokumentaci k tématu [z/OS MVS Nastavení prostředí sysplex](#).

2. Ujistěte se, že je k dispozici alternativní prostředek Coupling Facility, který je uveden v seznamu předvoleb CFRM pro všechny struktury prostředku Coupling Facility systému IBM MQ. To umožňuje správcům front, aby se co nejdříve pokusili znovu sestavit struktury do alternativního dostupného prostředku Coupling Facility pro obnovení přístupu ke strukturám.

Struktury IBM MQ musí být definovány s ENFORCEORDER (NO) v zásadě CFRM, aby mohl XCF zvolit optimální prostředek CF v konfiguraci, pokud IBM MQ potřebuje znovu přidělit strukturu.

Další informace o seznamech předvoleb struktury naleznete v dokumentaci [z/OS MVS Nastavení prostředí sysplex](#).
3. Změňte všechny struktury prostředku Coupling Facility aplikace, které musí tolerovat ztrátu konektivity na úrovni CFLEVEL (5). Jedná se o minimální úroveň, která může tolerovat ztrátu konektivity.
4. Určete hodnoty požadované pro atributy **QMGR CFCONLOS** a **CFSTRUCT CFCONLOS** a odpovídajícím způsobem je změňte. Atribut **QMGR CFCONLOS** řídí, zda je tolerována ztráta konektivity ke struktuře administrace, a atribut **CFSTRUCT CFCONLOS** řídí, zda je ztráta konektivity tolerována každou strukturou prostředku Coupling Facility aplikace. Jsou-li zachovány výchozí hodnoty těchto atributů, správce front bude ukončen po ztrátě konektivity k libovolné struktuře prostředku Coupling Facility.
5. Určete hodnoty vyžadované pro atribut **CFSTRUCT RECAUTO** pro každou strukturu prostředku Coupling Facility aplikace a odpovídajícím způsobem je změňte. Tento atribut řídí, zda by měly být struktury prostředku Coupling Facility automaticky obnoveny pomocí protokolovaných dat po celkové ztrátě konektivity. Je-li pro tento atribut zachována výchozí hodnota, neprovede se po celkové ztrátě konektivity pro struktury aplikace žádná automatická obnova.

Scénář 1-Ztráta konektivity ke struktuře administrace

Správci front mohou tolerovat ztrátu konektivity ke struktuře administrace bez ukončení.

Dojde-li ke ztrátě připojitelnosti ke struktuře administrace správcem front, který byl konfigurován tak, aby toleroval ztrátu připojitelnosti ke struktuře administrace, odpojí se všichni členové skupiny sdílení front od struktury administrace. Všichni aktivní správci front ve skupině sdílení front se poté pokusí znovu připojit ke struktuře administrace, což způsobí, že budou znovu umístěni v prostředku Coupling

Facility s nejlepší konektivitou pro všechny systémy v prostředí sysplex, a znovu sestaví data struktury administrace.

Poznámka: Nemusí se nutně jednat o prostředek CF, který má nejlepší konektivitu ke všem systémům s aktivními správci front.

Pokud se správce front nemůže znovu připojit ke struktuře administrace, například proto, že není k dispozici žádný z prostředků CF v seznamu předvoleb CFRM pro danou strukturu administrace, některé operace sdílených front zůstanou nedostupné, dokud se správce front nebude moci úspěšně znovu připojit ke struktuře administrace a znovu sestavit data struktury administrace. K opětovnému připojení dochází automaticky, když je v systému k dispozici vhodný prostředek CF.

Není tolerováno připojení ke struktuře administrace během spouštění správce front v důsledku nedostatku konektivity k prostředku Coupling Facility nebo není k dispozici žádný vhodný prostředek Coupling Facility pro přidělení struktury. Všichni aktivní správci front ve skupině sdílení front se poté pokusí znovu připojit ke struktuře administrace, což způsobí, že budou znovu umístěni v jiném prostředku Coupling Facility, je-li k dispozici, a znovu sestaví data struktury administrace.

Scénář 2-Ztráta konektivity ke struktuře aplikace

Ztrátu konektivity ke strukturám aplikací v systému **CFLEVEL (5)** nebo vyšším lze tolerovat bez ukončení správce front. Správci front připojení ke strukturám aplikací v systému **CFLEVEL (4)** nebo nižší nebo ke strukturám v systému **CFLEVEL (5)**, které nebyly nakonfigurovány tak, aby tolerovaly ztrátu konektivity, při ztrátě konektivity ke struktuře nemusejí být ukončeny kódem příčiny 00C510AB.

Dojde-li ke ztrátě připojitelnosti ke struktuře aplikace, která byla konfigurována tak, aby tolerovala ztrátu připojitelnosti, odpojí se všichni správci front, kteří ztratili připojitelnost ke struktuře. Následné chování správce front závisí na tom, zda je ztráta konektivity částečná nebo úplná.

Částečná ztráta konektivity ke struktuře aplikace

Pokud je ztráta konektivity určena jako částečná, správci front, kteří ztratili konektivitu ke struktuře, se pokusí zahájit znovusestavení spravované systémem s cílem přesunout strukturu do jiného prostředku Coupling Facility s vylepšenou konektivitou. Pokud je toto znovusestavení úspěšné, jsou trvalé i dočasné zprávy ve struktuře zkopírovány do jiného prostředku Coupling Facility a přístup k frontám ve struktuře je obnoven. Správci front, kteří neztratili připojitelnost, zůstávají připojeni ke struktuře, avšak operace, které přistupují ke struktuře, jsou během procesu znovusestavení spravovaného systémem zpožděny.

Pokud strukturu aplikace nelze znovu sestavit na jiný prostředek Coupling Facility s vylepšenou konektivitou nebo pokud někteří správci front stále nemají konektivitu ke struktuře poté, co byla znovu sestavena v jiném prostředku Coupling Facility, zůstávají fronty definované ve struktuře nedostupné ve správcích front, kteří nemají konektivitu ke struktuře, dokud není konektivita k prostředku Coupling Facility obnovena. Správci front se automaticky znovu připojí ke struktuře, jakmile bude k dispozici, a obnoví se přístup ke sdíleným frontám definovaným ve struktuře.

Celková ztráta konektivity ke struktuře aplikace

Pokud všechny systémy MVS v prostředí sysplex ztratily konektivitu k prostředku Coupling Facility, v němž je přidělena struktura aplikace, produkt z/OS dealokuje strukturu z prostředku Coupling Facility při každém pokusu o opětovné připojení ke struktuře. Správce front se může pokusit o opětovné připojení ke struktuře z několika důvodů, jako je například pokus aplikace o otevření sdílené fronty nebo oznámení ze systému, že byly k dispozici nové prostředky prostředku Coupling Facility. Je proto pravděpodobné, že všechny dočasné zprávy v ovlivněné struktuře budou ztraceny po celkové ztrátě připojitelnosti ke struktuře aplikace.

Zotavitelné struktury aplikací jsou automaticky obnoveny po celkové ztrátě konektivity, pokud byly definovány s produktem **RECAUTO (YES)**. Obnova se spustí téměř okamžitě, pokud je k dispozici alternativní prostředek Coupling Facility pro přidělení struktury nebo kdykoli je takový prostředek Coupling Facility k dispozici. Pokud nebyla struktura definována s parametrem **RECAUTO (YES)**, lze obnovu spustit zadáním příkazu **RECOVER CFSTRUCT**. Tím se obnoví všechny trvalé zprávy ve struktuře, ale všechny dočasné zprávy budou ztraceny. Vzhledem k tomu, že tento proces zahrnuje čtení protokolu správce front, jeho dokončení může trvat delší dobu, a proto se doporučuje pravidelně

provádět zálohy struktury, aby se snížil čas potřebný k obnovení přístupu ke sdíleným frontám ve struktuře.

Správci front se pokusí o opětovné připojení k nezotavitelným strukturám aplikací, jakmile se aplikace pokusí otevřít sdílenou frontu, která je ve struktuře definována, nebo je ze systému obdrženo oznámení o tom, že byly k dispozici nové prostředky prostředku Coupling Facility. Je-li k dispozici vhodný prostředek Coupling Facility pro přidělení struktury, je přidělena nová struktura a je obnoven přístup ke sdíleným frontám definovaným ve struktuře. Vzhledem k tomu, že trvalé zprávy nelze vkládat do front definovaných v nezotavitelných strukturách, budou všechny zprávy ve sdílených frontách ztraceny.

Provozní chování

Pokud správce front IBM WebSphere MQ 7.1 nebo novější, nakonfigurovaný tak, aby toleroval ztrátu konektivity ke konkrétní struktuře prostředku Coupling Facility, ztratí konektivitu, členové skupiny sdílení front se pokusí o automatické zotavení ze selhání a opětovné připojení ke struktuře. Tato aktivita může zahrnovat opětovné přidělení struktury v jiném prostředku Coupling Facility s lepší konektivitou, je-li k dispozici. K zotavení ze ztráty konektivity však může být zapotřebí zásah operátora.

Obvykle je požadovaná akce operátora:

1. Vyřešte příčinu selhání, které má za následek ztrátu konektivity.
2. Ujistěte se, že prostředek CF, kde lze přidělit struktury IBM MQ, je dostupný na všech systémech v prostředí sysplex

Všechny struktury, které byly automaticky znovu přiděleny v jiném prostředku Coupling Facility po ztrátě události konektivity, lze přesunout do prostředku Coupling Facility s optimální konektivitou pro všechny správce front ve skupině sdílení front. V případě potřeby to lze provést spuštěním příkazu znovusestavení spravovaného systémem **SETXCF START,REBUILD**, jak je uvedeno v části [z/OS MVS Popis příkazů systému](#).

V případě částečné ztráty konektivity ke struktuře aplikace se správci front, kteří ztratili konektivitu ke struktuře, pokusí zahájit znovusestavení spravované systémem. Tento proces alokuje strukturu v jiném prostředku Coupling Facility pouze v případě, že tento prostředek Coupling Facility má připojitelnost ke všem aktivním správcům front, kteří jsou k této struktuře aktuálně připojeni. Proto je možné, že v případě, že většina správců front ve skupině sdílení front ztratila konektivitu ke struktuře aplikace, nemohou tuto strukturu znovu sestavit do jiného prostředku Coupling Facility kvůli správcům front, kteří jsou stále připojeni k původní struktuře. V této situaci mohou být ukončeni správci front, kteří jsou stále připojeni k původní struktuře, aby bylo možné strukturu znovu sestavit, nebo může být vydán příkaz **RESET CFSTRUCT ACTION(FAIL)**, který strukturu selže. Obnovu lze zahájit na použitelných strukturách zadáním příkazu **RECOVER CFSTRUCT**.

Poznámka: Při selhání a obnově struktury jsou všechny dočasné zprávy ve struktuře ztraceny.

z/OS

Koncepce zabezpečení v produktu IBM MQ for z/OS

Prostřednictvím tohoto tématu můžete porozumět důležitosti zabezpečení produktu IBM MQ a důsledkům neexistence odpovídajícího nastavení zabezpečení ve vašem systému.

Proč musíte chránit prostředky IBM MQ

Produkt IBM MQ zpracovává přenos informací, které jsou potenciálně cenné. Použití zabezpečení zajišťuje, že prostředky, které IBM MQ vlastní a spravuje, jsou chráněny před neoprávněným přístupem. Takový přístup může vést ke ztrátě nebo zveřejnění informací.

Měli byste se ujistit, že žádný z následujících prostředků není zpřístupněn nebo změněn jakýmkoli neautorizovaným uživatelem nebo procesem:

- Připojení k produktu IBM MQ
- Objekty IBM MQ, jako jsou fronty, procesy a seznamy názvů.

- IBM MQ přenosové linky, tj. kanály IBM MQ
- IBM MQ řídicí příkazy systému
- Zprávy produktu IBM MQ
- Informace o kontextu přidružené ke zprávám

K zajištění nezbytného zabezpečení produkt IBM MQ používá prostředek SAF (System Authorization Facility) systému z/OS ke směrování požadavků na autorizaci na externí správce zabezpečení (ESM), například na server zabezpečení (dříve známý jako RACF). Produkt IBM MQ neprovádí žádné vlastní ověření zabezpečení. V případě použití distribuovaných front nebo klientů můžete požadovat další bezpečnostní opatření, pro která produkt IBM MQ poskytuje záznamy ověřování kanálu, uživatelské procedury kanálu, atribut kanálu MCAUSER a protokol TLS.

Rozhodnutí o povolení přístupu k objektu je učiněno modulem ESM a IBM MQ následuje po tomto rozhodnutí. Pokud ESM nemůže provést rozhodnutí, produkt IBM MQ zabrání přístupu k objektu.

Co se stane, když neochráníte prostředky IBM MQ

Pokud se zabezpečením nic neděláte, nejpravděpodobnější efekt je, že *všichni* uživatelé mohou přistupovat a měnit *každý* prostředek. To zahrnuje nejen lokální uživatele, ale také uživatele na vzdálených systémech, které používají distribuované fronty nebo klienty, kde mohou být řízení zabezpečení přihlášení méně přísná, než je obvyklé v případě produktu z/OS.

Chcete-li povolit kontrolu zabezpečení, musíte provést následující:

- Nainstalujte a aktivujte ESM (například Security Server).
- Definujte třídu MQADMIN, pokud používáte jiný ESM než Security Server.
- Aktivujte třídu MQADMIN.

Je třeba zvážit, zda by použití názvů prostředků s různými písmeny bylo výhodné pro váš podnik. Pokud ve svých profilech ESM používáte názvy prostředků s různými písmeny, musíte definovat a aktivovat třídu MXADMIN.

z/OS Šifrování datové sady

Šifrování datové sady (DSE) poskytuje schopnost šifrovat datové sady z/OS, takže data, která obsahují, mohou zobrazit nebo upravit pouze ID uživatelů, kterým bylo uděleno specifické oprávnění. To poskytuje šifrování dat v klidu v systému souborů a zabraňuje neúmyslnému zveřejnění citlivých informací uživatelům, kteří mají legitimní obchodní potřebu a oprávnění spravovat datové sady sami.

Před verzí IBM MQ for z/OS 9.1.4 produkt IBM MQ for z/OS nepodporuje použití DSE s aktivními protokoly, sadami stránek a sdílenými datovými sadami zpráv (SMDS), které poskytují primární mechanismy persistence pro zprávy produktu IBM MQ.

Místo toho produkt Advanced Message Security poskytuje komplexní řešení šifrování pro systém zpráv IBM MQ, které zahrnuje celou síť IBM MQ, šifrování dat za běhu, v klidu a dokonce i v běhových procesech IBM MQ.

Ostatní VSAM a sekvenční datové sady používané v subsystému IBM MQ lze šifrovat pomocí DSE. Příklad:

- zaváděcí datová sada
- Sekvenční soubory obsahující příkazy konfigurace systému (MQSC) načtené při spuštění pomocí CSQINPx DDNAMEs
- IBM MQ archivních protokolů, často používaných pro dlouhodobou archivaci dat protokolu IBM MQ pro účely auditu.

Pomocí DSE můžete šifrovat přidělením datové třídy, která je definována s popiskem klíče datové sady. Další informace viz [Plánování archivního úložiště protokolu](#).

From IBM MQ for z/OS 9.1.4, IBM MQ for z/OS supports use of DSE with the active logs and page sets in addition to the support provided in earlier releases.

Produkt IBM MQ for z/OS nepodporuje použití DSE pro sdílené datové sady zpráv (SMDS).

Viz část důvěrnost pro data v klidu na systému IBM MQ for z/OS se šifrováním datové sady. Další informace viz.

Související pojmy

[Koncepce zabezpečení](#)

[Záznamy ověření kanálu](#)

[Oprávnění pro práci s objekty IBM MQ na systému z/OS](#)

[Šifrovací bezpečnostní protokoly: TLS](#)

Související úlohy

[Nastavení zabezpečení na systému z/OS](#)

[Porovnání zabezpečení na úrovni propojení a zabezpečení na úrovni aplikace](#)

Související odkazy

[Zprávy pro IBM MQ for z/OS](#)

z/OS Ovládací prvky zabezpečení a volby v produktu IBM MQ for z/OS

Můžete určit, zda je zapnuto zabezpečení pro celý subsystém IBM MQ a zda chcete provádět kontroly zabezpečení na úrovni správce front nebo skupiny sdílení front. Můžete také řídit počet ID uživatelů kontrolovaných pro zabezpečení prostředků rozhraní API.

Zabezpečení subsystému

Zabezpečení subsystému je ovládací prvek, který určuje, zda je pro celého správce front prováděna kontrola zabezpečení. Pokud nevyžadujete kontrolu zabezpečení (například v testovacím systému) nebo pokud jste spokojeni s úrovní zabezpečení všech prostředků, které se mohou připojit k produktu IBM MQ (včetně klientů a kanálů), můžete vypnout kontrolu zabezpečení pro správce front nebo skupinu sdílení front, aby se neprováděla žádná další kontrola zabezpečení.

Jedná se o jedinou kontrolu, která může zcela vypnout zabezpečení a určit, zda jsou provedeny další kontroly zabezpečení. To znamená, že pokud vypnete kontrolu pro správce front nebo skupinu sdílení front, neprovede se žádná jiná kontrola IBM MQ ; pokud ji ponecháte zapnutou, produkt IBM MQ zkontroluje vaše požadavky na zabezpečení pro ostatní prostředky IBM MQ .

Můžete také zapnout nebo vypnout zabezpečení pro konkrétní sady prostředků, jako jsou příkazy.

Kontrola úrovně správce front nebo skupiny sdílení front

Zabezpečení můžete implementovat na úrovni správce front nebo na úrovni skupiny sdílení front. Pokud implementujete zabezpečení na úrovni skupiny sdílení front, budou všichni správci front ve skupině sdílet stejné profily. To znamená, že existuje méně profilů, které je třeba definovat a udržovat, což usnadňuje správu zabezpečení. Usnadňuje také přidání nového správce front do skupiny sdílení front, protože dědí existující profily zabezpečení.

Je také možné implementovat kombinaci obojího, pokud to vaše instalace vyžaduje, například během migrace, nebo pokud máte ve skupině sdílení front jednoho správce front, který vyžaduje různé úrovně zabezpečení pro ostatní správce front ve skupině.

Zabezpečení na úrovni skupiny sdílení front

Kontrola zabezpečení na úrovni skupiny sdílení front se provádí pro celou skupinu sdílení front. Umožňuje vám zjednodušit administraci zabezpečení, protože vyžaduje, abyste definovali méně profilů zabezpečení. Autorizace ID uživatele pro použití konkrétního prostředku je zpracována na úrovni skupiny sdílení front a je nezávislá na tom, který správce front toto ID uživatele používá pro přístup k prostředku.

Řekněme například, že serverová aplikace je spuštěna pod ID uživatele SERVER a chce přístup do fronty s názvem SERVER.REQUESTa chcete spustit instanci SERVER na každém obrazu z/OS v prostředí sysplex. Místo toho, aby bylo serveru umožněno otevřít server SERVER.REQUEST na každého správce front jednotlivě (zabezpečení na úrovni správce front), můžete povolit přístup pouze na úrovni skupiny sdílení front.

Profily zabezpečení na úrovni skupiny sdílení front lze použít k ochraně všech typů prostředků, ať už lokálních nebo sdílených.

zabezpečení na úrovni správce front

Profily zabezpečení na úrovni správce front můžete použít k ochraně všech typů prostředků, ať už lokálních nebo sdílených.

Kombinace obou úrovní

Můžete použít kombinaci zabezpečení na úrovni správce front i skupiny sdílení front.

Můžete přepsat nastavení zabezpečení na úrovni skupiny sdílení front pro konkrétního správce front, který je členem této skupiny. To znamená, že můžete provádět různé úrovně kontrol zabezpečení pro jednotlivé správce front oproti těm, které jsou prováděny pro ostatní správce front ve skupině.

Další informace naleznete v tématu [Profily pro řízení skupiny sdílení front nebo zabezpečení na úrovni správce front](#).

Řízení počtu kontrolovaných ID uživatelů

RESLEVEL je profil serveru zabezpečení, který řídí počet ID uživatelů kontrolovaných pro zabezpečení prostředků IBM MQ . Za normálních okolností, když se uživatel pokusí přistoupit k prostředku IBM MQ , server zabezpečení zkontroluje příslušné ID uživatele nebo ID, aby zjistil, zda je přístup k tomuto prostředku povolen. Definováním profilu RESLEVEL můžete řídit, zda se mají kontrolovat nula, jedno nebo případně dvě ID uživatelů.

Tyto ovládací prvky jsou prováděny na základě připojení a poslední po celou dobu trvání připojení.

Pro každého správce front existuje pouze jeden profil RESLEVEL. Řízení je implementováno přístupem, který má ID uživatele k tomuto profilu.

Třídy IBM MQ RACF s malými i velkými písmeny

Nyní můžete použít podporu profilů RACF se smíšenými velkými písmeny, která vám umožňuje používat názvy prostředků se smíšenými velkými a velkými písmeny a definovat profily produktu IBM MQ RACF k jejich ochraně.

Můžete zvolit jednu z následujících možností:

- Pokračujte v používání pouze velkých písmen IBM MQ RACF Třídy jako v předchozích verzích, nebo
- Použijte nové třídy IBM MQ RACF se smíšenými písmeny.

Bez použití profilů RACF se smíšenou velikostí písmen můžete i nadále používat názvy prostředků se smíšenou velikostí písmen v produktu IBM MQ for z/OS ; Tyto názvy prostředků však mohou být chráněny pouze generickými profily RACF ve třídách IBM MQ psaných velkými písmeny. Používáte-li podporu profilů IBM MQ RACF se smíšenými písmeny, můžete poskytnout podrobnější úroveň ochrany definováním profilů IBM MQ RACF ve třídách IBM MQ se smíšenými velkými a velkými písmeny.

z/OS Prostředky, které můžete chránit v produktu IBM MQ for z/OS

Při spuštění správce front nebo při zadání příkazu operátora produkt IBM MQ for z/OS určí, které prostředky chcete chránit.

Můžete určit, které kontroly zabezpečení mají být prováděny pro jednotlivé správce front. Můžete například implementovat řadu kontrol zabezpečení pro produkčního správce front, ale žádný pro testovacího správce front.

Zabezpečení připojení

Kontrola zabezpečení připojení se provádí buď při pokusu aplikačního programu o připojení ke správci front. Provádí se tak, že zadáte požadavek MQCONN nebo MQCONNX , nebo když iniciátor kanálu nebo adaptér CICS nebo IMS vydá požadavek na připojení.

Používáte-li zabezpečení na úrovni správce front, můžete vypnout kontrolu zabezpečení připojení pro konkrétního správce front. Provedete-li to však, může se k tomuto správci front připojit libovolný uživatel.

Pro adaptér CICS se pro kontrolu zabezpečení připojení používá pouze ID uživatele adresního prostoru CICS , nikoli individuální ID uživatele terminálu CICS . V případě adaptéru IMS se při připojení ovládacího prvku IMS nebo závislých oblastí k produktu IBM MQ kontroluje ID uživatele adresního prostoru IMS . Pro inicializátor kanálu je kontrolováno ID uživatele použité adresním prostorem inicializátoru kanálu.

Kontrolu zabezpečení připojení můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

Zabezpečení příkazů

Kontrola zabezpečení příkazu se provádí, když uživatel zadá příkaz MQSC z libovolného zdroje popsaného v části Zdroje, ze kterého můžete zadat příkazy MQSC a PCF v systému IBM MQ for z/OS. Můžete provést oddělenou kontrolu prostředku určeného příkazem, jak je popsáno v tématu [“Zabezpečení prostředků příkazu”](#) na stránce 264.

Pokud vypnete kontrolu příkazů, vydavatelé příkazů nebudou kontrolováni, aby zjistili, zda mají oprávnění k vydání příkazu.

Pokud jsou příkazy MQSC zadány z konzoly, musí mít konzola atribut oprávnění z/OS SYS. Příkazy vydané z datových sad CSQINP1 nebo CSQINP2 nebo interně správcem front jsou osvobozeny od všech kontrol zabezpečení, zatímco příkazy pro CSQINPX používají ID uživatele adresního prostoru inicializátoru kanálu. Musíte řídit, kdo může aktualizovat tyto datové sady prostřednictvím normální ochrany datových sad.

Kontrolu zabezpečení příkazů můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

Zabezpečení prostředků příkazu

Některé příkazy MQSC, například definování lokální fronty, zahrnují manipulaci s prostředky IBM MQ . Když je zabezpečení prostředků příkazu aktivní, při každém zadání příkazu zahrnujícího prostředek produkt IBM MQ zkontroluje, zda má uživatel povoleno změnit definici tohoto prostředku.

Zabezpečení prostředků příkazů můžete použít k vynucení standardů pojmenování. Administrátor mezd může mít například povoleno odstranit a definovat pouze fronty s názvy začínajícími na "PAYROLL". Je-li zabezpečení prostředků příkazu neaktivní, neprovádějí se žádné kontroly zabezpečení prostředku, se kterým je příkaz manipulován. Nezaměňujte zabezpečení prostředků příkazů se zabezpečením příkazů; tyto dva jsou nezávislé.

Vypnutí kontroly zabezpečení prostředků příkazů nemá vliv na kontrolu prostředků prováděnou specificky pro jiné typy zpracování, které nezahrnují příkazy.

Kontrolu zabezpečení prostředků příkazů můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

Aspekty zabezpečení kanálu

Zabezpečení kanálu

Používáte-li kanály, dostupné funkce zabezpečení závisí na tom, který komunikační protokol budete používat. Pokud používáte protokol TCP, nejsou k dispozici žádné funkce zabezpečení poskytované

s komunikačním protokolem, i když můžete použít protokol TLS. Používáte-li APPC, můžete informace o ID uživatele proudit z odesílajícího agenta MCA přes síť do cílového agenta MCA pro ověření.

Pro oba protokoly můžete určit, která ID uživatelů chcete kontrolovat pro účely zabezpečení, a kolik. Opět platí, že volby, které máte k dispozici, závisí na tom, který protokol používáte, na tom, co určíte při definování kanálu a na nastavení RESLEVEL pro inicializátor kanálu.

Další informace o typech zabezpečení kanálu, které jsou k dispozici, naleznete v části [Záznamy ověření kanálu a Přehled uživatelské procedury zabezpečení](#).

Související odkazy

“Zabezpečení prostředků rozhraní API v adresáři IBM MQ for z/OS” na stránce 265

Prostředky se kontrolují, když aplikace otevře objekt s voláním MQOPEN nebo MQPUT1. Přístup potřebný k otevření objektu závisí na tom, jaké volby otevření jsou určeny při otevření fronty.

Zabezpečení prostředků rozhraní API v adresáři IBM MQ for z/OS

Prostředky se kontrolují, když aplikace otevře objekt s voláním MQOPEN nebo MQPUT1. Přístup potřebný k otevření objektu závisí na tom, jaké volby otevření jsou určeny při otevření fronty.

Zabezpečení prostředků rozhraní API je rozděleno do následujících kontrol:

- [Fronta](#)
- [process, proces](#)
- [Namelist, Seznam názvů](#)
- [Alternativní uživatel](#)
- [Kontext](#)

Při otvírání objektu správce front nebo při přístupu k objektům paměťových tříd nejsou prováděny žádné kontroly zabezpečení.

Fronta

Kontrola zabezpečení fronty řídí, kdo může otevřít kterou frontu, a jaké volby ji mohou otevřít. Uživateli může být například povoleno otevřít frontu s názvem PAYROLL.INCREASE.SALARY projděte zprávy ve frontě (pomocí volby MQOO_BROWSE), ale neodebírejte zprávy z fronty (pomocí jedné z voleb MQOO_INPUT_*). Pokud vypnete kontrolu front, může kterýkoli uživatel otevřít libovolnou frontu s platnou volbou otevření (tj. s jakoukoli platnou volbou MQOO_* ve volání MQOPEN nebo MQPUT1).

Kontrolu zabezpečení fronty můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

Proces

Kontrola zabezpečení procesu se provádí, když uživatel otevře objekt definice procesu. Pokud vypnete kontrolu procesů, může kterýkoli uživatel otevřít libovolný proces.

Kontrolu zabezpečení procesů můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

Seznam názvů

Kontrola zabezpečení seznamu názvů se provádí, když uživatel otevře seznam názvů. Pokud vypnete kontrolu seznamů názvů, může kterýkoli uživatel otevřít libovolný seznam názvů.

Kontrolu zabezpečení seznamu názvů můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

Alternativní uživatel

Alternativní zabezpečení uživatele řídí, zda může jedno ID uživatele použít oprávnění jiného ID uživatele k otevření objektu IBM MQ .

Příklad:

- Program serveru spuštěný pod ID uživatele PAYSERV načte zprávu požadavku z fronty, která byla vložena do fronty podle ID uživatele USER1.
- Když program serveru obdrží zprávu požadavku, zpracuje požadavek a vloží odpověď zpět do fronty pro odpověď uvedené se zprávou požadavku.
- Místo použití vlastního ID uživatele (PAYSERV) k autorizaci otevření fronty pro odpověď může server zadat jiné ID uživatele, v tomto případě USER1. V tomto příkladu by alternativní zabezpečení uživatele řídilo, zda je ID uživatele PAYSERV povoleno uvést ID uživatele USER1 jako alternativní ID uživatele při otevírání fronty pro odpovědi.

Alternativní ID uživatele je určeno v poli *AlternateUserId* deskriptoru objektu (MQOD).

Na libovolném objektu IBM MQ můžete použít alternativní ID uživatelů, například procesy nebo seznamy názvů. Nemá vliv na ID uživatele použité jinými správci prostředků, například pro zabezpečení CICS nebo pro zabezpečení datové sady z/OS .

Není-li aktivní alternativní zabezpečení uživatele, může jako alternativní ID uživatele použít jakékoli jiné ID uživatele.

Můžete zapnout nebo vypnout alternativní kontrolu zabezpečení uživatele na úrovni správce front nebo skupiny sdílení front.

Kontext

Kontext je informace, která je použitelná pro konkrétní zprávu a je obsažena v deskriptoru zprávy (MQMD), který je součástí zprávy. Informace o kontextu jsou k dispozici ve dvou sekcích:

Sekce identity

Uživatel aplikace, který nejprve vložil zprávu do fronty. Skládá se z následujících polí:

- *UserIdentifier*
- *AccountingToken*
- *ApplIdentityData*

Oddíl původu

Aplikace, která vložila zprávu do fronty, kde je momentálně uložena. Skládá se z následujících polí:

- *PutApplType*
- *PutApplName*
- *PutDate*
- *PutTime*
- *ApplOriginData*

Aplikace mohou určit data kontextu při volání MQPUT nebo MQPUT1 . Aplikace může generovat data, data mohou být předána z jiné zprávy nebo může správce front standardně generovat data. Například serverové programy mohou používat data kontextu ke kontrole identity žadatele, tj. zda tato zpráva pochází ze správné aplikace? Pole *UserIdentifier* se obvykle používá k určení ID uživatele alternativního uživatele.

Pomocí zabezpečení kontextu můžete řídit, zda může uživatel zadat libovolnou z voleb kontextu pro libovolné volání MQOPEN nebo MQPUT . Informace o volbách kontextu viz [Volby MQOPEN související s kontextem zprávy](#). Popisy polí deskriptoru zpráv souvisejících s kontextem viz [MQMD-Message descriptor](#)

Pokud vypnete kontrolu zabezpečení kontextu, může kterýkoli uživatel použít libovolnou volbu kontextu, kterou zabezpečení fronty umožňuje.

Kontrolu zabezpečení kontextu můžete zapnout nebo vypnout na úrovni fronty, správce front nebo skupiny sdílení front.

Produkt IBM MQ for z/OS má mnoho funkcí pro vysokou dostupnost. Toto téma popisuje některé aspekty dostupnosti.

Několik funkcí produktu IBM MQ může zvýšit dostupnost systému v případě, že dojde k selhání správce front nebo inicializátoru kanálu. Další informace o těchto funkcích naleznete v následujících sekcích:

- [Aspekty prostředí sysplex](#)
- [Sdílené fronty](#)
- [Sdílené kanály](#)
- [IBM MQ dostupnost sítě](#)
- [Použití z/OS správce ARM \(Automatic Restart Manager\)](#)
- [Použití modulu z/OS Extended Recovery Facility \(XRF\)](#)
- [Použití atributu z/OS GROUPUR pro obnovu ve skupině sdílení front](#)
- [Kde najít další informace o dostupnosti](#)

Aspekty prostředí sysplex

V prostředí *sysplex* spolupracuje řada obrazů operačního systému z/OS na jednom obrazu systému a komunikuje pomocí prostředku Coupling Facility. Produkt IBM MQ může využívat zařízení prostředí *sysplex* pro rozšířenou dostupnost.

Odebrání afinit mezi správcem front a konkrétním obrazem z/OS umožňuje restartovat správce front v jiném obrazu produktu z/OS v případě selhání obrazu. Mechanismus restartování může být ruční, použít ARM nebo použít automatizaci systému, pokud zajistíte následující:

- Na sdílených svazcích jsou definovány všechny sady stránek, protokoly, datové sady samozavedení, knihovny kódu a datové sady konfigurace správce front.
- Definice subsystému má rozsah prostředí *sysplex* a jedinečný název v prostředí *sysplex*.
- Úroveň *raného kódu* instalovaného na každém obrazu z/OS v době IPL je na stejné úrovni.
- Virtuální adresy IP protokolu TCP (VIPA) jsou k dispozici v každém zásobníku TCP v prostředí *sysplex* a nakonfigurovali jste moduly listener protokolu TCP systému IBM MQ a příchozí připojení tak, aby používaly spíše VIPA než výchozí názvy hostitelů.

Další informace o použití protokolu TCP v prostředí *sysplex* viz *TCP/IP v prostředí sysplex*, SG24-5235, příručka IBM Redbooks .

Dále můžete nakonfigurovat více správců front spuštěných v různých obrazech operačního systému v prostředí *sysplex* tak, aby fungovaly jako skupina sdílení front, která může využívat výhod sdílených front a sdílených kanálů pro vyšší dostupnost a vyrovnávání pracovní zátěže.

Sdílené fronty

V prostředí skupiny sdílení front se může aplikace připojit k libovolnému správci front v rámci skupiny sdílení front. Vzhledem k tomu, že všichni správci front ve skupině sdílení front mohou přistupovat ke stejné sadě sdílených front, aplikace nezávisí na dostupnosti konkrétního správce front; kterýkoli správce front ve skupině sdílení front může obsluhovat libovolnou frontu. To poskytuje větší dostupnost, pokud se správce front zastaví, protože všichni ostatní správci front ve skupině sdílení front mohou pokračovat ve zpracování fronty. Informace o vysoké dostupnosti sdílených front naleznete v tématu [“Výhody používání sdílených front”](#) na stránce 180.

V zájmu dalšího rozšíření dostupnosti zpráv ve skupině sdílení front produkt IBM MQ zjistí, zda se jiný správce front v dané skupině nestandardně odpojuje od prostředku Coupling Facility, a pokud je to možné, dokončí jednotky práce pro tohoto správce front, které stále čekají na vyřízení. Nazývá se *obnova typu peer* a je popsána v části [“Obnova typu peer”](#) na stránce 254.

Obnova typu peer nemůže obnovit pracovní jednotky, které byly v době selhání v nejistém stavu. Pomocí správce ARM (Automatic Restart Manager) můžete restartovat všechny systémy zahrnuté do selhání (například CICS, Db2a IBM MQ) a zajistit, aby byly všechny restartovány na stejném novém procesoru. To znamená, že mohou znovu synchronizovat a poskytují rychlou obnovu nejistých pracovních jednotek. To je popsáno v tématu [“Použití správce automatického restartu z/OS \(ARM\)”](#) na stránce 268.

Sdílené kanály

V prostředí skupiny sdílení front poskytuje produkt IBM MQ funkce, které zajišťují vysokou dostupnost pro síť. Inicializátor kanálu vám umožňuje používat síťové produkty, které vyvažují síťové požadavky v rámci sady vhodných serverů a skrývají selhání serverů v síti (například generické prostředky VTAM). Produkt IBM MQ používá pro příchozí požadavky generický port, aby bylo možné požadavky na připojení směřovat na libovolný dostupný inicializátor kanálu ve skupině sdílení front. To je popsáno v tématu [“Sdílené kanály”](#) na stránce 200.

Sdílené odchozí kanály přijímají zprávy, které odesílají ze sdílené přenosové fronty. Informace o stavu sdíleného kanálu jsou uchovávány na jednom místě pro celou úroveň skupiny sdílení front. To znamená, že kanál lze automaticky restartovat v jiném inicializátoru kanálu ve skupině sdílení front v případě, že dojde k selhání inicializátoru kanálu, správce front nebo komunikačního subsystému. Tento název se nazývá *obnova rovnocenných kanálů* a je popsán v části [Sdílené odchozí kanály](#).

IBM MQ dostupnost sítě

Zprávy IBM MQ jsou přenášeny ze správce front do správce front v síti IBM MQ pomocí kanálů. Můžete změnit konfiguraci na několika úrovních, abyste zlepšili dostupnost sítě správce front, a schopnost kanálu IBM MQ zjistit problém se sítí a znovu se připojit.

Protokol TCP *Keepalive* je k dispozici pro kanály TCP/IP. Způsobuje, že TCP odesílá pakety pravidelně mezi relacemi, aby zjistil selhání sítě. Atribut kanálu KAINTE určuje frekvenci těchto paketů pro kanál.

AdoptMCA umožňuje, aby kanál blokováno ve zpracování příjmu v důsledku výpadku sítě byl ukončen a nahrazen novým požadavkem na připojení. Modul *AdoptMCA* můžete řídit pomocí vlastnosti správce front *ADOPTMCA* s obslužným programem MQSC nebo pomocí vlastnosti *AdoptNewMCAType* s rozhraním Programmable Command Formats.

ReceiveTimeout zabraňuje trvalému zablokování kanálu ve volání přijetí sítě. Parametry iniciátoru kanálu RCVTIME a RCVTMIN určují charakteristiku časového limitu příjmu pro kanály jako funkci jejich intervalu prezenčního signálu. Další podrobnosti viz [Parametr správce front](#).

Použití správce automatického restartu z/OS (ARM)

Můžete použít IBM MQ for z/OS ve spojení se správcem automatického restartu z/OS (ARM). Pokud došlo k selhání správce front nebo inicializátoru kanálu, modul ARM jej restartuje na stejném obrazu z/OS. Pokud z/OS selže, selže také celá skupina souvisejících subsystémů a aplikací. ARM může automaticky restartovat všechny systémy, které selhaly, v předdefinovaném pořadí na jiném obrazu z/OS v prostředí sysplex. Toto se nazývá křížový restart systému.

Modul ARM umožňuje rychlé zotavení neověřených transakcí v prostředí sdílených front. Také poskytuje vyšší dostupnost, pokud nepoužíváte skupiny sdílení front.

Pomocí modulu ARM můžete restartovat správce front v jiném obrazu systému z/OS v prostředí sysplex v případě selhání systému z/OS.

Chcete-li povolit automatický restart, musíte provést následující:

1. Nastavte datovou sadu spojení ARM.
2. Definujte akce automatického restartování, které má produkt z/OS provést v *zásadě ARM*.
3. Spusťte zásadu ARM.

Chcete-li automaticky restartovat správce front v různých obrazech produktu z/OS , musí být každý správce front v každém obraze produktu z/OS , v němž může být tento správce front restartován, definován s jedinečným 4znakovým názvem subsystému prostředí sysplex.

Použití ARM s IBM MQ je popsáno v tématu [Použití ARM v IBM MQ síti](#).

Použití zařízení z/OS Extended Recovery Facility (XRF)

Produkt IBM MQ můžete použít v prostředí XRF (extended recovery facility). Všechny datové sady vlastněné produktem IBM MQ (spustitelný kód, BSDSs, protokoly a sady stránek) musí být na serveru DASD sdíleném mezi aktivním a alternativním procesorem XRF.

Používáte-li modul XRF pro obnovu, musíte zastavit správce front na aktivním procesoru a spustit jej na alternativním procesoru. V případě operačního systému CICS to můžete provést pomocí tabulky seznamu příkazů (CLT), kterou poskytuje CICS, nebo to může provést systémový operátor ručně. V případě operačního systému IMS se jedná o ruční operaci, kterou musíte provést po dokončení koordinačního systému IMS přepínače procesoru.

Obslužné programy systému IBM MQ musí být dokončeny nebo ukončeny, aby bylo možné přepnout správce front na alternativní procesor. Při plánování plánů obnovy XRF pečlivě zvažte účinek tohoto potenciálního přerušení.

Před ukončením činnosti správce front v aktivním procesoru je třeba zabránit spuštění správce front v alternativním procesoru. Předčasný start může způsobit závažné problémy s integritou dat, katalogu a protokolu. Použití globální serializace prostředků (GRS) pomáhá vyhnout se problémům s integritou tím, že zabráňuje souběžnému použití produktu IBM MQ na obou systémech. Musíte zahrnout BSDS jako chráněný prostředek a do kruhu GRS musíte zahrnout aktivní a alternativní procesory XRF.

Použití atributu z/OS GROUPUR pro obnovu ve skupině sdílení front

Skupiny sdílení front (QSG) umožňují další transakční prostředky, které jsou popsány v tomto tématu. Atribut GROUPUR umožňuje klientským aplikacím XA provádět jakékoli pochybné zotavení transakce, které může být vyžadováno, a to na libovolném členovi skupiny QSG.

Pokud se klientská aplikace XA připojí ke skupině sdílení front (QSG) prostřednictvím prostředí sysplex, nemůže zaručit, že kterému konkrétnímu správci front se připojí. Použití atributu GROUPUR správcem front v rámci skupiny sdílení front může povolit jakékoli zotavení transakcí, které může být nutné provést v kterémkoli členovi skupiny sdílení front. I v případě, že správce front, ke kterému byla aplikace původně připojena, není k dispozici, může dojít k zotavení transakcí.

Tato funkce uvolní klientskou aplikaci XA z jakékoli závislosti na specifických členech skupiny sdílení front, a tím rozšíří dostupnost správce front. Skupina sdílení front se transakční aplikaci zobrazuje jako jediná entita poskytující všechny funkce produktu IBM MQ a bez jediného bodu selhání správce front.

Tato funkce není pro transakční aplikaci zjevná.

Kde najít další informace o dostupnosti

Další informace o těchto tématech naleznete z následujících zdrojů:

<i>Tabulka 24. Kde najít další informace o dostupnosti</i>	
Téma	Kam se podívat
Skupiny sdílení front	“Sdílené fronty a skupiny sdílení front” na stránce 161
Parametry systému	Konfigurace parametrů systému

Tabulka 24. Kde najít další informace o dostupnosti (pokračování)

Téma	Kam se podívat
Použití správce automatického restartu Obslužné programy	Použití ARM v IBM MQ síti
Příkazy MQSC	Příkazy MQSC

z/OS

Monitorování a statistiky na IBM MQ for z/OS

Produkt IBM MQ for z/OS má sadu prostředků pro monitorování správce front a shromažďování statistických údajů.

Produkt IBM MQ poskytuje zařízení pro monitorování systému a shromažďování statistik. Další informace o těchto zařízeních naleznete v následujících sekcích:

- [“Monitorování online”](#) na stránce 270
- [“IBM MQ trasování”](#) na stránce 270
- [“Události”](#) na stránce 271

Monitorování online

Produkt IBM MQ zahrnuje následující příkazy pro monitorování stavu objektů IBM MQ :

- DISPLAY CHSTATUS zobrazuje stav určeného kanálu.
- DISPLAY QSTATUS zobrazuje stav uvedené fronty.
- DISPLAY CONN zobrazí stav zadaného připojení.

Další informace o těchto příkazech naleznete v tématu [Příkazy MQSC](#).

IBM MQ trasování

Produkt IBM MQ poskytuje prostředek trasování, který lze použít ke shromažďování následujících informací za běhu správce front:

Statistika výkonu

Trasování statistiky shromažďuje následující informace, které vám pomohou monitorovat výkon a vyladit systém:

- Počet různých požadavků MQI (statistika správce zpráv)
- Počet různých požadavků na objekty (statistika správce dat)
- Informace o Db2 využití (Db2 statistiky správce)
- Informace o využití prostředku Coupling Facility (statistika správce prostředku Coupling Facility)
- Informace o využití SMDS (statistika datové sady sdílených zpráv)
- Informace o využití fondu vyrovnávacích pamětí (statistika správce vyrovnávacích pamětí)
- Informace o protokolování (statistika správce protokolů)
- Informace o využití úložiště (statistika správce datových úložišť)
- Informace o požadavcích na zámek (statistika správce zámků)

Účetní data

- Trasování evidence shromažďuje informace o čase procesoru stráveném zpracováním volání MQI a o počtu požadavků MQPUT a MQGET zadaných konkrétním uživatelem.

- Produkt IBM MQ může také shromažďovat informace o každé úloze pomocí produktu IBM MQ. Tato data se shromažďují jako evidenční záznam na úrovni podprocesů. Pro každý podproces produkt IBM MQ také shromažďuje informace o každé frontě používané tímto podprocesem.

Data generovaná trasováním jsou odeslána do SMF (System Management Facility) nebo GTF (generalized trace facility).

Události

Události produktu IBM MQ poskytují informace o chybách, varováních a dalších významných výskytech ve správci front. Začleněním těchto událostí do své vlastní aplikace pro správu systému můžete monitorovat aktivity v rámci mnoha správců front pro více aplikací IBM MQ. Konkrétně můžete monitorovat všechny správce front v systému z jednoho správce front.

Události lze hlásit prostřednictvím mechanismu vytváření sestav napsaného uživatelem do administrační aplikace, která podporuje prezentaci událostí operátorovi. Události také umožňují aplikacím, které vystupují jako agenti pro jiné administrativní sítě, například NetView, monitorovat sestavy a vytvářet odpovídající výstrahy.

Související úlohy

[Použití trasování produktu IBM MQ](#)

[Použití událostí IBM MQ](#)

z/OS

Dispozice jednotky zotavení na systému z/OS

Určité transakční aplikace mohou při připojení ke správci front ve skupině sdílení front (QSG) namísto správce front QMGR používat dispoziční jednotku zotavení GROUP, a to zadáním názvu skupiny sdílení front při připojení namísto názvu správce front. To umožňuje, aby zotavení transakcí bylo flexibilnější a robustnější, a to odebráním požadavku na opětovné připojení ke stejnému správci front v rámci skupiny sdílení front.

Transakce spuštěné aplikacemi, které se připojily pomocí názvu skupiny sdílení front, mají také dispoziční jednotku zotavení GROUP.

Když se transakční aplikace připojí k jednotce zotavení GROUP, je logicky připojena ke skupině sdílení front a nemá afinitu k žádnému konkrétnímu správci front. Všechny transakce dvoufázového potvrzování, které byly spuštěny a které dokončily phase-1 procesu potvrzování, tj. jsou v nejistém stavu, lze při připojení k libovolnému správci front v rámci skupiny sdílení front zjišťovat a řešit. Ve scénáři zotavení to znamená, že koordinátor transakcí se nemusí znovu připojit ke stejnému správci front, který může být v daném okamžiku nedostupný.

Aplikace, které se připojují pomocí jednotky zotavení QMGR, mají přímou afinitu ke správci front, ke kterému jsou připojeny. Ve scénáři zotavení se musí koordinátor transakcí znovu připojit ke stejnému správci front, aby vyřešil všechny neověřené transakce bez ohledu na to, zda daný správce front patří do skupiny sdílení front.

Pokud aplikace určují název skupiny sdílení front, a proto se připojují ke správci front v rámci skupiny sdílení front s dispozicí zotavení GROUP, je skupina sdílení front logicky samostatným správcem prostředků. To znamená, že neověřené transakce jsou pro aplikaci viditelné pouze v případě, že se znovu připojí se stejnou dispozicí zotavení. Neověřené transakce s dispozicí zotavení jednotky QMGR nejsou viditelné pro aplikace, které se připojily k dispoziční jednotce zotavení GROUP a naopak.

Související pojmy

[“Povolení jednotek obnovy GROUP” na stránce 272](#)

Skupina sdílení front může konfigurovat a povolit podporu pro jednotky zotavení GROUP.

[“Podpora aplikací” na stránce 272](#)

Na této stránce můžete určit, které aplikace se mohou připojit k jednotce zotavení GROUP.

Povolení jednotek obnovy GROUP

Skupina sdílení front může konfigurovat a povolit podporu pro jednotky zotavení GROUP.

Chcete-li použít jednotky zotavení GROUP ve správci front v rámci skupiny sdílení front, povolte atribut správce front GROUPUR. Další informace o tomto konceptu viz [“Dispozice jednotky zotavení na systému z/OS”](#) na stránce 271 před přečtením zbytku tohoto tématu.

Je-li povolen atribut správce front GROUPUR, správce front přijme nová připojení s dispozicí zotavení jednotky GROUP. Pokud tento atribut zakážete, nebudou přijata nová připojení s touto dispozicí, ačkoli aplikace, které jsou již připojeny, nebudou ovlivněny, dokud se neodpojí.

Když se aplikace připojí k dispoziční jednotce zotavení GROUP a zjišťuje, které transakce jsou pochybné, nebo se pokusí vyřešit transakci, která byla spuštěna jinde ve skupině sdílení front (QSG), musí být správce front, ke kterému je nyní připojen, schopen komunikovat s ostatními členy skupiny sdílení front, aby mohl zpracovat požadavek. K tomu používá sdílenou frontu s názvem SYSTEM.QSG.UR.RESOLUTION.QUEUE. Tato fronta musí být ve struktuře obnovitelné aplikace s názvem CSQSYSAPPL. Struktura musí být obnovitelná, protože trvalé zprávy jsou uloženy v této frontě při zpracování požadavků na vyřešení.

Než budete moci povolit jednotky zotavení GROUP, musíte se ujistit, že je definována struktura prostředku Coupling Facility a sdílená fronta. Definice můžete použít v ukázce CSQ4INSS. Když je fronta definována nebo zjištěna během spuštění, každý správce front ve skupině sdílení front ji otevře, aby mohl přijímat příchozí požadavky. Chcete-li frontu odstranit nebo přesunout, protože byla nesprávně definována, můžete požádat správce front, aby v ní zavřeli své otevřené manipulátory, a to aktualizací objektu fronty tak, aby blokovaly požadavky MQGET. Po provedení nezbytných oprav umožní aplikace získat zprávy z fronty ještě jednou nařídít, aby je každý správce front znovu otevřel. Pomocí příkazu DISPLAY QSTATUS identifikujte, které manipulátory jsou ve frontě otevřeny.

Po dokončení tohoto nastavení můžete povolit jednotky zotavení GROUP pro každého správce front, ke kterému se mají transakční aplikace připojovat s použitím jednotky odebrání zotavení GROUP. Nemusí se jednat o všechny správce front v rámci skupiny sdílení front, ale pokud se rozhodnete povolit tuto funkci pouze v části skupiny sdílení front, musíte zajistit, aby se vaše aplikace pouze pokusily připojit ke správcům front, pro které jste ji povolili. Další informace viz téma [“Podpora aplikací”](#) na stránce 272.

Při pokusu o povolení atributu správce front GROUPUR je provedena řada kontrol konfigurace. Správce front kontroluje, zda:

- Patří do skupiny sdílení front.
- Sdílená fronta s názvem SYSTEM.QSG.UR.RESOLUTION.QUEUE byla definována podle definice v CSQ4INSS.
- Systém SYSTEM.QSG.UR.RESOLUTION.QUEUE je ve struktuře obnovitelného prostředku CF s názvem CSQSYSAPPL.

Pokud některá z výše uvedených kontrol selže, atribut GROUPUR zůstane vypnutý a vrátí se kód zprávy.

Tyto kontroly konfigurace se provádějí také při spuštění správce front, je-li povolen atribut správce front. Dojde-li k selhání některé z kontrol během spuštění jednotek zotavení GROUP a správce front vydá zprávu identifikující, která kontrola se nezdařila. Po provedení nezbytné nápravné akce musíte znovu povolit atribut správce front.

Podpora aplikací

Na této stránce můžete určit, které aplikace se mohou připojit k jednotce zotavení GROUP.

Podpora dispoziční jednotky zotavení GROUP je omezena na určité typy transakčních aplikací, pro které je produkt IBM MQ for z/OS správcem prostředků, nikoli však koordinátorem transakcí. Aktuálně podporované transakční aplikace jsou:

- IBM MQ rozšířené transakční klientské aplikace
- Aplikace IBM MQ classes for JMS spuštěné na aplikačním serveru, například WebSphere Application Server.

- Aplikace CICS spuštěné v produktu CICS Transaction Server 4.2 nebo novějším, když je definice prostředku CICS MQCONN nakonfigurována s RESYNCMEMBER (GROUPRESYNC).

Související pojmy

“IBM MQ rozšířené transakční klientské aplikace” na stránce 273

Na této stránce můžete určit, jak mohou rozšířené transakční klientské aplikace IBM MQ používat dispoziční jednotku GROUP pro zotavení.

“Aplikace rozhraní CICS” na stránce 273

Na této stránce můžete určit, jak může produkt CICS používat dispoziční jednotku GROUP pro obnovu.

z/OS IBM MQ rozšířené transakční klientské aplikace

Na této stránce můžete určit, jak mohou rozšířené transakční klientské aplikace IBM MQ používat dispoziční jednotku GROUP pro zotavení.

Příkladem rozšířené transakční klientské aplikace systému IBM MQ je aplikace, která používá produkt JMS a spouští se v produktu WebSphere Application Servera připojuje se k produktu IBM MQ prostřednictvím protokolu TCP/IP, nikoli pomocí lokálních vazeb. Tyto klientské aplikace se připojují k produktu IBM MQ for z/OS prostřednictvím síťových připojení, například prostřednictvím protokolu TCP/IP. Pro tyto aplikace je to hodnota zadaná pro parametr QMNAME řetězce xa_info předaného ve volání xa_open, která určuje, zda se má použít dispozice zotavení QMGR nebo GROUP. Další informace o xa_open viz [Formát řetězce xa_open](#) a [Další zpracování chyb pro xa_open](#). V případě aplikací JMS se toto provádí zadáním názvu skupiny sdílení front (QSG) v ConnectionFactory namísto názvu specifického správce front.

Aby mohly klientské aplikace XA využívat výhod použití jednotky zotavení GROUP, je třeba konfigurovat nastavení protokolu TCP/IP tak, aby bylo možné klientské aplikace směřovat na správce front ve skupině sdílení front, které mají povolen atribut GROUPUR, a nikoli na konkrétního správce front. Jednou z technologií dynamické virtuální adresy IP, kterou k tomu můžete použít, je distributor z/OS SysPlex . Další podrobnosti viz [z/OS Communications Server a z/OS Základní dovednosti: Dynamické virtuální adresování](#) . Chcete-li povolit jednotky zotavení GROUP pro podmnožinu správců front ve vaší skupině sdílení front, ujistěte se, že aplikace klienta nelze směřovat na ty, pro které není povolena.

Klientské aplikace se nemusí připojovat ke skupině sdílení front pomocí sdílených kanálů.

z/OS Aplikace rozhraní CICS

Na této stránce můžete určit, jak může produkt CICS používat dispoziční jednotku GROUP pro obnovu.

CICS 4.2 a novější poskytuje volbu resynchronizace skupiny, RESYNCMEMBER (GROUPRESYNC) v definici prostředku MQCONN. Agent CICS konfigurovaný s touto volbou se může připojit k libovolnému vhodnému správci front ve skupině sdílení front, která je spuštěna ve stejné oblasti LPAR jako daná oblast CICS . Chcete-li podporovat volbu CICS GROUPRESYNC, musí být správce front spuštěn v produktu MQ V7.1 nebo novější a musí být povolen pro podporu GROUPUR.

Transakce spuštěné v rámci oblasti CICS připojené k produktu MQ pomocí příkazu GROUPRESYNC vytvářejí jednotky práce s dispozicí obnovy GROUP.

Pomocí příkazu RESYNCMEMBER (GROUPRESYNC) můžete povolit rychlejší zotavení po selhání správce front, protože umožňuje oblasti CICS okamžité připojení k alternativnímu vhodnému správci front spuštěnému ve stejné oblasti LPAR a vyřešení všech neověřených transakcí podle potřeby bez čekání na restart správce front.

RESYNCMEMBER (GROUPRESYNC) také umožňuje flexibilnější volby restartu pro CICS. Oblast CICS s připojením MQ konfigurovanou pro použití sdílených front GROUPRESYNC a MQ lze restartovat v libovolné oblasti LPAR, kde je spuštěn správce front jako člen stejné skupiny sdílení front.

z/OS IBM MQ a další produkty z/OS

Toto téma popisuje, jak může produkt IBM MQ pracovat s jinými produkty z/OS .

Související pojmy

“IBM MQ a CICS” na stránce 274

Všechny verze produktu CICS podporované produktem IBM MQ 9.0.0a novější používají verzi adaptéru a mostu dodanou s produktem CICS .

[“IBM MQ for z/OS a WebSphere Application Server” na stránce 280](#)

V tomto tématu jsou uvedeny informace o použití produktu IBM MQ for z/OS produktem WebSphere Application Server.

Související odkazy

[“IBM MQ a IMS” na stránce 275](#)

Toto téma popisuje, jak produkt IBM MQ pracuje s produktem IMS. Adaptér IMS umožňuje připojení správce front k produktu IMSa umožňuje aplikacím IMS používat rozhraní MQI.

[“IBM MQ a adaptéry z/OS Batch, TSO a RRS” na stránce 279](#)

Toto téma popisuje, jak produkt IBM MQ pracuje s adaptéry z/OS Batch, TSO a RRS.

z/OS IBM MQ a CICS

Všechny verze produktu CICS podporované produktem IBM MQ 9.0.0a novější používají verzi adaptéru a mostu dodanou s produktem CICS .

Další informace o konfiguraci adaptéru IBM MQ CICS a komponentách IBM MQ CICS bridge naleznete v části [Konfigurace připojení k produktu IBM MQ](#) v dokumentaci k produktu CICS .

Související úlohy

[Použití IBM MQ s CICS](#)

z/OS CICS připojení skupiny

Připojení skupiny CICS umožňuje oblasti systému CICS připojit se k libovolnému aktivnímu členovi skupiny sdílení front systému IBM MQ ve stejné oblasti LPAR a neurčovat individuálního správce front. Produkt CICS se stále připojuje k jednomu správci front v daném okamžiku.

Chcete-li podporovat připojení skupiny CICS , potřebujete alespoň dva správce front v oblasti LPAR. Použití skupinového připojení poskytuje vyšší dostupnost, protože nepotřebujete, aby byl konkrétní správce front aktivní. Produkt CICS se připojí k libovolnému správci front ve skupině sdílení front v oblasti LPAR.

Další informace naleznete v dokumentaci k produktu CICS pro prostředek MQCONN.

Produkt CICS se pokusí o připojení k MQNAME předané, jako by se jednalo o správce front:

- Pokud správce front existuje a je aktivní, bude připojení fungovat.
- Pokud připojení selže, produkt CICS se dotáže na stav správců front ve skupině, aby se zjistily, které jsou aktivní ve stejné oblasti LPAR.
- Pokud je aktivní více správců front, produkt CICS zkontroluje stav RESYNCMEMBER (YES) a UOW, aby zjistil, zda se produkt CICS potřebuje připojit nebo by se měl připojit ke konkrétnímu členovi, nebo vyčkejte, pokud není aktivní.
- Není-li třeba se připojovat ke konkrétnímu členovi, produkt CICS vybere správce front (pomocí algoritmu náhodného výběru).
- Produkt CICS se pokusí připojit k vybranému správci front.
- Pokud se pokus nezdaří, pak v závislosti na návratovém kódu produkt CICS vybere další člen a pak znovu projde smyčkou výběru.
- Pokud nejsou aktivní žádní správci front, produkt CICS vydá více připojení k seznamu správců front a čeká na ECBLIST, dokud nebude k dispozici první správce front.

Související pojmy

[“Skupinové jednotky zotavení \(GROUPPUR\) pro CICS” na stránce 275](#)

IBM MQ GROUPPUR for CICS poskytuje zotavení typu peer pro neověřené pracovní jednotky ve skupině sdílení front (QSG). Jeden správce front IBM MQ může vyřešit neověřené pracovní jednotky jménem jiného správce front ve skupině sdílení front. To znamená, že pokud se produkt CICS znovu připojí

prostřednictvím připojení skupiny k jinému správci front v rámci skupiny sdílení front, může vyřešit neověřené transakce z předchozího připojení produktu IBM MQ .

Související informace

[Podpora skupin sdílení front IBM MQ](#)

z/OS Skupinové jednotky zotavení (GROUPUR) pro CICS

IBM MQ GROUPUR for CICS poskytuje zotavení typu peer pro neověřené pracovní jednotky ve skupině sdílení front (QSG). Jeden správce front IBM MQ může vyřešit neověřené pracovní jednotky jménem jiného správce front ve skupině sdílení front. To znamená, že pokud se produkt CICS znovu připojí prostřednictvím připojení skupiny k jinému správci front v rámci skupiny sdílení front, může vyřešit neověřené transakce z předchozího připojení produktu IBM MQ .

Pokud oblast CICS pracuje se správcem front a správce front skončí abnormálně, dojde k zotavení všech neověřených transakcí. Tím se eliminuje potřeba, aby oblast CICS čekala na správce front, se kterým pracoval, aby se restartoval, a poté vyřešila všechny sporné pracovní jednotky. To znamená, že v oblasti LPAR potřebujete alespoň dva správce front, aby se produkt CICS mohl připojit k jinému správci front v případě nestandardního ukončení prvního správce front.

Nové nastavení RESYNCMEMBER (GROUPRESYNC) na definici CICS MQCONN:

- Používá funkci připojení skupiny IBM MQ a obnovu typu peer.
- Vyžaduje správce front s povoleným atributem GROUPUR.
- Stále podporuje existující nastavení CICS MQCONN RESYNCMEMBER (YES a NO):
 - Použijte existující funkci připojení skupiny CICS a bez obnovy rovnocenných uzlů.
 - Změna nastavení RESYNCMEMBER se projeví při příštím CICS připojení k IBM MQ.

Související pojmy

[“Povolení jednotek obnovy GROUP” na stránce 272](#)

Skupina sdílení front může konfigurovat a povolit podporu pro jednotky zotavení GROUP.

z/OS IBM MQ a IMS

Toto téma popisuje, jak produkt IBM MQ pracuje s produktem IMS. Adaptér IMS umožňuje připojení správce front k produktu IMS a umožňuje aplikacím IMS používat rozhraní MQI.

Volitelný další most IBM MQ - IMS umožňuje aplikacím spouštět aplikaci IMS , která nepoužívá rozhraní MQI. To znamená, že starší aplikace můžete používat s produktem IBM MQ, aniž byste je museli přepisovat.

Další informace o těchto komponentách naleznete v následujících dílčích tématech:

Související pojmy

[IMS a IMS přemostění aplikací na IBM MQ for z/OS](#)

Související úlohy

[Nastavení adaptéru IMS](#)

[Nastavení mostu IMS](#)

[Provoz adaptéru IMS](#)

Související odkazy

[Záhlaví informací MQIIH- IMS](#)

z/OS Adaptér IMS

Adaptér IMS je rozhraní mezi aplikačními programy IMS a subsystémem IBM MQ .

Adaptéry IBM MQ umožňují různým aplikačním prostředím odesílat a přijímat zprávy prostřednictvím sítě pro řazení zpráv do fronty. Adaptér IMS je rozhraní mezi aplikačními programy IMS a subsystémem IBM MQ . Umožňuje aplikačním programům IMS používat rozhraní MQI.

Adaptér IMS přijímá a interpretuje požadavky na přístup k produktu IBM MQ pomocí External Subsystem Attach Facility (ESAF) poskytované společností IMS. Obvykle se produkt IMS připojí k produktu IBM MQ automaticky bez zásahu operátora.

Adaptér IMS poskytuje přístup k prostředkům IBM MQ pro programy spuštěné v následujících režimech nebo stavech:

- Režim úlohy (TCB)
- Stav problému
- Nekřížový režim paměti
- Režim registru bez přístupu

Adaptér poskytuje podproces připojení z řídicího bloku úlohy aplikace (TCB) do produktu IBM MQ.

Adaptér podporuje protokol dvoufázového potvrzování pro změny prostředků vlastněných uživatelem IBM MQ s tím, že IMS vystupuje jako koordinátor synchronizačního bodu. Konverzace, kde IMS není koordinátorem synchronizačního bodu, například konverzace chráněné APPC (SYNCLVL = SYNCPT), nejsou adaptérem IMS podporovány.

Adaptér také poskytuje transakci monitoru spouštěčů (CSQQTRMN). To je popsáno v tématu “Monitor spouštěčů IMS” na stránce 276.

Produkt IBM MQ můžete použít s produktem IMS Extended Recovery Facility (XRF) pro podporu zotavení z chyby IMS .

Poznámka: Od verze IMS 15.2 již není funkce XRF (Extended Recovery Facility) podporována. Další informace naleznete v dokumentaci k produktu IMS .

Použití adaptéru

Aplikační programy a adaptér IMS jsou spuštěny ve stejném adresním prostoru. Správce front je ve svém vlastním adresním prostoru oddělen.

Musíte propojit-upravit každý program, který vydá jedno nebo více volání MQI na vhodný modul rozhraní jazyka IMS , a pokud nepoužívá dynamická volání MQI, program typu stub rozhraní API dodaný IBM MQ, CSQOSTUB. Když aplikace vydá volání MQI, stub přenesení řízení do adaptéru prostřednictvím externího rozhraní subsystému IMS , které spravuje zpracování požadavku správcem front zpráv.

Administrace a provoz systému s produktem IMS

Autorizovaný operátor terminálu IMS může zadat příkazy IMS pro řízení a monitorování připojení k produktu IBM MQ. Operátor terminálu IMS však nemá žádnou kontrolu nad adresním prostorem IBM MQ . Operátor například nemůže vypnout IBM MQ z adresního prostoru IMS .

Omezení

Následující volání rozhraní API IBM MQ nejsou podporována v rámci aplikace používající adaptér IMS :

- MQCB
- MQCB_FUNCTION
- MQCTL

Monitor spouštěčů IMS

Monitor spouštěčů IMS (**CSQQTRMN**) je IBM MQdodaná IMS aplikace, která spustí transakci IMS , když dojde k události IBM MQ , například když je zpráva vložena do specifické fronty.

Způsob činnosti

Když je zpráva vložena do fronty zpráv aplikace, je generován spouštěč, pokud jsou splněny podmínky spouštěče. Správce front poté zapíše zprávu (obsahující některá data definovaná uživatelem), známou jako *zpráva spouštěče*, do inicializační fronty, která byla pro tuto frontu zpráv zadána. V prostředí IMS můžete spustit instanci CSQQTRMN, abyste monitorovali inicializační frontu a načtli z ní zprávy spouštěče, jakmile dorazí. Obvykle CSQQTRMN plánuje další transakci IMS pomocí příkazu INSERT (ISRT) do fronty zpráv IMS. Spuštěná aplikace IMS přečte zprávu z fronty zpráv aplikace a pak ji zpracuje. CSQQTRMN musí být spuštěn jako BMP bez zprávy.

Každá kopie CSQQTRMN poskytuje jednu inicializační frontu. Po spuštění se monitor spouštěčů spustí až do ukončení IBM MQ nebo IMS.

Makro APPLCTN pro CSQQTRMN musí určovat hodnotu SCHDTYP=PARALLEL.

Protože monitor spouštěčů je dávkově orientovaný BMP, transakce IMS, které jsou spuštěny monitorem spouštěčů, obsahují následující:

- Mezery v poli LTERM IOPCB
- Název PSB monitoru spouštěčů BMP v poli ID uživatele IOPCB

Pokud je cílová transakce IMS chráněna serverem zabezpečení (dříve známá jako RACF), možná budete muset definovat CSQQTRMN jako ID uživatele serveru zabezpečení.

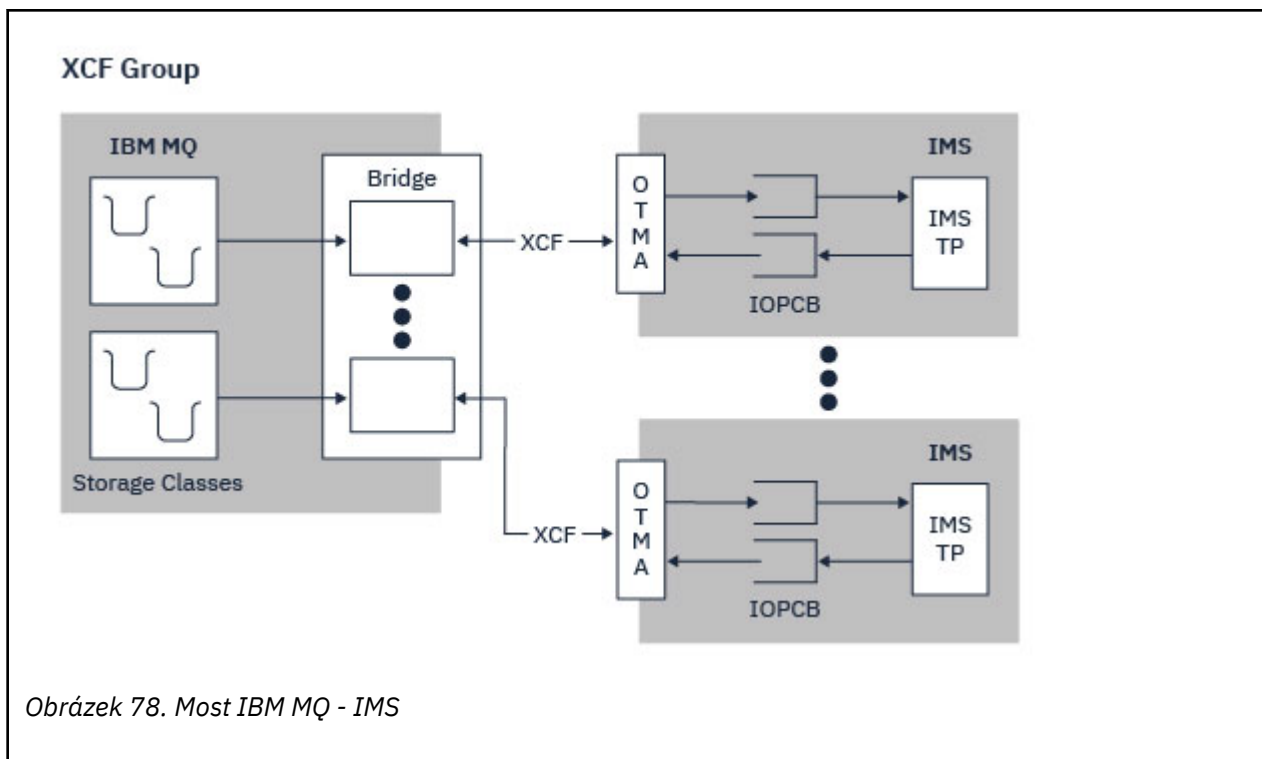
Most IBM MQ - IMS

Most IBM MQ - IMS je komponentou produktu IBM MQ for z/OS, která umožňuje přímý přístup z aplikací IBM MQ k aplikacím v systému IMS.

Most IBM MQ - IMS povoluje *implicitní podporu MQI*. To znamená, že můžete znovu navrhnout starší aplikace, které byly řízeny terminály připojenými k terminálu 3270, aby byly řízeny zprávami systému IBM MQ, aniž byste je museli přepsat, znovu zkompileovat nebo znovu propojit. Most je klientem produktu IMS *Open Transaction Manager Access (OTMA)*.

V aplikacích mostu nejsou v aplikaci IMS žádná volání IBM MQ. Aplikace získá svůj vstup pomocí GET UNIQUE (GU) na IOPCB a odešle svůj výstup pomocí ISRT na IOPCB. Aplikace systému IBM MQ používají záhlaví IMS (strukturu MQIIH) v datech zprávy, aby zajistily, že aplikace budou moci pracovat tak, jak to dělají, když jsou řízeny neprogramovatelnými terminály. Používáte-li aplikaci IMS, která zpracovává vícesegmentové zprávy, mějte na paměti, že všechny segmenty by měly být obsaženy v jedné zprávě IBM MQ.

Most IMS je znázorněn v souboru [Obrázek 78](#) na stránce 278.



Obrázek 78. Most IBM MQ - IMS

Správce front se může připojit k jednomu nebo více systémům IMS a více než jeden správce front se může připojit k jednomu systému IMS . Jediným omezením je, že všechny musí patřit do stejné skupiny XCF a musí být ve stejném prostředí sysplex.

Informace o nastavení mostu IMS a přidání dalšího připojení IMS ke stejnému správci front naleznete v tématu [Nastavení mostu IMS](#) .

Co je to OTMA?

Prostředek IMS OTMA je protokol klienta/serveru bez připojení založený na transakcích, který běží na systému IMS. Funguje jako rozhraní pro hostitelské komunikační servery, které přistupují k aplikacím IMS TM prostřednictvím z/OS prostředku XCF (Cross Systems coupling facility).

OTMA umožňuje klientům připojit se k produktu IMS , aby poskytovaly vysoký výkon pro interakce mezi klienty a produktem IMS pro velkou síť nebo velký počet relací. Aplikace OTMA je implementována v prostředí sysplex z/OS . Proto je doména OTMA omezena na doménu XCF.

Monitorování prostředků OTMA

Podpora pro zprávy protokolu OTMA x '3C', která je k dispozici v produktu IMS v10 nebo vyšší, je součástí mostu IBM MQ - IMS v produktu IBM MQ for z/OS. Tyto zprávy jsou odesílány klientům OTMA společností IMS , aby ohlásili jejich stav.

Pokud partner IMS není schopen zpracovat objem odesílaných požadavků na transakce, upozorní IBM MQ , že došlo k varování při zahlcení. V odezvě IBM MQ zpomalí rychlost, jakou jsou požadavky odesílány přes most.

Pokud produkt IMS stále není schopen zpracovat požadavky transakce a dojde k úplnému stavu zahlcení, všechny TPIPE pro partnera IMS jsou pozastaveny. Po oznámení od partnera IMS , že došlo k odlehčení stavu povodně nebo varování před záplavou, bude produkt IBM MQ v případě potřeby pokračovat ve všech pozastavených TPIPE a postupně zvyšovat rychlost odesílání požadavků na transakce, dokud nebude dosaženo maximální rychlosti. Zprávy konzoly jsou vydávány produktem IBM MQ v reakci na změnu stavu partnerů systému IMS .

Pokud se používají partneři IMS v10 , měli byste se ujistit, že byla použita oprava PTF UK45082 .

Odeslání IMS transakcí z IBM MQ

Chcete-li odeslat transakci IMS, která používá most, aplikace vloží zprávy do fronty IBM MQ jako obvykle. Zprávy obsahují IMS transakční data; mohou mít záhlaví IMS (strukturu MQIIH) nebo mohou mostu IBM MQ - IMS umožnit vytvářet předpoklady o datech ve zprávě.

IBM MQ pak vloží zprávu do fronty IMS (nejprve je zařazena do fronty IBM MQ, aby umožnila použití synchronizačního bodu k zajištění integrity dat). Paměťová třída fronty IBM MQ určuje, zda se jedná o frontu OTMA (tj. frontu používanou k přenosu zpráv do mostu IBM MQ - IMS) a konkrétního partnera systému IMS, kterému se odesílají data zpráv.

Vzdálení správci front mohou také spouštět transakce produktu IMS zápisem do těchto front OTMA v systému IBM MQ for z/OS.

Data vrácená ze systému IMS se zapisují přímo do fronty IBM MQ reply-to určené ve struktuře deskriptoru zpráv (MQMD). (Může se jednat o přenosovou frontu pro správce front určeného v poli **ReplyToQMgr** deskriptoru MQMD.)

Související pojmy

[IMS a IMS přemostění aplikací na IBM MQ for z/OS](#)

Související úlohy

[Přizpůsobení mostu IMS](#)

Související odkazy

[“IBM MQ a IMS” na stránce 275](#)

Toto téma popisuje, jak produkt IBM MQ pracuje s produktem IMS. Adaptér IMS umožňuje připojení správce front k produktu IMS a umožňuje aplikacím IMS používat rozhraní MQI.

z/OS

IBM MQ a adaptéry z/OS Batch, TSO a RRS

Toto téma popisuje, jak produkt IBM MQ pracuje s adaptéry z/OS Batch, TSO a RRS.

Úvod do adaptéru dávek

Adaptéry Batch/TSO jsou rozhraní mezi aplikačními programy IBM MQ a z/OS spuštěnými pod aplikacemi JES, TSO nebo z/OS UNIX System Services. Tyto adaptéry umožňují aplikačním programům z/OS používat rozhraní MQI.

Adaptéry poskytují přístup k prostředkům IBM MQ pro programy spuštěné v následujících režimech nebo stavech:

- Režim úlohy (TCB)
- Stav problému nebo supervizora
- Nekřížový režim paměti
- Režim registru bez přístupu

Připojení mezi aplikačními programy a produktem IBM MQ jsou na úrovni úlohy. Adaptéry poskytují podproces připojení z řídicího bloku úloh aplikace (TCB) do produktu IBM MQ.

Adaptér Batch/TSO podporuje protokol jednofázového potvrzování pro změny prostředků vlastních společností IBM MQ. Nepodporuje protokoly vícefázového potvrzování. Adaptér RRS umožňuje aplikacím IBM MQ účastnit se protokolů dvofázového potvrzování s dalšími produkty podporujícími službu RRS, které koordinuje služba RRS (z/OS Resource Recovery Services).

Adaptéry používají službu z/OS STIMERM k plánování asynchronní události každou sekundu. Tato událost spustí blok požadavků přerušení (IRB), který nezahrnuje žádné čekání úlohy aplikace dávky. Tato IRB kontroluje, zda byla ECB při ukončení IBM MQ vyslána. Pokud byla ECB ukončena, IRB zveřejní všechny ECB aplikace, které čekají na událost v IBM MQ (například signál nebo čekání).

Adaptér Batch/TSO

Adaptér IBM MQ Batch/TSO poskytuje IBM MQ podporu pro aplikace z/OS Batch a TSO. Všechny aplikační programy, které jsou spuštěny v rámci z/OS Batch nebo TSO, musí mít s sebou upravený odkaz na program stub rozhraní API CSQBSTUB. Stub poskytuje aplikaci přístup ke všem voláním MQI. Pro aplikace používáte jednofázové potvrzování a odvolání pomocí volání MQI **MQCMIT** a **MQBACK**.

Adaptér RRS

Resource Recovery Services (RRS) je dílčí komponenta produktu z/OS, která poskytuje celosystémovou službu pro koordinaci dvoufázového potvrzování v rámci produktů z/OS. Adaptér IBM MQ Batch/TSO RRS (adaptér RRS) poskytuje IBM MQ podporu pro aplikace z/OS Batch a TSO, které chtějí používat tyto služby. Adaptér RRS umožňuje produktu IBM MQ stát se úplným účastníkem koordinace RRS. Aplikace se mohou podílet na zpracování dvoufázového potvrzování s jinými produkty, které podporují službu RRS (například Db2).

Adaptér RRS poskytuje dva stuby; musíte propojit-upravit aplikační programy, které chtějí používat RRS s jedním z těchto stubů.

CSQBRSTB

Tento stub vám umožňuje používat dvoufázové potvrzování a vracení pro aplikace pomocí volatelných služeb zotavení prostředků RRS namísto volání MQI **MQCMIT** a **MQBACK**.

Musíte také propojit modul ATRSCSS z knihovny SYS1.CSSLIB s vaší aplikací. Pokud použijete volání MQI **MQCMIT** a **MQBACK**, obdržíte návratový kód MQRC_ENVIRONMENT_ERROR.

CSQBRRSI

Tento stub vám umožňuje používat volání MQI **MQCMIT** a **MQBACK**; IBM MQ skutečně implementuje tato volání jako volání **SRRCMIT** a **SRRBACK** RRS.

Informace o sestavování aplikačních programů, které používají adaptér RRS, naleznete v tématu [Dávkový adaptér RRS](#).

Kde najít další informace o adaptérech z/OS Batch, TSO a RRS

Další informace o tématech v této sekci naleznete v následujících zdrojích:

<i>Tabulka 25. Kde najít další informace o použití z/OS Dávka s IBM MQ</i>	
Téma	Kam se podívat
Nastavení adaptérů dávek	Úloha 19: Nastavení adaptérů Batch, TSO a RRS
Volatelné služby zotavení prostředků RRS	Programování MVS: Volatelné služby pro jazyky vysoké úrovně

z/OS

IBM MQ for z/OS a WebSphere Application Server

V tomto tématu jsou uvedeny informace o použití produktu IBM MQ for z/OS produktem WebSphere Application Server.

Aplikace napsané v adresáři Java, které jsou spuštěny v adresáři WebSphere Application Server, mohou používat specifikaci Java Message Service (JMS) k provádění systému zpráv. Dvoubodový systém zpráv v tomto prostředí může být poskytován správcem front IBM MQ for z/OS.

Výhodou použití správce front IBM MQ for z/OS k poskytování systému zpráv je, že připojení aplikací JMS může plně využívat funkce sítě IBM MQ. Mohou například používat most IMS nebo vyměňovat zprávy se správci front spuštěnými na jiných platformách.

Připojení mezi produktem WebSphere Application Server a správcem front

Další informace viz [Použití IBM MQ a WebSphere Application Server společně](#) .

Použití funkcí IBM MQ z aplikací JMS

Standardně zprávy JMS zadržené ve frontách IBM MQ používají záhlaví MQRFH2 k uchování některých informací záhlaví zpráv JMS . Mnoho starších aplikací IBM MQ nemůže zpracovat zprávy s těmito záhlavími a vyžaduje jejich vlastní charakteristická záhlaví, například MQCIH for CICS Bridge nebo MQWIH for IBM MQ Workflow. Další podrobnosti o těchto zvláštních aspektech naleznete v tématu [Mapování JMS zpráv na IBM MQ zprávy](#).

Managed File Transfer

Produkt Managed File Transfer přenáší soubory mezi systémy spravovaným a auditovatelným způsobem bez ohledu na velikost souboru nebo použité operační systémy.

Produkt Managed File Transfer můžete použít k vytvoření přizpůsobeného, rozšiřitelného a automatizovaného řešení, které vám umožní spravovat, důvěřovat a zabezpečit přenosy souborů. Produkt Managed File Transfer eliminuje nákladné redundance, snižuje náklady na údržbu a maximalizuje vaše stávající investice do IT.

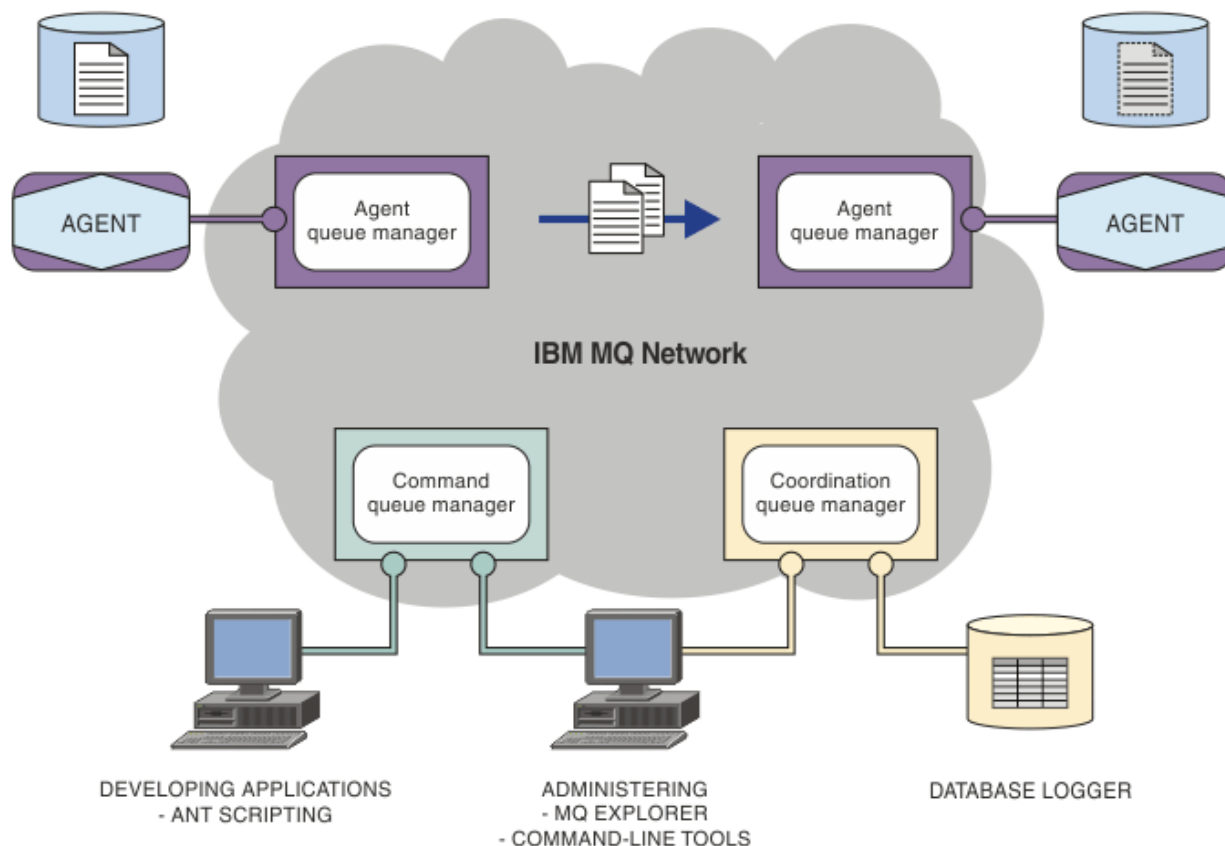

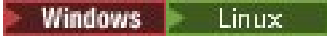
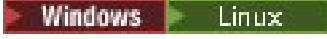


Diagram zobrazuje jednoduchou topologii Managed File Transfer . Existují dva agenti, z nichž každý se připojuje k vlastnímu správci front agenta v síti IBM MQ . Soubor je přenesen z agenta na jedné straně diagramu prostřednictvím sítě IBM MQ na agenta na druhé straně diagramu. V síti IBM MQ jsou také koordinační správce front a správce front příkazů. Aplikace a nástroje se připojují k těmto správcům front za účelem konfigurace, administrace, provozu a protokolování Managed File Transfer aktivity v síti IBM MQ .

Produkt Managed File Transfer lze nainstalovat jako čtyři různé volby v závislosti na operačním systému a celkovém nastavení. Tyto volby jsou Managed File Transfer Agent, Managed File Transfer Logger, Managed File Transfer Service nebo Managed File Transfer Tools. Další informace viz [Volby produktu Managed File Transfer](#).

Produkt Managed File Transfer můžete použít k provedení následujících úloh:

- Vytvořit spravované přenosy souborů
 -  Vytvořte nové přenosy souborů z IBM MQ Explorer na platformách Linux nebo Windows .
 - Vytvořte nové přenosy souborů z příkazového řádku na všech podporovaných platformách.
 - Integrujte funkci přenosu souborů do nástroje Apache Ant .
 - Zapisuje aplikace, které řídí produkt Managed File Transfer vložím zpráv do front příkazů agenta.
 - Naplánujte přenosy souborů, které se mají uskutečnit později. Můžete také spustit naplánované přenosy souborů na základě rozsahu událostí systému souborů, například vytvářený nový soubor.
 - Průběžně monitorujte prostředek, například adresář, a když obsah tohoto prostředku splňuje nějakou předdefinovanou podmínku, spusťte úlohu. Tato úloha může být přenosem souborů, skriptem Ant nebo úlohou JCL.
 - Přeneste soubory do a z front IBM MQ .
 - Přenos souborů na a ze serverů FTP, FTPS nebo SFTP.
 - Přeneste soubory do a z uzlů Connect:Direct .
 - Přenos textových i binárních souborů. Textové soubory se automaticky převádějí mezi kódovými stránkami a konvence konce řádku zdrojového a cílového systému.
 - Přenosy mohou být zabezpečeny pomocí odvětvových standardů pro připojení založená na protokolu SSL (Secure Socket Layer).
- Zobrazit probíhající přenosy a protokolovat informace o všech přenosech ve vaší síti
 -  Zobrazte stav probíhajících přenosů z IBM MQ Explorer na platformách Linux nebo Windows .
 -  Zkontrolujte stav dokončených přenosů pomocí IBM MQ Explorer na platformách Linux nebo Windows .
 - Pomocí funkce modulu protokolování databáze Managed File Transfer můžete ukládat zprávy protokolu do databáze Db2 nebo Oracle .

Produkt Managed File Transfer je postaven na produktu IBM MQ, který poskytuje zajištěné jednorázové doručení zpráv mezi aplikacemi. Můžete využít různých funkcí produktu IBM MQ. Můžete například použít kompresi kanálu ke kompresi dat, která odesíláte mezi agenty přes kanály IBM MQ , a použít kanály SSL k zabezpečení dat, která odesíláte mezi agenty. Soubory jsou přenášeny spolehlivě a mohou tolerovat selhání infrastruktury, přes kterou je přenos souborů prováděn. Dojde-li k výpadku sítě, přenos souborů se při obnově konektivity restartuje od místa, kde byl vypnut.

Konsolidací přenosu souborů s existující sítí IBM MQ se můžete vyhnout výdajům na prostředky potřebné k údržbě dvou samostatných infrastruktur. Pokud ještě nejste zákazníkem IBM MQ , vytvořením IBM MQ sítě pro podporu Managed File Transfer vytváříte páteř pro budoucí implementaci architektury SOA. Pokud jste již IBM MQ zákazníkem, Managed File Transfer může využít stávající infrastrukturu IBM MQ včetně IBM MQ Internet Pass-Thru a IBM Integration Bus.

Můžete využít výhod řešení vysoké dostupnosti IBM MQ ke zlepšení odolnosti konfigurace produktu Managed File Transfer . Pokud vaši agenti používají správce front replikovaných dat (RDQM), musíte je nakonfigurovat tak, aby používaly funkci plovoucí adresy IP. To znamená, že agenti používají stejnou adresu IP ke komunikaci s jakoukoli ze tří instancí RDQM, která je momentálně spuštěna, a automaticky se znovu připojí při překonání selhání (viz [Vysoká dostupnost RDQM](#) a [Vytvoření a odstranění plovoucí adresy IP](#)). Používáte-li řešení správce front s více instancemi, budou aplikace při komunikaci s každou

instancí používat jinou adresu IP, která je obsluhována opětovným připojením klienta při překonání selhání (viz [Správci front s více instancemi](#) a [Opětovné připojení kanálu a klienta](#)).

Produkt Managed File Transfer se integruje s řadou dalších produktů IBM :

IBM Integration Bus

Soubory procesu, které byly přeneseny produktem Managed File Transfer jako součást toku IBM Integration Bus . Další informace viz [Práce s MFT z IBM Integration Bus](#).

IBM Sterling Connect:Direct

Přeneste soubory do a z existující sítě Connect:Direct pomocí mostu Managed File Transfer Connect:Direct . Další informace naleznete v tématu [Most Connect:Direct](#).

IBM Tivoli Composite Application Manager

Produkt IBM Tivoli Composite Application Manager poskytuje agenta, kterého můžete použít k monitorování informací publikovaných v koordinačním správci front.

Související pojmy

Volby produktu Managed File Transfer

[“MFT přehled topologie” na stránce 284](#)

[Přehled způsobu připojení agentů Managed File Transfer ke koordinačnímu správci front v síti IBM MQ .](#)

[“Jak MFT pracuje s produktem IBM MQ?” na stránce 283](#)

Produkt Managed File Transfer spolupracuje s produktem IBM MQ mnoha způsoby.

Jak MFT pracuje s produktem IBM MQ?

Produkt Managed File Transfer spolupracuje s produktem IBM MQ mnoha způsoby.

- Produkt Managed File Transfer přenáší soubory mezi procesy agenta rozdělením každého souboru do jedné nebo více zpráv a přenosem zpráv prostřednictvím sítě IBM MQ .
- Agent zpracovává data souboru přesunu pomocí dočasných zpráv, aby minimalizoval dopad na vaše protokoly IBM MQ . Vzájemnou komunikací procesy agenta regulují tok zpráv obsahujících data souboru. Tím se zabrání vytváření zpráv obsahujících data souboru v přenosových frontách systému IBM MQ a zajistí se, že v případě, že některá z přechodných zpráv nebude doručena, budou data souboru znovu odeslána.
- Agenti Managed File Transfer používají několik front IBM MQ . Další informace viz [MFT systémové fronty a systémové téma](#).
- Ačkoli jsou některé z těchto front striktně určeny pro interní použití, agent může přijímat požadavky ve formě speciálně formátovaných zpráv příkazů odeslaných do specifické fronty, ze které agent čte. Jak příkazy příkazového řádku, tak i modul plug-in IBM MQ Explorer odesílají IBM MQ zprávy agentovi, aby instruoval agenta, aby provedl požadovanou akci. Tímto způsobem můžete psát aplikace IBM MQ , které interaktivně spolupracují s agentem. Další informace viz [Řízení MFT vložení zpráv do fronty příkazů agenta](#).
- Agenti produktu Managed File Transfer odesílají informace o svém stavu a průběhu a výsledku přenosů do správce front produktu MQ , který byl určen jako koordinační správce front. Tyto informace jsou publikovány koordinačním správcem front a mohou být přihlášeny k odběru aplikacemi, které chtějí monitorovat průběh přenosu nebo uchovávat záznamy o přenosech, k nimž došlo. Jak příkazy příkazového řádku, tak modul plug-in IBM MQ Explorer mohou používat publikované informace. Můžete psát aplikace IBM MQ , které používají tyto informace. Další informace o tématu, do kterého jsou informace publikovány, viz [SYSTEM.FTE téma](#).
- Klíčové komponenty produktu Managed File Transfer využívají schopnosti správců front produktu IBM MQ ukládat a předávat zprávy. To znamená, že pokud dojde k výpadku, mohou neovlivněné části vaší infrastruktury pokračovat v přenosu souborů. To se vztahuje na koordinačního správce front, kde kombinace uložení a předání a trvalých odběrů umožňuje koordinačnímu správci front tolerovat nedostupnost, aniž by došlo ke ztrátě klíčových informací o probíraných přenosech souborů.

MFT přehled topologie

Přehled způsobu připojení agentů Managed File Transfer ke koordinačnímu správci front v síti IBM MQ .

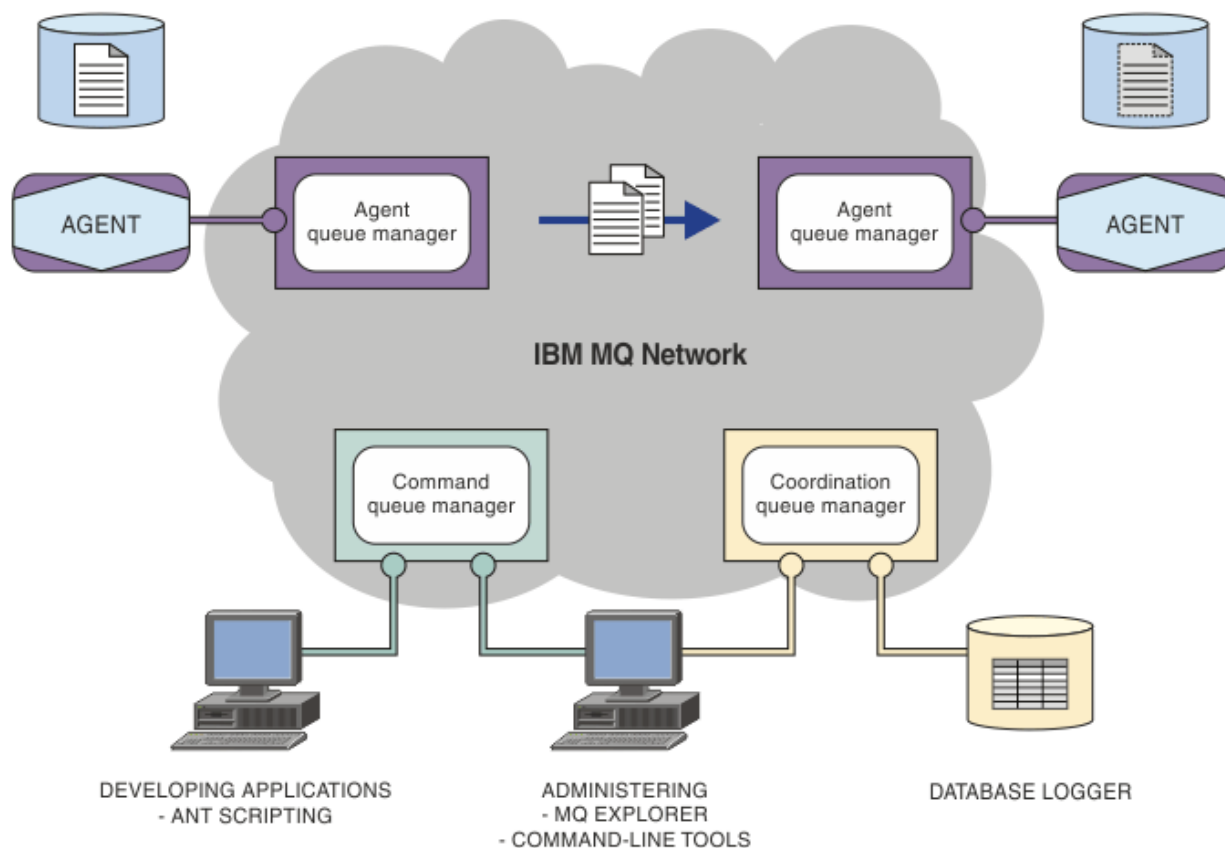
Agenti Managed File Transfer odesílají a přijímají přenášené soubory. Každý agent má ve svém přidruženém správci front vlastní sadu front a je připojen ke svému správci front buď v režimu vazeb, nebo v režimu klienta. Agent může také použít koordinačního správce front jako svého správce front.

Koordinační správce front vysílá informace o auditu a přenosu souborů. Koordinační správce front představuje jeden bod pro shromažďování informací o agentu, stavu přenosu a auditu přenosu. Koordinační správce front nemusí být k dispozici, aby bylo možné provádět přenosy. Pokud se koordinační správce front stane dočasně nedostupným, přenosy budou pokračovat jako obvykle. Zprávy auditu a stavové zprávy jsou uloženy ve správci front agenta, dokud není koordinační správce front k dispozici, a pak je lze zpracovat jako obvykle.

Agenti se registrují v koordinačním správci front a publikují své podrobnosti do tohoto správce front. Tyto informace o agentovi používá modul plug-in Managed File Transfer k povolení spuštění přenosů z produktu IBM MQ Explorer. Informace o agentovi shromážděné v koordinačním správci front jsou také použity příkazy k zobrazení informací o agentovi a stavu agenta.

Stav přenosu a informace o auditu přenosu jsou publikovány v koordinačním správci front. Stav přenosu a informace o auditu přenosu používá modul plug-in Managed File Transfer k monitorování průběhu přenosů z produktu IBM MQ Explorer. Informace o auditu přenosu uložené v koordinačním správci front lze zachovat pro zajištění auditovatelnosti.

Správce front příkazů se používá pro připojení k síti IBM MQ a je správcem front připojeným při zadávání příkazů systému Managed File Transfer .



Související pojmy

“Managed File Transfer” na stránce 281

Produkt Managed File Transfer přenáší soubory mezi systémy spravovaným a auditovatelným způsobem bez ohledu na velikost souboru nebo použité operační systémy.

[“Jak MFT pracuje s produktem IBM MQ?” na stránce 283](#)

Produkt Managed File Transfer spolupracuje s produktem IBM MQ mnoha způsoby.

[Managed File Transfer scénář](#)

Přehled produktu MFTREST API

REST API podporuje určité příkazy Managed File Transfer , včetně výpisu přenosů a podrobností o agentech přenosu souborů.

Z produktu IBM MQ 9.1.0 produkt REST API zahrnuje volby pro vypsání všech aktuálních Managed File Transfer přenosů a pro dotazování na stav agentů Managed File Transfer . Další informace viz téma [Začínáme s produktem REST API MFT](#).

IBM MQ Internet Pass-Thru

IBM MQ Internet Pass-Thru (MQIPT) je volitelná komponenta produktu IBM MQ , kterou lze použít k implementaci řešení systému zpráv mezi vzdálenými servery v Internetu.

V produktu IBM MQ 9.2.0 je MQIPT volitelnou komponentou produktu IBM MQ. Chcete-li získat instalační soubory MQIPT pro produkt IBM MQ 9.3.x, přejděte na [IBM Fix Central pro IBM MQ](#). Před IBM MQ 9.2.0, MQIPT byl k dispozici jako balík podpory.

Nemusíte spouštět IBM MQ 9.3 , abyste mohli používat MQIPT v IBM MQ 9.3. Produkt MQIPT můžete použít pro připojení k libovolné podporované verzi produktu IBM MQ a nemusíte instalovat žádné další komponenty produktu IBM MQ ve stejné verzi jako produkt MQIPT.

Pokud jste zakoupili oprávnění k produktu IBM MQ , můžete nainstalovat libovolný počet kopií produktu MQIPT. Instalace produktu MQIPT se nezapočítávají do zakoupeného oprávnění IBM MQ . Další informace o licencování systému IBM MQ viz [IBM MQ informace o licenci](#).

Poznámka: Tato dokumentace se týká MQIPT v IBM MQ 9.3. Dokumentace k balíku podpory MQIPT (verze 2.1) v adresáři IBM Documentation viz [MQIPT \(SupportPac MS81\)](#) v dokumentaci k produktu IBM MQ 9.0 .

Poznámka: Používáte-li produkt MQIPT 2.1 nebo starší, doporučuje se provést upgrade na verzi MQIPT pro systém IBM MQ 9.3, protože datum ukončení podpory pro balík podpory MQIPT bylo 30th září 2020.

Produkt IBM MQ Internet Pass-Thru je spuštěn jako samostatná služba, která může přijímat a předávat IBM MQ toky zpráv, a to buď mezi dvěma správci front IBM MQ , nebo mezi klientem IBM MQ a správcem front IBM MQ .

Produkt MQIPT povolí toto připojení, pokud klient a server nejsou ve stejné fyzické síti.

Jednu nebo více instancí produktu MQIPT lze umístit do komunikační cesty mezi dvěma správci front IBM MQ nebo mezi klientem IBM MQ a správcem front IBM MQ . Instance systému MQIPT umožňují dvěma systémům IBM MQ výměnu zpráv bez nutnosti přímého připojení TCP/IP mezi těmito dvěma systémy. To je užitečné, pokud konfigurace brány firewall zakazuje přímé připojení TCP/IP mezi těmito dvěma systémy.

Produkt MQIPT naslouchá na jednom nebo více portech TCP/IP pro příchozí připojení, která mohou přenášet buď normální IBM MQ zprávy, IBM MQ zprávy v rámci protokolu HTTP, nebo zprávy šifrované pomocí protokolu TLS (Transport Layer Security) nebo SSL (Secure Sockets Layer). Produkt MQIPT může zpracovat více souběžných připojení.

Kanál IBM MQ , který vytváří počáteční požadavek na připojení TCP/IP, je označován jako *volající*, kanál, ke kterému se pokouší připojit jako *odpovídací modul*, a správce front, kterého se v konečném důsledku pokouší kontaktovat jako *správce cílové fronty*.

Produkt MQIPT uchovává data v paměti při jejich předávání ze svého zdroje do svého místa určení. Na disk se neukládají žádná data (s výjimkou paměti stránkované operačním systémem na disk). Jediným okamžikem, kdy produkt MQIPT explicitně přistupuje k disku, je čtení jeho konfiguračního souboru a zápis protokolu připojení a trasovacích záznamů.

Úplný rozsah typů kanálů IBM MQ lze vytvořit prostřednictvím jedné nebo více instancí produktu MQIPT. Přítomnost MQIPT v komunikační cestě nemá žádný vliv na funkční charakteristiku připojených komponent produktu IBM MQ, ale může mít určitý vliv na výkon přenosu zpráv.

MQIPT lze použít ve spojení s IBM MQ a IBM Integration Bus, jak je popsáno v tématu [“Možné konfigurace produktu MQIPT”](#) na stránce 289.

Chcete-li nainstalovat produkt MQIPT, prohlédněte si téma [Instalace MQIPT](#).

Související úlohy

[Konfigurace produktu IBM MQ Internet Pass-Thru](#)

[Administrace a konfigurace produktu IBM MQ Internet Pass-Thru](#)

Související odkazy

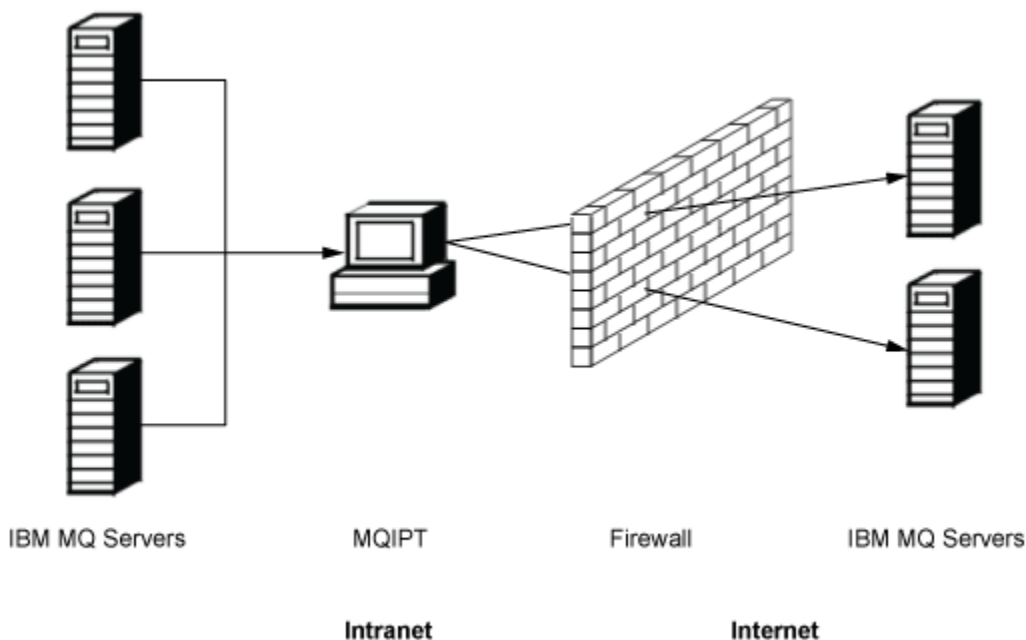
[IBM MQ Internet Pass-Thru odkaz na konfiguraci](#)

Použití MQIPT

Existuje řada potenciálních použití pro IBM MQ Internet Pass-Thru (MQIPT).

MQIPT lze použít jako koncentrátor kanálů

Při použití MQIPT tímto způsobem se kanály do více samostatných hostitelů nebo z více samostatných hostitelů mohou objevit v bráně firewall, jako by byly všechny do hostitele MQIPT nebo z něj. To usnadňuje definování a správu pravidel filtrování brány firewall.

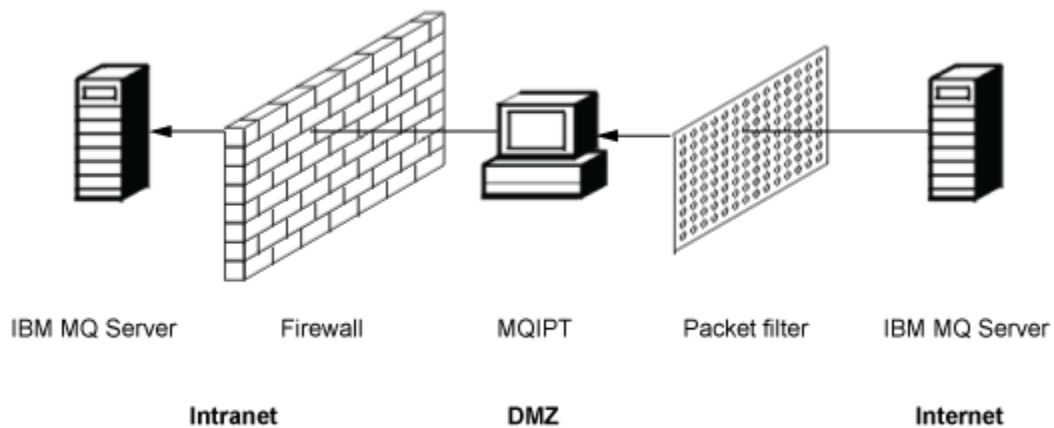


Obrázek 79. Příklad MQIPT jako koncentrátor kanálu

Produkt MQIPT lze umístit do zóny DMZ, aby poskytoval jediný přístupový bod.

Je-li produkt MQIPT umístěn v rámci brány firewall DMZ (konfigurace brány firewall pro zabezpečení lokálních sítí), lze v počítači se známou a důvěryhodnou adresou IP (Internet Protocol) produkt MQIPT použít k naslouchání příchozím připojením kanálu IBM MQ, která pak může předat důvěryhodnému intranetu; vnitřní brána firewall musí tomuto důvěryhodnému počítači povolit vytváření příchozích připojení. V této konfiguraci produkt MQIPT zabraňuje externím požadavkům na přístup v přijímání skutečných adres IP počítačů v důvěryhodném intranetu. Tímto způsobem produkt MQIPT poskytuje

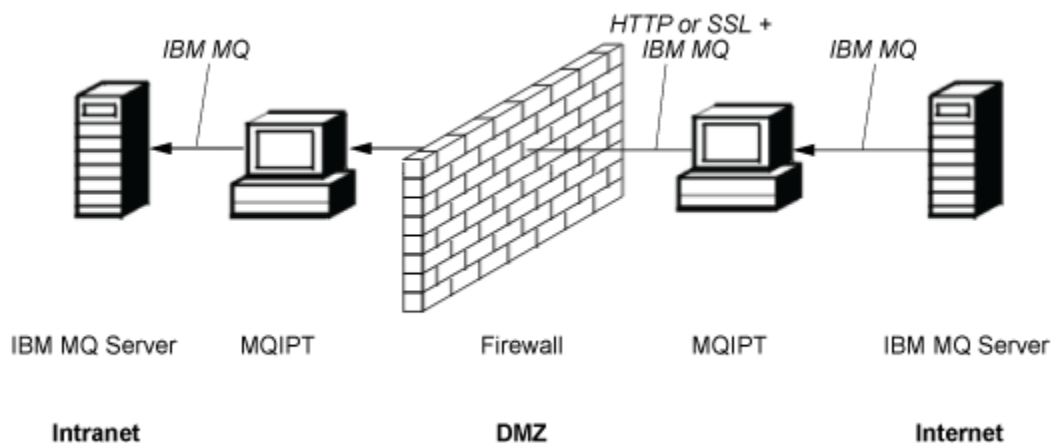
jediný přístupový bod. V případě potřeby lze produkt MQIPT nakonfigurovat tak, aby přijímal připojení TLS a předával data do místa určení pomocí samostatného připojení TLS, a ukončil tak relaci TLS v zóně DMZ.



Obrázek 80. Příklad MQIPT v bráně firewall DMZ

Produkt MQIPT může komunikovat prostřednictvím tunelového propojení HTTP .

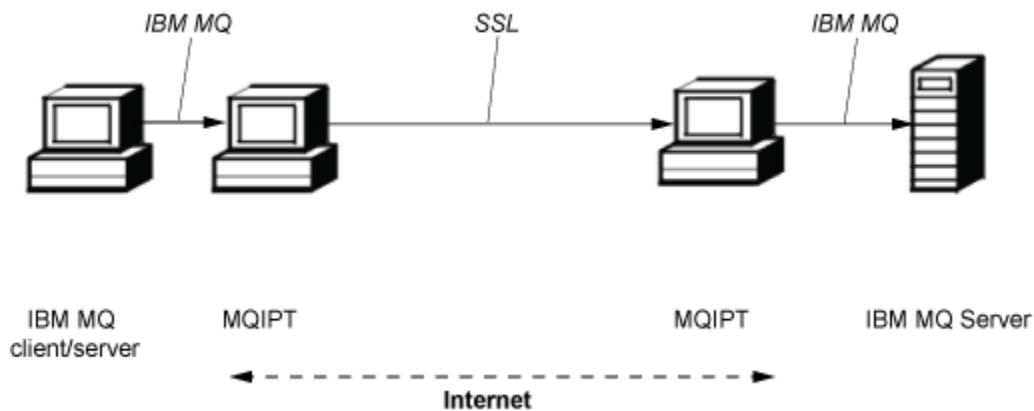
Pokud jsou dvě instance produktu MQIPT implementovány na lince, mohou komunikovat pomocí HTTP. Funkce tunelového propojení HTTP umožňuje přenos požadavků přes brány firewall pomocí existujících serverů proxy HTTP . První MQIPT vloží protokol IBM MQ do protokolu HTTP a druhý extrahuje protokol IBM MQ ze svého modulu wrapper HTTP a předá jej správci cílové fronty.



Obrázek 81. Příklad tunelového propojení MQIPT a HTTP

MQIPT může šifrovat zprávy

Pokud je MQIPT nakonfigurován jako v předchozím příkladu, požadavky mohou být před přenosem přes brány firewall zašifrovány. První MQIPT zašifruje data a druhý je dešifruje pomocí SSL/TLS před jejich odesláním do cílového správce fronty.



Obrázek 82. Příklad MQIPT a SSL/TLS

Jak MQIPT funguje

Ve své nejjednodušší konfiguraci se produkt MQIPT chová jako přesměrovač protokolu IBM MQ . Naslouchá na portu TCP/IP a přijímá požadavky na připojení z kanálů IBM MQ .

Pokud je přijat správně utvořený požadavek, produkt MQIPT vytvoří další připojení TCP/IP mezi sebou a cílovým správcem front IBM MQ . Poté předá všechny pakety protokolu, které obdrží od svého příchozího připojení, do cílového správce front a vrátí pakety protokolu z cílového správce front zpět do původního příchozího připojení.

Není zahrnuta žádná změna protokolu IBM MQ (klient/server nebo správce front na správce front), protože ani jeden konec si není přímo vědom přítomnosti prostředníka. Nové verze kódu klienta nebo serveru IBM MQ nejsou vyžadovány.

Chcete-li použít produkt MQIPT, musí být kanál volajícího konfigurován tak, aby používal název hostitele a port systému MQIPT , nikoli název hostitele a port cílového správce front. Je definován s vlastností **CONNNAME** kanálu IBM MQ . Produkt MQIPT přečte příchozí data a jednoduše je předá do cílového správce front. Další konfigurační pole, například ID uživatele a heslo v kanálu klient/server, jsou podobně předána správci cílové fronty.

Více správců front

Produkt MQIPT lze použít k povolení přístupu k více než jednomu správci cílových front. Aby to fungovalo, musí existovat mechanismus, který produktu MQIPT sdělí, ke kterému správci front se má připojit, takže produkt MQIPT použije číslo příchozího portu TCP/IP k určení, ke kterému správci front se má připojit.

Proto můžete nakonfigurovat produkt MQIPT tak, aby naslouchal na více portech TCP/IP. Každý naslouchající port je mapován na cílového správce front prostřednictvím MQIPT *trasy*. Můžete definovat až 100 takových přenosových cest, které přidruží naslouchající port TCP/IP k názvu hostitele a portu cílového správce front. To znamená, že název hostitele (adresa IP) cílového správce front není nikdy viditelný pro původní kanál. Každá trasa může obsluhovat více připojení mezi svým naslouchajícím portem a cílem, přičemž každé připojení pracuje nezávisle.

MQIPT konfigurační soubor

Produkt MQIPT používá konfigurační soubor s názvem `mqipt.conf`. Tento soubor obsahuje definice všech přenosových cest a jejich přidružené vlastnosti. Další informace o produktu `mqipt.conf` naleznete v tématu [Administrace a konfigurace produktu IBM MQ Internet Pass-Thru](#) .

Když je spuštěn produkt MQIPT , spustí každou trasu, která je uvedena v konfiguračním souboru. Zprávy se zapisují do systémové konzoly, která zobrazuje stav každé přenosové cesty. Je-li pro trasu zobrazena zpráva MQCP1078 , je tato trasa připravena přijmout požadavky na připojení.

Možné konfigurace produktu MQIPT

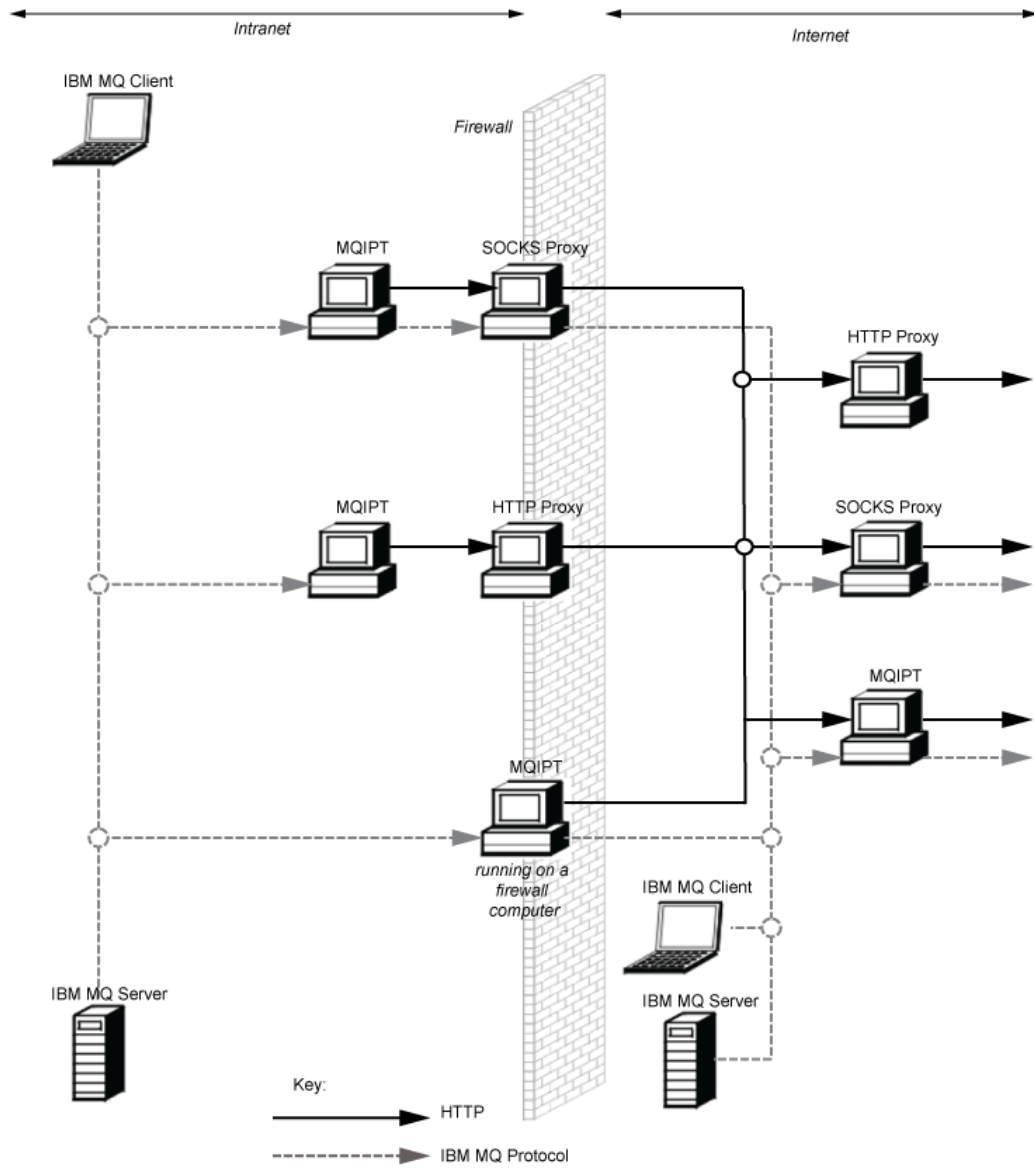
MQIPT lze použít ve spojení s IBM MQ a IBM Integration Bus.

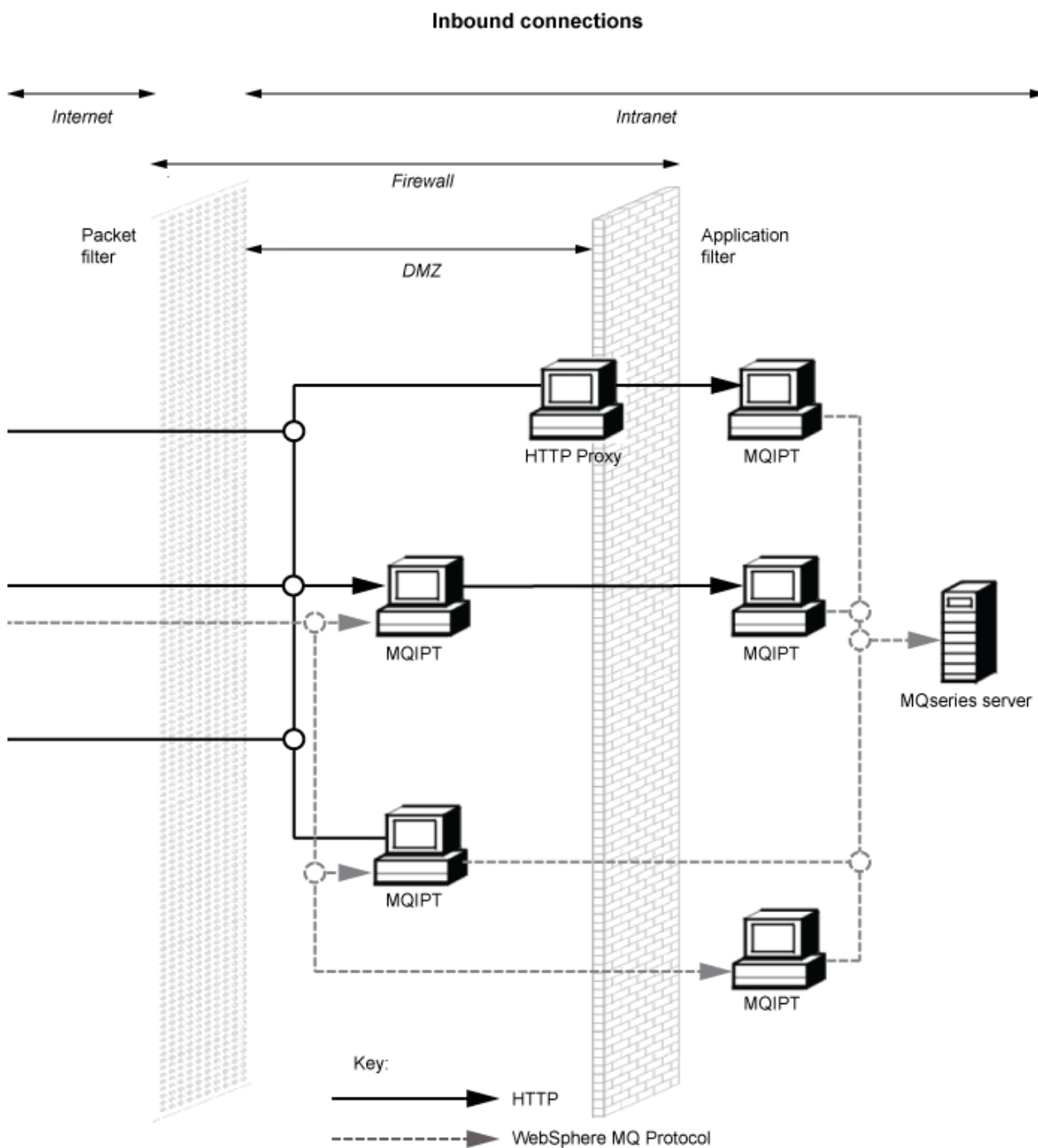
Následující vícedílný obrázek ukazuje mnoho možných konfigurací pro produkt MQIPT v topologii produktu IBM MQ . Ilustruje různé způsoby, jak může produkt MQIPT odesílat zprávy. Zobrazuje klienty a servery na intranetu, uvnitř brány firewall a na Internetu mimo bránu firewall a předává zprávy serveru proxy MQIPT, HTTP nebo serveru proxy SOCKS, který je předává.

Zprávy jsou přijaty serverem proxy MQIPT nebo serverem proxy HTTP v zóně DMZ před předáním zprávy přes příchozí bránu firewall na server.

Všimněte si, že počítače HTTP proxy, SOCKS proxy a MQIPT na intranetové straně brány firewall představují možnost, že více počítačů bude zřetězeno na internetu. Počítač se systémem MQIPT může například komunikovat prostřednictvím jednoho nebo více počítačů se serverem proxy SOCKS nebo HTTP nebo dalších počítačů se systémem MQIPT , než dosáhne svého cíle.

Outbound connections





Kompatibilní konfigurace

Scénáře kompatibilního připojení, kdy klient IBM MQ nebo správce front komunikuje s produktem MQIPT. Stejná nebo druhá přenosová cesta MQIPT se používá ke komunikaci se správcem cílové fronty.

Kompatibilní konfigurace s jednou přenosovou cestou MQIPT

Ke komunikaci s produktem IBM MQ můžete použít jednu přenosovou cestu MQIPT .

Sloupce v souboru [Tabulka 26](#) na stránce 292 obsahují následující informace:

1. Protokol použitý mezi IBM MQ a přenosovou cestou MQIPT . Připojení může být vytvořeno buď klientem IBM MQ , nebo správcem front a může používat buď protokol IBM MQ FAP (Formats and Protocols), nebo protokol SSL/TLS.
2. Režim, ve kterém trasa MQIPT funguje. Formát komunikace přes Internet mezi MQIPT a IBM MQ je určen konfigurací přenosové cesty MQIPT . Všimněte si, že tam, kde tabulka zmiňuje SSL, můžete také použít TLS.
3. Protokol použitý mezi přenosovou cestou MQIPT a správcem cílové fronty.

<i>Tabulka 26. Platné konfigurace s jedinou instancí MQIPT</i>		
1. IBM MQ zdrojový protokol	2. Režim MQIPT přenosové cesty	3. IBM MQ cílový protokol
FAP	FAP-proxy (výchozí)	FAP
	FAP-server a SSL-klient	SSL/TLS
SSL/TLS	SSL-proxy	SSL/TLS
	SSL-server a FAP-klient	FAP
	SSL-server a SSL-klient	SSL/TLS

Kompatibilní konfigurace s více než jednou přenosovou cestou MQIPT

Pro komunikaci s produktem IBM MQ můžete použít více než jednu trasu v jedné nebo více instancích produktu MQIPT.

Sloupce v souboru [Tabulka 27](#) na stránce 293 obsahují následující informace:

1. Protokol použitý mezi IBM MQ a první přenosovou cestou MQIPT . Připojení může být vytvořeno buď klientem IBM MQ , nebo správcem front a může používat buď protokol IBM MQ FAP (Formats and Protocols), nebo protokol SSL/TLS.
2. Režim, ve kterém funguje první přenosová cesta MQIPT . Formát komunikace přes Internet mezi MQIPT a IBM MQ je určen konfigurací přenosové cesty MQIPT . Všimněte si, že tam, kde tabulka zmiňuje SSL, můžete také použít TLS.
3. Režim, ve kterém druhá přenosová cesta MQIPT funguje.
4. Protokol použitý mezi druhou přenosovou cestou MQIPT a správcem cílové fronty.

Tabulka 27. Platné konfigurace s více instancemi MQIPT

1. IBM MQ zdrojový protokol	2. Režim první MQIPT přenosové cesty	3. Režim druhé MQIPT přenosové cesty	4. IBM MQ cílový protokol
FAP (výchozí)	FAP-proxy (výchozí)	FAP-proxy (výchozí)	FAP
	FAP-server a SSL-klient	SSL-proxy	SSL/TLS
		SSL-server a FAP-klient	FAP
		SSL-server a SSL-klient	SSL/TLS
	HTTP-klient	HTTP-server a SSL-klient	SSL/TLS
	HTTPS-klient	HTTPS-server a SSL-klient	SSL/TLS
	HTTP-klient	HTTP-server	FAP
HTTPS-klient	HTTPS-server	FAP	
SSL/TLS	SSL-proxy	SSL-proxy	SSL/TLS
		SSL-server a FAP-klient	FAP
		SSL-server a SSL-klient	SSL/TLS
	HTTP-klient	HTTP-server	FAP
	HTTPS-klient	HTTPS-server	SSL/TLS
	HTTP-klient	HTTP-server a SSL-klient	FAP
	HTTPS-klient	HTTPS-server a SSL-klient	SSL/TLS

Podporované konfigurace kanálů

Všechny typy kanálů IBM MQ jsou podporovány, ale konfigurace je omezena na připojení TCP/IP. Pro IBM MQ klienta nebo správce front MQIPT se zobrazí, jako by se jedná o cílového správce front. Kde konfigurace kanálu vyžaduje cílového hostitele a číslo portu, je uveden název hostitele MQIPT a číslo portu modulu listener.

Kanály klient/server

Produkt MQIPT naslouchá příchozím požadavkům na připojení klienta a předává je buď pomocí tunelového propojení HTTP, SSL/TLS, nebo jako standardní pakety protokolu IBM MQ. Pokud produkt MQIPT používá tunelové propojení HTTP nebo SSL/TLS, předá je na připojení do druhého MQIPT. Pokud nepoužívá tunelové propojení HTTP, předá je na připojení tomu, co vidí jako správce cílové fronty (i když to může být další MQIPT). Po přijetí připojení klienta správcem cílové fronty jsou pakety přenášeny mezi klientem a serverem.

Odesílací/přijímací kanály klastru

Pokud produkt MQIPT přijme příchozí požadavek od kanálu odesílatele klastru, předpokládá, že správce front má povolen SOCKS, a skutečná cílová adresa bude získána během procesu navázání komunikace SOCKS. Postoupí požadavek dalšímu správci front MQIPT nebo cílovému správci front přesně stejným způsobem jako pro kanály připojení klienta. To zahrnuje také automaticky definované odesílací kanály klastru.

Odesílatel/příjemce

Pokud produkt MQIPT přijme příchozí požadavek od odesílatelého kanálu, předá jej dalšímu MQIPT nebo cílovému správci front přesně stejným způsobem jako pro kanály připojení klienta. Správce cílové fronty ověří příchozí požadavek a v případě potřeby spustí příjímací kanál. Veškerá komunikace mezi odesílacím a příjímacím kanálem (včetně toků zabezpečení) je přenášena.

Žadatel/server

Tato kombinace je zpracována stejným způsobem jako předchozí konfigurace. Ověření požadavku na připojení provádí kanál serveru v cílovém správci front.

Žadatel/odesílatel

Konfigurace zpětného volání může být použita v případě, že dvěma správčům front není povoleno vzájemně navazovat přímá připojení, ale mohou se připojovat k produktu MQIPT i přijímat od něj připojení.

Server/žadatel a server/příjemce

Tyto jsou obsluhovány produktem MQIPT stejným způsobem, jakým obsluhuje konfiguraci produktu Sender/Receiver.

Podmínky ukončení a selhání kanálu

Když produkt MQIPT zjistí uzavření (normální nebo abnormální) kanálu IBM MQ, rozšíří uzavření kanálu. Pokud zavřete trasu pomocí produktu MQIPT, všechny kanály procházející touto trasou se zavřou.

Produkt MQIPT poskytuje volitelný prostředek časového limitu nečinnosti. Pokud produkt MQIPT zjistí, že kanál byl nečinný po dobu překračující časový limit, provede okamžité ukončení práce na dvou dotčených připojeních.

Systémy IBM MQ na obou koncích kanálu sledují tyto nestandardní podmínky ukončení práce buď jako selhání sítě, nebo jako ukončení kanálu jejich partnerem. Kanál je pak schopen provést restart a zotavení (pokud k selhání dojde během období nejistoty protokolu), jako by nebyl používán produkt MQIPT.

Bezpečnost zpráv

Správa distribuovaných front systému IBM MQ zajišťuje správné doručení zpráv. K tomu dochází i v případě, že je mezi dvěma konci kanálu uveden parametr MQIPT. Produkt MQIPT neukládá žádná data zprávy ani se neúčastní procedury synchronizačního bodu, která zajišťuje správné doručení zprávy.

Používáte-li rychlé, dočasné zprávy IBM MQ, může dojít ke ztrátě zprávy v případě, že trasa MQIPT selže nebo je restartována při přenosu zprávy IBM MQ. Před restartováním přenosové cesty se ujistěte, že všechny kanály IBM MQ používající přenosovou cestu MQIPT jsou neaktivní.

Další informace o bezpečnosti zpráv v produktu IBM MQ naleznete v tématu [Bezpečnost zpráv](#).

Správci front s více instancemi a vysoká dostupnost

Produkt MQIPT lze použít se správci front s více instancemi v prostředích s vysokou dostupností.

Produkt MQIPT nemá trvalý stav, a proto nemá žádný přínos v tom, že by překonal MQIPT selhání na jiném systému. Místo toho mějte více instancí MQIPT s identickými konfiguračními soubory `mqipt.conf` spuštěných na různých systémech. Monitorujte dostupnost každé instance produktu MQIPT a v případě potřeby ji restartujte (na stejném systému). To poskytuje sadu identických instancí produktu MQIPT, které lze použít ke směrování připojení. Poté se musíte ujistit, že produkt IBM MQ může směrovat připojení do produktu MQIPT a že produkt MQIPT může tato připojení postoupit do cílového správce front.

Odchozí kanály IBM MQ mohou být směrovány na dostupnou instanci produktu MQIPT různými způsoby, například:

- Použijte prostředek pro vyrovnávání zátěže nebo směrovač s vysokou dostupností, například IBM Network Dispatcher z produktu WebSphere Edge Components.
- Zadejte více názvů připojení v definici kanálu IBM MQ pomocí seznamu odděleného čárkami. Produkt IBM MQ se následně pokusí připojit ke každé adrese MQIPT, dokud nenajde dostupnou instanci MQIPT.

Musíte také směrovat připojení z produktu MQIPT do cílového správce front. Pokud konfigurace vysoké dostupnosti zajišťuje překonání selhání adresy IP se správcem cílové fronty, není vyžadována žádná speciální konfigurace MQIPT: zadejte cílovou adresu IP ve vlastnosti trasy **Destination** a umožněte operaci překonání selhání přesunout adresu IP se správcem front.

Pokud se však po překonání selhání změní adresa IP správce front, je třeba zajistit, aby produkt MQIPT předával připojení do správného místa určení. To lze provést jedním z několika způsobů:

- Zapište uživatelskou proceduru směrování, která kontroluje, která adresa IP a číslo portu jsou přístupné, a pak přepište cíl přenosové cesty pro každé připojení. Některé uživatelské procedury směrování vzorků jsou poskytovány s produktem MQIPT; pro tento účel je lze upravit.
- K přesměrování připojení použijte prostředek pro vyrovnávání zátěže s vysokou dostupností.
- Definujte více tras MQIPT, jednu pro každou adresu IP a port, kde může být spuštěn správce front. Poté směřujte připojení IBM MQ na různé přenosové cesty MQIPT, například vypsáním všech adres IP přenosové cesty a čísel portů v seznamu s čárkami jako oddělovači v názvu připojení odchozího kanálu.

Je také důležité vyladit všechny komponenty typu end-to-end na síťové cestě:

1. Pokusy o připojení k nedostupným systémům musí okamžitě selhat, aby se pokusy o opětovné připojení mohly přesunout do prvního dostupného místa určení.

Pro MQIPT trasy SSL vyladte vlastnost trasy **SSLClientConnectTimeout**, abyste zajistili selhání připojení výzvy k zadání pro nedostupná místa určení. Podrobné informace o parametrech ladění IBM MQ naleznete v dokumentaci k systému IBM MQ. Podrobné informace o ladění protokolu TCP/IP pro operační systém naleznete v dokumentaci k operačnímu systému. Ve všech případech by nezdařené pokusy o připojení měly rychle vrátit selhání sítě (například paket resetování TCP) nebo by měly bez zbytečného odkladu dojít k překročení časového limitu.

2. Aktivní připojení k systému, který selhal, musí být okamžitě přerušena, aby bylo možné navázat nová připojení.

Měli byste také zvážit dopad překonání selhání v době, kdy připojení aktivně používají produkt MQIPT. Je pravděpodobné, že síťová připojení budou během překonání selhání přerušena. V případě klientských aplikací můžete použít funkci automatického opětovného připojení klienta IBM MQ k opětovnému vytvoření přerušovaných připojení. Pro kanály zpráv můžete zadat krátký interval opakování, aby se kanál rychle znovu připojil. Další informace o automatickém opětovném připojení klienta a konfiguraci opakování kanálu zpráv naleznete v dokumentaci k systému IBM MQ.

V 9.3.5 IBM MQ Console a REST API

Pomocí protokolů IBM MQ Console a REST API můžete spravovat produkt IBM MQ a provádět operace systému zpráv pomocí protokolu HTTP.

- Pomocí konzoly IBM MQ Console můžete provádět základní administrativní úlohy z webového prohlížeče. Další informace naleznete v tématu [Administrace pomocí konzoly IBM MQ Console](#).
- Produkt administrativní REST API můžete použít k administraci objektů IBM MQ, jako jsou správci front a fronty, a agentů a přenosů Managed File Transfer. Další informace naleznete v tématu [Administrace pomocí konzoly REST API](#).
- Pomocí konzoly messaging REST API můžete provádět jednoduché dvoubodové zprávy a publikovat zprávy. Další informace naleznete v tématu [Zasílání zpráv pomocí konzoly REST API](#).

Volby instalace

IBM MQ Console a REST API jsou spuštěny na serveru WebSphere Liberty s názvem mqweb. V produktu IBM MQ 9.3.5 můžete nainstalovat server mqweb buď jako volitelnou komponentu v instalaci produktu IBM MQ, nebo jako samostatnou instalaci produktu IBM MQ Web Server.

V 9.3.5

Linux

Samostatná instalace produktu IBM MQ Web Server

V produktu IBM MQ 9.3.5 lze server mqweb spustit v samostatné instalaci produktu IBM MQ Web Server. Samostatná instalace produktu IBM MQ Web Server vám umožňuje nainstalovat a spustit server mqweb na systémech, které jsou oddělené od vašich instalací produktu IBM MQ. Instalace samostatného serveru IBM MQ Web Server poskytuje větší flexibilitu, pokud jde o systémy a počet systémů, na kterých se rozhodnete spouštět servery mqweb. V případě potřeby lze několik instancí serveru mqweb spustit na různých počítačích a zajistit tak potřebnou rozšiřitelnost a dostupnost.

Pokud jste zakoupili oprávnění IBM MQ , můžete nainstalovat libovolný počet kopií samostatného produktu IBM MQ Web Server. Instalace produktu IBM MQ Web Server se nezapočítávají do zakoupeného oprávnění IBM MQ . Další informace o licencování systému IBM MQ viz [IBM MQ informace o licenci](#).

V samostatné instalaci produktu IBM MQ Web Server platí následující omezení:

- Produkt IBM MQ Console lze použít pouze k administraci vzdálených správců front.
- messaging REST API lze použít pouze se vzdálenými správci front.
- administrative REST API není k dispozici.

Samostatný soubor IBM MQ Web Server je podporován pouze na platformách Linux .

Další informace o instalaci samostatného serveru IBM MQ Web Server naleznete v tématu [Instalace samostatného serveru IBM MQ Web Server](#).






Volitelná komponenta instalace produktu IBM MQ

Můžete zvolit instalaci komponent IBM MQ Console a REST API jako součást instalace produktu IBM MQ .

Při spuštění serveru mqweb v instalaci produktu IBM MQ jsou k dispozici všechny funkce IBM MQ Console a REST API .

- Produkt IBM MQ Console lze použít k administraci lokálních a vzdálených správců front.
- messaging REST API lze použít s lokálními a vzdálenými správci front.
- Produkt administrative REST API lze použít k administraci lokálních a vzdálených správců front.

Chcete-li použít komponenty IBM MQ Console a REST API , nainstalujte následující komponentu jako součást instalace produktu IBM MQ :

-  V systému AIX nainstalujte sadu souborů mqm.web.rte .
-  V systému IBM i nainstalujte webovou komponentu.
-  V systému Linux nainstalujte komponentu MQSeriesWeb .
-  V systému Windows nainstalujte funkci Web Administration .
-  V systému z/OS nainstalujte funkci IBM MQ for z/OS UNIX System Services Web Components .

Tyto informace byly vyvinuty pro produkty a služby poskytované v USA.

Společnost IBM nemusí nabízet produkty, služby nebo funkce uvedené v tomto dokumentu v jiných zemích. Informace o produktech a službách, které jsou ve vaší oblasti aktuálně dostupné, získáte od místního zástupce společnosti IBM. Odkazy na produkty, programy nebo služby společnosti IBM v této publikaci nejsou míněny jako vyjádření nutnosti použití pouze uvedených produktů, programů či služeb společnosti IBM. Místo toho lze použít jakýkoli funkčně ekvivalentní produkt, program nebo službu, které neporušují žádná práva k duševnímu vlastnictví IBM. Ověření funkčnosti produktu, programu nebo služby pocházející od jiného výrobce je však povinností uživatele.

Společnost IBM může vlastnit patenty nebo nevyřízené žádosti o patenty zahrnující předměty popsané v tomto dokumentu. Vlastnictví tohoto dokumentu neposkytuje licenci k těmto patentům. Dotazy týkající se licencí můžete posílat písemně na adresu:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Odpovědi na dotazy týkající se licencí pro dvoubajtové znakové sady (DBCS) získáte od oddělení IBM Intellectual Property Department ve vaší zemi, nebo tyto dotazy můžete zasílat písemně na adresu:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Následující odstavec se netýká Spojeného království ani jiných zemí, ve kterých je takovéto vyjádření v rozporu s místními zákony: SPOLEČNOST INTERNATIONAL BUSINESS MACHINES CORPORATION TUTO PUBLIKACI POSKYTUJE "TAK, JAK JE" BEZ JAKÝCHKOLIV ZÁRUK, VYJÁDŘENÝCH VÝSLOVNĚ NEBO VYPLÝVAJÍCÍCH Z OKOLNOSTÍ, VČETNĚ, A TO ZEJMÉNA, ZÁRUK NEPORUŠENÍ PRÁV TŘETÍCH STRAN, PRODEJNOSTI NEBO VHODNOSTI PRO URČITÝ ÚČEL. Některé právní řády u určitých transakcí nepřipouštějí vyloučení záruk výslovně vyjádřených nebo vyplývajících z okolností, a proto se na vás toto omezení nemusí vztahovat.

Uvedené údaje mohou obsahovat technické nepřesnosti nebo typografické chyby. Údaje zde uvedené jsou pravidelně upravovány a tyto změny budou zahrnuty v nových vydáních této publikace. Společnost IBM může kdykoli bez upozornění provádět vylepšení nebo změny v produktech či programech popsaných v této publikaci.

Veškeré uvedené odkazy na webové stránky, které nespravuje společnost IBM, jsou uváděny pouze pro referenci a v žádném případě neslouží jako záruka funkčnosti těchto webů. Materiály uvedené na tomto webu nejsou součástí materiálů pro tento produkt IBM a použití uvedených stránek je pouze na vlastní nebezpečí.

Společnost IBM může použít nebo distribuovat jakékoli informace, které jí sdělíte, libovolným způsobem, který společnost považuje za odpovídající, bez vyžádání vašeho svolení.

Vlastníci licence k tomuto programu, kteří chtějí získat informace o možnostech (i) výměny informací s nezávisle vytvořenými programy a jinými programy (včetně tohoto) a (ii) oboustranného využití vyměňovaných informací, mohou kontaktovat informační středisko na adrese:

IBM Corporation
Kordinátor interoperability softwaru, oddělení 49XA
3605 Dálnice 52 N

Rochester, MN 55901
U.S.A.

Poskytnutí takových informací může být podmíněno dodržením určitých podmínek a požadavků zahrnujících v některých případech uhrazení stanoveného poplatku.

Licencovaný program popsáný v těchto informacích a veškerý licencovaný materiál, který je pro něj k dispozici, jsou poskytovány společností IBM na základě podmínek IBM Smlouvy se zákazníkem, IBM Mezinárodní licenční smlouvy pro programy nebo jiné ekvivalentní smlouvy mezi námi.

Jakékoli údaje o výkonnosti obsažené v této publikaci byly zjištěny v řízeném prostředí. Výsledky získané v jakémkoli jiném operačním prostředí se proto mohou výrazně lišit. Některá měření mohla být prováděna na vývojových verzích systémů a není zaručeno, že tato měření budou stejná i na běžně dostupných systémech. Některá měření mohla být navíc odhadnuta pomocí extrapolace. Skutečné výsledky mohou být jiné. Čtenáři tohoto dokumentu by měli zjistit použitelné údaje pro své specifické prostředí.

Informace týkající se produktů jiných výrobců pocházejí od dodavatelů těchto produktů, z jejich veřejných oznámení nebo z jiných veřejně dostupných zdrojů. Společnost IBM tyto produkty netestovala a nemůže potvrdit správný výkon, kompatibilitu ani žádné jiné výroky týkající se produktů jiných výrobců než IBM. Otázky týkající se kompatibility produktů jiných výrobců by měly být směřovány dodavatelům těchto produktů.

Veškerá tvrzení týkající se budoucího směru vývoje nebo záměrů společnosti IBM se mohou bez upozornění změnit nebo mohou být zrušena a reprezentují pouze cíle a plány společnosti.

Tyto údaje obsahují příklady dat a sestav používaných v běžných obchodních operacích. Aby byla představa úplná, používají se v příkladech jména osob a názvy společností, značek a produktů. Všechna tato jména a názvy jsou fiktivní a jejich podobnost se jmény, názvy a adresami používanými ve skutečnosti je zcela náhodná.

LICENČNÍ INFORMACE:

Tyto informace obsahují ukázkové aplikační programy ve zdrojovém jazyce ilustrující programovací techniky na různých operačních platformách. Tyto ukázkové programy můžete bez závazků vůči společnosti IBM jakýmkoli způsobem kopírovat, měnit a distribuovat za účelem vývoje, používání, odbytu či distribuce aplikačních programů odpovídajících rozhraní API pro operační platformu, pro kterou byly ukázkové programy napsány. Tyto příklady nebyly plně testovány za všech podmínek. Společnost IBM proto nemůže zaručit spolehlivost, upotřebitelnost nebo funkčnost těchto programů.

Při prohlížení těchto dokumentů v elektronické podobě se nemusí zobrazit všechny fotografie a barevné ilustrace.

Informace o programovacím rozhraní

Informace o programovacím rozhraní, jsou-li poskytnuty, jsou určeny k tomu, aby vám pomohly vytvořit aplikační software pro použití s tímto programem.

Tato příručka obsahuje informace o zamýšlených programovacích rozhraních, která zákazníkům umožňují psát programy za účelem získání služeb produktu WebSphere MQ.

Tyto informace však mohou obsahovat i diagnostické údaje a informace o úpravách a ladění. Informace o diagnostice, úpravách a vyladění jsou poskytovány jako podpora ladění softwarových aplikací.

Důležité: Tyto informace o diagnostice, úpravách a ladění nepoužívejte jako programovací rozhraní, protože se mohou měnit.

Ochranné známky

IBM, logo IBM, ibm.com, jsou ochranné známky společnosti IBM Corporation, registrované v mnoha jurisdikcích po celém světě. Aktuální seznam ochranných známek společnosti IBM je k dispozici na webu "Copyright and trademark information" www.ibm.com/legal/copytrade.shtml. Další názvy produktů a služeb mohou být ochrannými známkami společnosti IBM nebo jiných společností.

Microsoft a Windows jsou ochranné známky společnosti Microsoft Corporation ve Spojených státech a případně v dalších jiných zemích.

UNIX je registrovaná ochranná známka skupiny The Open Group ve Spojených státech a případně v dalších jiných zemích.

Linux je registrovaná ochranná známka Linuse Torvaldse ve Spojených státech a případně v dalších jiných zemích.

Tento produkt zahrnuje software vyvinutý projektem Eclipse (<https://www.eclipse.org/>).

Java a všechny ochranné známky a loga založené na termínu Java jsou ochranné známky nebo registrované ochranné známky společnosti Oracle anebo příbuzných společností.



Číslo položky:

(1P) P/N: