

9.2

IBM MQ Technický přehled

IBM

Poznámka

Než začnete používat tyto informace a produkt, který podporují, přečtěte si informace, které uvádí [“Poznámky” na stránce 295](#).

Toto vydání se vztahuje k verzi 9 vydání 2 produktu IBM® MQ a ke všem následujícím vydáním a modifikacím, dokud nebude v nových vydáních uvedeno jinak.

Když odešlete informace do IBM, udělíte společnosti IBM nevýlučné právo použít nebo distribuovat informace libovolným způsobem, který společnost považuje za odpovídající, bez vzniku jakýchkoliv závazků vůči vám.

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

Obsah

Technický přehled.....	5
Úvod do systému front zpráv.....	5
Hlavní vlastnosti a výhody front zpráv.....	7
Terminologie řazení zpráv.....	9
Zprávy a fronty.....	11
IBM MQObjekty.....	13
Typy objektů.....	14
Pojmenování objektů IBM MQ.....	32
Atributy objektu.....	38
Skupiny sdílení front.....	39
Systémové výchozí objekty.....	39
Distribuované fronty a klastry.....	39
Distribuované komponenty front.....	43
Komponenty klastru.....	52
Publikování/odběr zpráv.....	58
Publikování/odběr komponent.....	58
Příklad konfigurace publikování/odběru pro jednu správce front.....	83
Distribuované sítě typu publikování/odběr.....	84
IBM MQ Výběrové vysílání.....	101
Výchozí koncepce výběrového vysílání.....	102
Přehled produktu MQ Telemetry.....	103
Úvod do produktu MQ Telemetry.....	105
Příklady použití Telemetrie.....	106
Připojení telemetrických zařízení ke správci front.....	112
Protokoly připojení telemetrie.....	112
Služba (MQXR) telemetrie.....	112
Kanály telemetrie.....	113
IBM MQ Telemetry Transport protokol.....	113
MQTT klienti.....	113
Odeslání zprávy klientovi MQTT.....	114
Odeslání zprávy aplikaci IBM MQ z klienta MQTT.....	123
Aplikace pro publikování/odběr produktu MQTT.....	124
Aplikace Telemetrie.....	125
Integrace produktu MQ Telemetry se správcem front.....	125
MQTT bezstavové a stavové relace.....	127
Není-li klient MQTT připojen.....	128
Volné spojení mezi klienty MQTT a aplikacemi IBM MQ.....	128
Zabezpečení MQ Telemetry.....	129
MQ Telemetry globalizace.....	129
Výkon a rozšiřitelnost produktu MQ Telemetry.....	130
Zařízení podporovaná produktem MQ Telemetry.....	132
Zabezpečení v produktu IBM MQ.....	133
Podpora TLS spravovaného klienta IBM MQ.NET.....	134
Klienti a servery.....	134
Přehled produktu IBM MQ MQI clients.....	135
Správa a podpora transakcí.....	141
Rozšíření zařízení správce front.....	143
Rozhraní jazyka produktu IBM MQ Java.....	144
IBM MQ classes for JMS.....	144
IBM MQ Poskytovatel systému zpráv.....	152
IBM MQ for z/OS koncepce.....	152
Správce front v systému z/OS.....	153

Inicializátor kanálu na systému z/OS.....	154
Výrazy a úlohy pro správu IBM MQ for z/OS.....	156
Sdílené fronty a skupiny sdílení front.....	158
Použití front v rámci skupiny.....	203
Správa úložišť na systému z/OS.....	216
přihlášení IBM MQ for z/OS.....	220
Definování systému v systému z/OS.....	231
Zotavení a restartování v systému z/OS.....	242
Koncepce zabezpečení v produktu IBM MQ for z/OS.....	258
Dostupnost v systému z/OS.....	265
Monitorování a statistika v systému IBM MQ for z/OS.....	268
Dispozice jednotky zotavení v systému z/OS.....	269
IBM MQ a další produkty z/OS.....	272
IBM MQ a CICS.....	272
IBM MQ a IMS.....	273
IBM MQ a adaptéry dávky, TSO a RRS produktu z/OS.....	277
IBM MQ for z/OS a WebSphere Application Server.....	278
Managed File Transfer.....	279
Jak MFT pracuje s IBM MQ?.....	282
MFT přehled topologie.....	282
Přehled produktu MFTREST API.....	283
IBM MQ Internet Pass-Thru.....	283
Použití MQIPT.....	284
Jak MQIPT funguje.....	286
Možné konfigurace produktu MQIPT.....	287
Kompatibilní konfigurace.....	289
Podporované konfigurace kanálů.....	291
Podmínky ukončení a selhání kanálu.....	292
Bezpečnost zpráv.....	292
Správci front s více instancemi a vysoká dostupnost.....	292
Poznámky.....	295
Informace o programovacím rozhraní.....	296
Ochranné známky.....	296

IBM MQ Technický přehled

Použijte produkt IBM MQ k připojení aplikací a správě distribuce informací v rámci organizace.

Produkt IBM MQ umožňuje vzájemnou komunikaci programů v rámci sítě na rozdíl od komponent (procesorů, operačních systémů, subsystémů a komunikačních protokolů) pomocí konzistentního rozhraní API. Aplikace navržené a zapsané pomocí tohoto rozhraní jsou známy jako aplikace front zpráv.

Následující dílčí témata použijte k vyhledání informací o front zpráv a dalších funkcích, které poskytuje produkt IBM MQ.

Související pojmy

[Úvod do produktu IBM MQ](#)

Související úlohy

[Plánování architektury produktu IBM MQ](#)

Související odkazy

[“Hlavní vlastnosti a výhody front zpráv” na stránce 7](#)

Tyto informace zvýrazňují některé funkce a výhody ve frontě zpráv. Popisuje funkce, jako je například zabezpečení a integrita dat front zpráv.

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

Úvod do systému front zpráv

Produkty IBM MQ umožňují vzájemnou komunikaci programů v rámci sítě na rozdíl od komponent (procesorů, operačních systémů, subsystémů a komunikačních protokolů) pomocí konzistentního programovacího rozhraní aplikací.

Aplikace navržené a zapsané pomocí tohoto rozhraní jsou známy jako aplikace *front zpráv*, protože používají styl *zasílání zpráv a řazení do fronty*:

- Systém zpráv znamená, že programy komunikují zasíláním dat ve zprávách, a nevolají si navzájem přímo.
- Fronta znamená, že zprávy se umísťují do front v paměti, takže programy mohou běžet nezávisle na sobě, při různých rychlostech a časech, v různých lokalitách a bez nutnosti logického spojení mezi nimi.

Řazení zpráv do fronty bylo použito pro zpracování dat po mnoho let. Nejčastěji se používá dnes v elektronické poště. Bez zařazení do fronty vyžaduje odeslání elektronické zprávy na dlouhé vzdálenosti každý uzel na trase, který má být k dispozici pro předávání zpráv, a adresátům, kteří mají být přihlášení, a vědomé skutečnosti, že se jim snažíte odeslat zprávu. V systému řazení do front jsou zprávy uloženy v intermediačních uzlech, dokud je systém není připraven k jejich předání. Ve svém posledním místě určení jsou uloženy v elektronické poštovní schránce, dokud je adresát není připraven je číst.

I tak je dnes mnoho složitých obchodních transakcí zpracováno bez řazení do fronty. Ve velké síti by systém mohl udržovat mnoho tisíc připojení ve stavu přečteného do použití. Pokud jedna část systému trpí problémem, mnoho částí systému se stane nepoužitelným.

Můžete si představit, že systém front zpráv je používán jako elektronická pošta pro programy. V prostředí front zpráv provádí každý program, který tvoří součást sady aplikací, dobře definovanou, samostatnou funkci v odpovědi na konkrétní požadavek. Chcete-li komunikovat s jiným programem, program musí vložit zprávu do předdefinované fronty. Druhý program načte zprávu z fronty a zpracuje požadavky a informace obsažené ve zprávě. Řazení zpráv do front je tedy styl komunikace mezi programy.

Čekání na řazení zpráv do front je mechanismus, do kterého jsou zprávy drženy, dokud není aplikace připravena ke zpracování. 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.

- Vyvážení zatížení na systému tím, že uspořádáte více než jeden program pro obsluhu fronty, pokud počet zpráv překročí prahovou hodnotu.
- Zvyšte dostupnost aplikací tím, že uspořádáte alternativní systém pro obsluhu front, pokud váš primární systém není k dispozici.

Co je fronta zpráv?

Fronta zpráv, známá jednoduše jako fronta, je pojmenovaným místem určení, do kterého mohou být odesílány zprávy. Zprávy se hromadí ve frontách, dokud tyto fronty nenačtou programy, které tyto fronty obslouží.

Queues reside in, and are managed by, a queue manager, (see [“Terminologie řízení zpráv”](#) na stránce 9). Fyzikální povaha fronty závisí na operačním systému, v němž je správce front spuštěn. Fronta může být buď nestálá oblast vyrovnávací paměti v paměti počítače, nebo datovou sadou na trvalém úložném zařízení (jako je disk). Fyzická správa front je odpovědností správce front a není jasně viditelná pro zúčastněné aplikační programy.

Programy přistupují k frontám pouze prostřednictvím externích služeb správce front. Mohou otevřít frontu, vkládat zprávy do ní zprávy, získávat z ní zprávy a zavírat frontu. Mohou také nastavit a dotázat se na atributy front.

Různé styly front zpráv

dvoubodový

Jedna zpráva se umístí do fronty a jedna aplikace obdrží tuto zprávu.

V systému zpráv typu point-to-point musí odesílající aplikace znát informace o přijímající aplikaci před tím, než může odeslat zprávu do této aplikace. Odesílající aplikace může například potřebovat znát název fronty, do níž se mají odeslat informace, a může také zadat název správce front.

Publikování/odběr

Kopie každé zprávy publikovaná publikujícími aplikací je doručena do každé zainteresované aplikace. Může jich být mnoho, jedna nebo žádné zainteresované aplikace. Ve publish/subscribe je zúčastněná aplikace známá jako odběratel a zprávy jsou zařazeny do fronty ve frontě určené prostřednictvím odběru.

Systém zpráv typu publikování/odběr umožňuje oddělit poskytovatele informací od spotřebitelů o těchto informacích. Odesílající aplikace a přijímající aplikace nepotřebují vědět o sobě tolik informací, které by měly být zaslány a přijaty. Další informace viz [“Publikování/odběr zpráv”](#) na stránce 58.

Výhody front zpráv s návrhářem aplikací a vývojářem

Produkt IBM MQ umožňuje aplikačním programům využít *front zpráv* k účasti na zpracování zpráv. Aplikační programy mohou komunikovat přes různé platformy s použitím vhodných softwarových produktů pro řízení zpráv do front. Aplikace produktu z/OS mohou například komunikovat prostřednictvím produktu IBM MQ for z/OS. Aplikace jsou chráněny před mechanikou základní komunikace. Některé z dalších výhod front zpráv jsou:

- Aplikace je možné navrhovat pomocí malých programů, které můžete sdílet mezi mnoha aplikacemi.
- Nové aplikace můžete rychle sestavit opětovným použitím těchto stavebních bloků.
- Aplikace napsané tak, aby používaly techniky front zpráv, nejsou ovlivněny změnami ve způsobu, jakým správci front pracují.
- Není třeba používat žádné komunikační protokoly. Správce front pracuje se všemi aspekty komunikace pro vás.
- Programy, které přijímají zprávy, nemusí být spuštěny v době, kdy jim jsou zasílány zprávy. Zprávy jsou uchovány ve frontách.

Návrháři mohou snížit náklady na své aplikace, protože vývoj je rychlejší, je zapotřebí méně vývojářů a požadavky na schopnosti programování jsou nižší než schopnosti pro aplikace, které nepoužívají řazení zpráv do front.

Produkt IBM MQ implementuje běžné rozhraní API známé jako *rozhraní fronty zpráv* (nebo MQI) bez ohledu na to, kde jsou aplikace spuštěny. To usnadňuje portům aplikačních programů z jedné platformy do druhé.

Podrobnosti o rozhraní MQI naleznete v tématu [Přehled rozhraní fronty zpráv](#).

Hlavní vlastnosti a výhody front zpráv

Tyto informace zvýrazňují některé funkce a výhody ve frontě zpráv. Popisuje funkce, jako je například zabezpečení a integrita dat front zpráv.

Mezi hlavní funkce aplikací, které používají techniky front zpráv, patří:

- Mezi programy nejsou žádná přímá spojení.
- Komunikace mezi programy může být nezávislá na čase.
- Práce může být prováděna malými, samozamenými programy.
- Komunikace může být řízena událostmi.
- Aplikace mohou ke zprávě přiřadit prioritu.
- Ochranka.
- Integrita dat.
- Podpora obnovy.

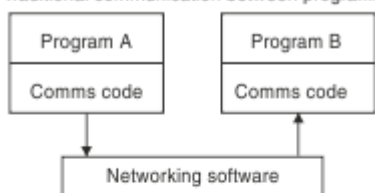
Žádné přímé spojení mezi programy

Řazení zpráv do front je technika pro nepřímou komunikaci mezi programem. Lze ji použít v rámci libovolné aplikace, ve které spolu programy komunikují. Komunikace probíhá jedním programem, který vkládá zprávy do fronty (ve vlastnictví správce front) a jiný program, který získává zprávy z fronty.

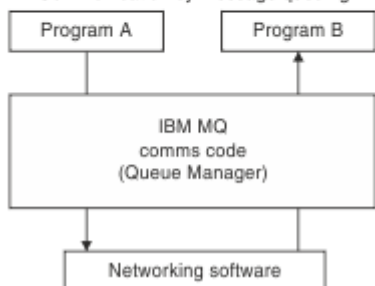
Programy mohou získat zprávy, které byly vloženy do fronty jinými programy. Ostatní programy mohou být připojeny ke stejnému správci front jako přijímající program nebo k jinému správci front. Tento jiný správce front může být na jiném systému, jiném počítačovém systému, nebo dokonce v rámci jiného podniku 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 7](#)).

Traditional communication between programs



Communication by message queuing



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

Jako u elektronické pošty se jedná o jednotlivé zprávy, které jsou součástí transakce v rámci sítě v úložišti a v daném úložišti základ. Dojde-li k selhání propojení mezi uzly, bude zpráva uchována, dokud se neobnoví odkaz, nebo pokud operátor nebo program tuto zprávu přeměrovává.

Mechanismus, pomocí kterého je zpráva přesunuta z fronty do fronty, je z programů skryta. Proto jsou tyto programy jednodušší.

Časově nezávislá komunikace

Programy, které požadují jiné úlohy k provedení práce, nemusí čekat na odpověď na požadavek. Mohou provádět jinou práci a zpracovat odpověď buď při příchodu, nebo později. Při zápisu aplikace systému zpráv není třeba znát (nebo být dotčen), když program odešle zprávu nebo když je cíl schopen přijmout zprávu. Zpráva nebyla ztracena. Je uchována správcem front, dokud není cíl připraven na zpracování. Zpráva zůstane ve frontě, dokud ji neodstraní programem. To znamená, že odesílající a přijímající aplikační programy jsou oddělené; odesílatel může pokračovat ve zpracování, aniž by čekal na potvrzení přijetí zprávy příjemcem. Cílová aplikace nemusí být spuštěna, když je zpráva odeslána. Může načíst zprávu po jejím spuštění.

Malé programy

Fronty zpráv vám umožňují využívat výhody používání malých, samoobslužných programů. Místo jednoho, velkého programu, který provádí všechny části úlohy sekvenčně, můžete úlohu šířit přes několik menších nezávislých programů. Žádající program odesílá zprávy každému ze samostatných programů a požádá je o jejich provedení; jakmile je každý program dokončen, výsledky se odešlou jako jedna nebo více zpráv.

Zpracovávání řízené zprávami

Když zprávy dorazí do fronty, mohou automaticky spustit aplikaci pomocí *spouštěče*. Je-li to nezbytné, mohou být aplikace zastaveny, když je zpráva (nebo zprávy) zpracována.

Zpracování řízené událostmi

Programy mohou být řízeny podle stavu front. Například můžete zařídit, aby se program spustil, jakmile zpráva dorazí do fronty, nebo můžete uvést, že se program nespustí, dokud se neobjeví například 10 zpráv nad určitou prioritou ve frontě, nebo 10 zpráv libovolné priority ve frontě.

Priorita zpráv

Program může při vložení zprávy do fronty přiřadit prioritu zprávy. Určuje pozici ve frontě, do níž je přidána nová zpráva.

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

Zabezpečení

K dispozici jsou služby zabezpečení, včetně ověření aplikací při použití správce front, autorizace při použití prostředků, jako je fronta ve správcích front, a šifrování dat zprávy při přenosu po síti a ukládání do front ve frontách. Další informace o zabezpečení najdete v tématu [Přehled zabezpečení](#).

Integrita dat

Integrita dat je poskytována jednotkami práce. Synchronizace začátku a konce jednotek práce je plně podporována jako volba na každém MQGET nebo MQPUT, což umožňuje potvrdit nebo odvolat výsledky transakce. Podpora synchronizačního bodu funguje buď interně, nebo externě na IBM MQ v závislosti na formě koordinace bodu synchronizace vybrané pro aplikaci.

Podpora obnovy

Má-li být zotavení možné, jsou protokolovány všechny trvalé aktualizace produktu IBM MQ. Je-li obnova nezbytná, všechny trvalé zprávy se obnoví, všechny transakce v průběhu transakce se odvolají a všechny operace commit a vrácení bodu synchronizace jsou zpracovány běžným způsobem správcem synchronizačních bodů v řízení. Další informace o trvalých zprávách naleznete v tématu [Perzistence zpráv](#).

Poznámka: Při zvažování klientů a serverů IBM MQ nemusíte měnit serverovou aplikaci pro podporu dalších IBM MQ MQI clients na nových platformách. Podobně platí, že produkt IBM MQ MQI client může bez změny fungovat s dalšími typy serverů.

Terminologie řazení zpráv

Tyto informace poskytují vhled do některých termínů používaných ve frontě zpráv.

Mezi ně patří:

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

Zpráva

Ve frontě zpráv se jedná o kolekci dat posílaných jedním programem a určených pro jiný program. Viz [Zprávy produktu IBM MQ](#). Chcete-li získat informace o typech zpráv, prohlédněte si téma [Typy zpráv](#).

deskriptor zprávy

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

Řídicí informace jsou definovány ve struktuře deskriptoru zpráv (MQMD) a obsahují takové věci jako:

- Typ zprávy
- Identifikátor pro zprávu
- Priorita doručení zprávy

Struktura a obsah dat aplikace jsou určovány zúčastněnými programy, nikoli produktem IBM MQ.

Fronta

Pojmenovaný cíl, do kterého mohou být odesílány zprávy. Zprávy se hromadí ve frontách, dokud tyto fronty nenačtou programy, které tyto fronty obslouží.

Správce front

Správce front je systémový program, který poskytuje služby systému front aplikacím.

Poskytuje rozhraní pro programování aplikací, takže programy mohou vkládat zprávy do front a získávat zprávy z nich. Správce front nabízí další funkce, aby administrátoři mohli vytvářet nové fronty, měnit vlastnosti stávajících front a řídit činnost správce front.

Aby bylo v systému k dispozici služby řazení zpráv do fronty zpráv produktu IBM MQ, 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 pomocí *manipulátoru připojení (Hconn)*.

Řadu různých aplikací může využívat služeb správce front současně a tyto aplikace mohou být zcela nesouvisející. Má-li program používat služby správce front, musí vytvořit připojení k tomuto správci front.

Aby aplikace mohly odesílat zprávy do aplikací, které jsou připojeny k jiným správcům front, musí být tito správci front schopni komunikovat mezi sebou. Produkt IBM MQ implementuje protokol *uložit-a-předat*, aby se zajistilo bezpečné doručování zpráv mezi těmito aplikacemi.

Kanály

Kanály jsou objekty, které poskytují komunikační cestu z jednoho správce front do jiného. Kanály se používají k přesouvání zpráv od jednoho správce front do jiného a zašití aplikace ze základních komunikačních protokolů. Správci front mohou existovat na stejném systému nebo na různých systémech na stejné platformě nebo na různých platformách. Odeslaná zpráva může pocházet z mnoha míst:

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

Agent oznamovacího kanálu

Agent oznamovacího kanálu je jeden konec kanálu. Dvojice agentů kanálu zpráv, jedna odesílající a jedna přijímající, vytvářejí kanál a přesouvají zprávy z jednoho správce front do jiného.

Klastr

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

V síti produktu IBM MQ, která používá distribuované ukládání do fronty bez klastrování, je každý správce front nezávislý. Pokud jeden správce front potřebuje odeslat zprávy jinému správci front, musí mít definovanou přenosovou frontu a kanál ke vzdálenému správci front.

Pro použití klastrů existují dva různé důvody: ke snížení administrace systému a ke zlepšení dostupnosti a vyrovnávání pracovní zátěže.

Jakmile zavedete i ten nejmenší klastr, získáte výhodu ze zjednodušené správy systému. Správci front, kteří jsou součástí klastru, potřebují méně definic a tím snižuje riziko, že dojde k chybě v definicích.

Další informace o klastrování najdete v tématu [Klastry](#).

Sdílená fronta

Sdílená fronta je typ lokální fronty se zprávami, ke kterým může přistupovat jeden nebo více správců front, kteří jsou v prostředí sysplex. To není stejné jako fronta, která je sdílána více než jednou aplikací, a to pomocí stejného správce front. Toto platí pouze pro IBM MQ for z/OS.

Skupina sdílení front

Správci front, kteří mohou přistupovat ke stejné sadě sdílených front, tvoří skupinu s názvem *skupina sdílení front* (QSG). Komunikovali spolu s prostředkem pro připojení (CF), který ukládá sdílené fronty. Toto platí pouze pro IBM MQ for z/OS. Další informace o skupinách sdílení front viz [“Sdílené fronty a skupiny sdílení front”](#) na stránce 158.





Použití front v rámci skupiny

Správci front v rámci skupiny sdílení front mohou komunikovat s použitím běžných kanálů nebo můžete použít techniku s názvem *intra-group queuing* (IGQ), která umožňuje provádět rychlý přenos zpráv bez definování kanálů. Toto platí pouze pro IBM MQ for z/OS.

IBM MQ MQI client

IBM MQ MQI *klienti* jsou nezávisle instalovatelné komponenty produktu IBM MQ. Klient MQI umožňuje spouštění aplikací produktu IBM MQ s komunikačním protokolem pro interakci s jedním nebo více servery MQI (Message Queue Interface) na jiných platformách a k připojení k jejich správcům front.

Úplné podrobnosti o tom, jak instalovat a používat komponenty produktu IBM MQ MQI client najdete v následujících tématech:

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

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

Dvoubodový systém zpráv

V systému zpráv typu point-to-point cestuje každá zpráva z jedné produkující aplikace do jedné náročné aplikace. Zprávy se přenášejí prostřednictvím aplikace, která vytváří zprávy do fronty a spotřebovávající aplikace je dostane z této fronty.

Publikování/odběr zpráv

V systému zasílání zpráv typu publikování/odběr je každá zúčastněná aplikace doručena kopie každé zprávy publikované publikující aplikací. Může jich být mnoho, jedna či žádná zainteresovaná aplikace. Ve publish/subscribe je zúčastněná aplikace známá jako odběratel a zprávy jsou zařazeny do fronty ve frontě určené prostřednictvím odběru. Další informace viz [“Publikování/odběr zpráv”](#) na stránce 58.

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 fronty ztotožní téma se seznamem odběratelů, kteří jsou přihlášení k jeho odběru, a doručí zprávu všem těmto odběratelům.

Předplatné

Aplikace typu publikování/odběr může zaregistrovat zájem o zprávy o specifických tématech. Když aplikace to dělá, je známá jako odběratel a termín odběru definuje, jak jsou odpovídající zprávy zařazeny do fronty pro zpracování.

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

Zprávy a fronty

Zprávy a fronty jsou základní komponenty systému front zpráv.

Co je zpráva?

Zpráva je řetězec bajtů, který je srozumitelný pro aplikace, které jej používají. Zprávy se používají pro přenos 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 ji používají.
- *Deskriptor zprávy*. Deskriptor zpráv identifikuje zprávu a obsahuje další řídicí informace, jako např. typ zprávy a prioritu přiřazené zprávě odesílající aplikací.

Formát deskriptoru zpráv je definován produktem IBM MQ. Úplný popis deskriptoru zpráv viz [MQMD-Message descriptor](#).

- *Vlastnosti zprávy*. Meta-data o zprávě. Obsah vlastností zprávy je definován aplikačními programy, které je používají. Další informace naleznete v tématu [Vlastnosti zprávy](#).

Délky zpráv

Výchozí maximální délka zprávy je 4 MB, ačkoli ji můžete zvýšit až 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 přijímající frontu
- Maximální délka zprávy definovaná pro správce front
- Maximální délka zprávy definovaná frontou
- Maximální délka zprávy definovaná v odesílající nebo přijímající aplikaci
- Množství úložiště dostupné pro zprávu

K odeslání všech informací, které aplikace vyžaduje, může dojít k odeslání několika 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 za účelem vložení zprávy do fronty.

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

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

Co je 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 své běžné činnosti.

Předdefinované fronty a dynamické fronty

Fronty mohou být charakterizovány tak, jak jsou vytvořeny:

- **Předdefinované fronty** vytváří administrátor pomocí vhodných příkazů MQSC nebo PCF. Předdefinované fronty jsou trvalé; existují nezávisle na aplikacích, které je používají a přežijí IBM MQ restartů.
- **Dynamické fronty** se vytvářejí při aplikaci vydá požadavek MQOPEN , který uvádí název *modelové fronty*. Vytvořená fronta je založena na *definici modelové fronty*, která se nazývá modelová fronta. Frontu modelu můžete vytvořit pomocí příkazu MQSC DEFINE QMODEL. Atributy modelové fronty (například maximální počet zpráv, které lze uložit) jsou zděděny libovolnou dynamickou frontou, která je z ní vytvořena.

Modelové fronty mají atribut, který uvádí, zda dynamická fronta má být trvalá nebo dočasná. Trvalé fronty přežijí aplikace a správce front se restartuje; dočasné fronty se ztratí při restartu.

Načítání zpráv z front

Autorizované autorizované aplikace mohou načítat zprávy z fronty podle následujících algoritmů pro načítání:

- FIFO (First-in-first-out).

- Priorita zprávy, jak je definováno v deskriptoru zpráv. Zprávy, které mají stejnou prioritu, jsou načítány na bázi FIFO.
- Požadavek programu na specifickou zprávu.

Použitá metoda určuje požadavek MQGET z aplikace.

IBM MQObjekty

Správci front definují vlastnosti objektů produktu IBM MQ . Hodnoty těchto vlastností ovlivňují způsob, jakým produkt IBM MQ zpracovává tyto objekty. Objekty vytváříte a spravujete pomocí příkazů a rozhraní produktu IBM MQ . Ve svých aplikacích můžete řídit objekty pomocí rozhraní MQI (Message Queue Interface). Objekty jsou identifikovány objektem IBM MQ *object descriptor* (MQOD), je-li adresován z programu.




Administrace objektů zahrnuje následující úlohy:

- Spouštění a zastavování správců front.
- Vytváření objektů, zejména front, pro aplikace.
- Zobrazení nebo změna atributů objektů.
- Probíhá odstraňování objektů.
- Práce s kanály pro vytváření komunikačních cest ke správcům front na jiných (vzdálených) systémech.
- Vytváření *klastrů* správců front pro zjednodušení celkového administrativního procesu a vyvážení pracovní zátěže.

S výjimkou dynamických front je třeba objekty pro správce front definovat dříve, než s nimi budete moci pracovat.


Použijete-li příkaz IBM MQ k provedení administrační operace objektu, zkontroluje správce front, zda máte k provedení operace požadovanou úroveň oprávnění. Podobně platí, že pokud aplikace používá volání MQOPEN k otevření objektu, správce front zkontroluje, zda má aplikace požadovanou úroveň oprávnění před tím, než povolí přístup k danému objektu. Kontroly jsou prováděny na jménu otevíraný objekt.


Objekty můžete definovat a spravovat pomocí následujících metod:

- Příkazy PCF popsané v [Referenční příručce formátovaných příkazů](#) a [Automatizace administračních úloh](#)
- Příkazy MQSC popsané v části [Příkazy MQSC](#)
-  Ovládací panely a ovládací panely produktu IBM MQ for z/OS jsou popsány v části [Provoz IBM MQ for z/OS](#) .
-   Příkaz IBM MQ Explorer (Windows a Linux pouze pro systémy Intel). Další informace naleznete v tématu [Úvod do produktu MQ Explorer](#).

Objekty můžete spravovat také pomocí následujících metod:

- Řídící příkazy, které se zadávají z klávesnice. Viz [Správa pomocí řídicích příkazů](#).
- Volání produktu IBM MQ Administration Interface (MQAI) v programu. Viz rozhraní [IBM MQ Administration Interface \(MQAI\)](#).

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

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

Pro posloupanosti příkazů IBM MQ v systému AIX, Linux, and Windows můžete použít prostředek MQSC pro spuštění řady příkazů uložených v souboru. Další informace najdete v tématu [Administrace pomocí příkazů MQSC](#).

Typy objektů

Mnohé z administračních úloh zahrnují manipulaci s různými typy objektů IBM MQ *objects*.

Informace o pojmenování objektů IBM MQ naleznete v příručce [“Pojmenování objektů IBM MQ”](#) na stránce 32.

Informace o výchozích objektech vytvořených ve správci front naleznete v tématu [“Systémové výchozí objekty”](#) na stránce 39.

Informace o různých typech objektů produktu IBM MQ naleznete v následujících dílčích tématech:

Související pojmy

[“Úvod do systému front zpráv”](#) na stránce 5

Produkty IBM MQ umožňují vzájemnou komunikaci programů v rámci sítě na rozdíl od komponent (procesorů, operačních systémů, subsystémů a komunikačních protokolů) pomocí konzistentního programovacího rozhraní aplikací.

[“Atributy objektu”](#) na stránce 38

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

Související odkazy

[Příkazy MQSC](#)

Fronty

Úvod do front produktu IBM MQ a atributů fronty.

IBM MQ *fronta* je pojmenovaný objekt, na kterém mohou aplikace vkládat zprávy a ze kterých mohou aplikace získávat zprávy.

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

Dříve, než bude možné zprávu vložit do fronty, musí fronta již být vytvořena. Frontu vlastní správce front a tento správce front může vlastnit mnoho front. Každá fronta však musí mít název, který je jedinečný v rámci daného správce front.

Fronta se udržuje prostřednictvím správce front. Ve většině případů je každá fronta fyzicky spravována svým správcem front, ale to není zjevné pro aplikační program. IBM MQ for z/OS sdílené fronty může být spravováno libovolným správcem front ve skupině sdílení front.

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

Lokální fronty pro dočasné úlohy můžete vytvářet *dynamicky* z vaší aplikace. Můžete například vytvořit *reply-to* fronty (které nejsou potřeba po ukončení aplikace). Další informace viz [“Dynamické a modelové fronty”](#) na stránce 19.

Před použitím fronty je třeba frontu otevřít a uvést, co s ní chcete dělat. Můžete například otevřít frontu pro:

- Pouze procházet zprávy (nenačítat je)
- Načítání zpráv (a buď sdílení přístupu s jinými programy nebo s výlučným přístupem)
- Vložení zpráv do fronty
- Ukončení platnosti informací o attributech fronty
- Nastavení atributů fronty

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

Atributy front

Některé z atributů fronty jsou určeny, když je fronta definována, a nelze ji později změnit (například typ fronty). Dalšími atributy front lze seskupit do těch, které lze změnit:

- Ve správci front během zpracování fronty (například aktuální hloubka fronty).
- Pouze pro příkazy (například textový popis fronty)
- Pomocí volání MQSET (například, zda jsou povoleny operace vložení ve frontě)

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

Určuje, zda mají programy povoleno získávat zprávy z fronty. Zprávy ze vzdálených front však nikdy nelze načíst.

InhibitPut

Určuje, zda mají programy povoleno vkládat zprávy do fronty.

DefPriority

Výchozí priorita pro zprávy vkládané do fronty.

DefPersistence

Výchozí trvalost pro zprávy zařazené 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 najdete v tématu [Atributy pro fronty](#).

Související pojmy

 Sdílené fronty

Sdílená fronta je typem lokální fronty. K zprávám v této frontě může přistupovat jeden nebo více správců front, kteří jsou v prostředí sysplex.

“Fronty klastru” na stránce 54

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

“Fronty nedoručených zpráv” na stránce 46

Fronta nedoručených zpráv (nebo fronta nedoručených zpráv) je fronta, do níž jsou odesílány zprávy, pokud je nelze směřovat na jejich správné místo určení. Pro každého správce front je obvykle fronta nedoručených zpráv.

Související úlohy

[Odkaz na vývoj aplikací](#)

Související odkazy

[Příkazy MQSC](#)

“Porovnání mezi sdílenými frontami a frontami klastru” na stránce 54

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

Lokální fronty

Přenos, inicializace, nedoručené písmeno, příkaz, výchozí, kanál 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

Odešle-li aplikace zprávu do vzdálené fronty, lokální správce front uloží tuto zprávu do speciální lokální fronty, která se nazývá *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 na atributu XMITQ lokální definice vzdálené fronty.
2. Přenosová fronta se stejným názvem jako vzdálený správce front. Tato hodnota je výchozí hodnotou atributu XMITQ lokální definice vzdálené fronty.
3. Přenosová fronta uvedená na atributu DEFXMITQ lokálního správce front.

Agent kanálu zpráv je přidružený k přenosové frontě a doručí zprávu do dalšího cíle. Další místo určení je správce front, ke kterému je kanál zpráv připojen. Není to nutně stejný správce front jako konečný cíl zprávy. Je-li zpráva doručena do dalšího místa určení, je odstraněna z přenosové fronty. Je možné, že zpráva bude muset projít mnoha správci front na své cestě do svého konečného cíle. Musíte definovat přenosovou frontu na každém správci front po trase, přičemž každá zadržovací zpráva čeká na přenos do dalšího místa určení. Normální přenosová fronta zadržuje zprávy pro další cíl, ačkoli zprávy mohou mít odlišná konečná místa určení. Přenosová fronta klastru uchovává zprávy pro více cílů. `correlID` každé zprávy identifikuje kanál, na který je zpráva převedena, aby ji přenesl na další místo určení.

Ve správci front je možné definovat několik přenosových front. Pro stejné místo určení můžete definovat několik přenosových front, přičemž každá z nich bude použita pro jinou provozní třídu. Například můžete chtít vytvořit různé přenosové fronty pro malé zprávy a velké zprávy, které se budou provádět na stejném místě určení. Pak můžete zprávy přenést pomocí různých kanálů zpráv, takže velké zprávy nebudou obsahovat menší zprávy. Ve výchozím nastavení jsou všechny zprávy do front klastru nebo témat klastru umístěny do jedné přenosové fronty klastru `SYSTEM.CLUSTER.TRANSMIT.QUEUE`. Jako volbu můžete změnit výchozí nastavení a oddělit provoz zpráv pro různé správce front klastru na různé přenosové fronty klastru. Pokud nastavíte atribut správce front `DEFCLXQ` na hodnotu `CHANNEL`, bude každý odesílací kanál klastru vytvářet samostatnou přenosovou frontu klastru. Jako alternativu můžete ručně definovat přenosové fronty klastru pro kanály odesílatele klastru, které se mají použít.

Přenosové fronty mohou spustit agenta kanálu zpráv k odesílání zpráv dále; viz téma [Spuštění aplikací produktu IBM MQ pomocí spouštěčů](#).

 Pokud v systému IBM MQ for z/OS používáte řazení do front v rámci skupiny, je pro přenosové fronty obslužen *agent 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 vloží zprávu spouštěče, když dojde k události spouštěče v aplikační frontě.

Událost spouštěče je událost, která má za cíl způsobit, že program začne zpracovávat frontu. Například, událost může být více než 10 příchozích zpráv. Další informace o tom, jak spouštění funguje, najdete v tématu [Spuštění aplikací produktu IBM MQ pomocí spouštěčů](#).

Fronta nedoručených zpráv (nedoručená zpráva)


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

Když správce front vloží zprávu do fronty nedoručených zpráv, přidá k této zprávě záhlaví. Informace v záhlaví obsahují informace o příčině vložení zprávy do fronty nedoručených zpráv do fronty nedoručených zpráv. Také obsahuje 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 \(nedoručené zprávy\)](#).

Systémová fronta příkazů

Fronta příkazů systému je fronta, do níž mohou vhodné autorizované aplikace odesílat příkazy IBM MQ. Tyto fronty přijímají příkazy PCF, MQSC a CL, které jsou podporovány na vaší platformě, a jsou připraveny pro správce front, aby je mohli provést.

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

Výchozí systémové fronty

Systémové výchozí fronty obsahují počáteční definice front pro váš systém. Pokud vytvoříte definici fronty, správce front zkopíruje definici z příslušné výchozí systémové fronty. Vytvoření definice fronty se liší od vytvoření dynamické fronty. Definice dynamické fronty je založena na modelové frontě, kterou jste vybrali jako šablonu pro dynamickou frontu.

fronty událostí

Fronty událostí obsahují zprávy událostí. Tyto zprávy jsou vykazovány správcem front nebo kanálem.

Vzdálené fronty

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

Pokud byl vytvořen komunikační spoj, 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 potřebné pro lokálního správce front k vyhledání fronty, do níž má vaše zpráva jít. Tento objekt je známý jako *lokální definice vzdálené fronty*. Všechny atributy vzdálené fronty jsou zadrženy správcem front, který ji vlastní, protože se jedná o lokální frontu pro daného správce front.

Při otevírání vzdálené fronty pro identifikaci fronty je třeba určit jednu z následujících možností:

- Název lokální definice, která definuje vzdálenou frontu. Z pohledu aplikace je to stejné jako při otevírání lokální fronty. Aplikace nemusí vědět, zda je fronta lokální nebo vzdálená.

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

 V systému IBM i použijte příkaz `CRTMQMQ`.

- Název vzdáleného správce front a název fronty, jak je znám pro daného vzdáleného správce front.

Lokální definice vzdálených front mají kromě obecných atributů popsanych v [“Atributy front”](#) na stránce 15 také tři atributy. Tyto tři atributy jsou:

RemoteQName

Název, pod kterým ji správce front, který je vlastníkem fronty, zná.

RemoteQmgrName

Název vlastníčího 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 proti lokální definici vzdálené fronty, správce front vrátí atributy pouze lokální definice, tj. název vzdálené fronty, název vzdáleného správce front a název přenosové fronty, nikoli atributy odpovídající lokální fronty ve vzdáleném systému.

Viz též [Transmission queues](#).

Alias fronty

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

Fronta, která je výsledkem rozlišení názvu aliasu, známého jako základní fronta, může být kterýkoli z následujících typů front, jak je podporován platformou:

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

Název aliasu může být také možné interpretovat jako téma. Pokud aplikace aktuálně vkládá zprávy do fronty, lze ji publikovat do tématu tak, že název fronty bude pojmenováním alias pro dané téma. Nevyžaduje se žádná změna kódu aplikace.

Poznámka: Alias nemůže být přímo možné přeložit na jiný alias ve stejném správcí front.

Příkladem použití alias front alias je, aby administrátor systému udělil různým přístupům oprávnění k názvu základní fronty (tj. frontu, do níž je určen alias), a k názvu aliasu fronty. To znamená, že program nebo uživatel může být oprávněn používat alias frontu, ale ne základní frontu.

Případně může být autorizace nastavena tak, aby blokovala operace vložení pro alias, ale povoluje je pro základní frontu.

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

Produkt IBM MQ při pokusu o použití tohoto názvu provádí kontrolu autorizace s použitím aliasu. Nekontroluje, zda je program autorizován pro přístup k názvu, pro který je alias vyřešen. Program proto může být autorizován pro přístup k názvu alias fronty, ale ne k rozlišnému názvu fronty.

Kromě obecných atributů fronty popsaných v [“Fronty” na stránce 14](#) mají alias fronty také atribut **BaseQName** . Jedná se o název základní fronty, na kterou se rozlišuje název aliasu. Podrobný popis tohoto atributu naleznete v tématu [BaseQName \(MQCHAR48\)](#).

Atributy *InhibitGet* a **InhibitPut** (viz [“Fronty” na stránce 14](#)) alias front alias patří k názvu aliasu. Je-li například název alias fronty ALIAS1 interpretován jako název základní fronty BASE, poruchy v systému ALIAS1 ovlivní pouze ALIAS1 a BASE není blokováno. Avšak zábrany na BASE mají také vliv na ALIAS1.

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


Dynamické a modelové fronty

Tyto informace poskytují přehled o dynamických frontách, vlastnostech dočasných a trvalých dynamických front, použití dynamických front, některé pokyny při použití dynamických front a modelových front.

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

Vlastnosti dočasných dynamických front

Dočasné dynamické fronty mají následující vlastnosti:

-  Nemohou být sdílené fronty, které jsou přístupné ze správců front ve skupině sdílení front.
Všimněte si, že skupiny sdílení front jsou k dispozici pouze v produktu IBM MQ for z/OS.
- Jsou zadrženy pouze přechodné zprávy.
- Jsou nezotavitelné.
- Vymažou se při spuštění správce front.
- Odstraňují se, když aplikace, která vydala volání MQOPEN, která vytvořila frontu, zavře frontu nebo se ukončí.
 - Pokud ve frontě existují potvrzené zprávy, budou odstraněny.
 - Pokud v této chvíli existují nepotvrzené příkazy MQGET, MQPUT nebo MQPUT1 vůči frontě, je tato fronta označena jako logicky odstraněná a je fyzicky odstraněna (poté, co byla tato volání potvrzena) jako součást zpracování zavření nebo při ukončení aplikace.
 - Je-li fronta v současné době používána (při vytvoření nebo jiné aplikaci), je fronta označena jako logicky odstraněná a je fyzicky odstraněna pouze při zavření poslední aplikací pomocí fronty.
 - Pokusy o přístup k logicky odstraněné fronty (jiné než k zavření) se nezdařily s kódem příčiny MQRC_Q_DELETED.
 - MQCO_NONE, MQCO_DELETE a MQCO_DELETE_PURGE jsou všechny považovány za volání MQCO_NONE, jsou-li zadány při 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.
- Odstraňují se při úspěšném zavření fronty pomocí volby MQCO_DELETE nebo MQCO_DELETEPURGE nebo MQCO_DELETE nebo MQCO_DELETE_MQCO_DELETE_MQOPEN.
 - Zavření požadavku s volbou MQCO_DELETE selže, pokud se ve frontě nacházejí nějaké zprávy (potvrzené nebo nepotvrzené). Zavření požadavku s volbou MQCO_DELETE_PURGE uspěje i v případě, že ve frontě jsou potvrzené zprávy (zprávy jsou odstraněny jako součást zavření), ale selžou, pokud existují nepotvrzené příkazy MQGET, MQPUT nebo MQPUT1 pro nevyřízené volání.
 - Je-li požadavek na odstranění úspěšný, ale fronta je používána (vytvořením nebo jinou aplikací), je fronta označena jako logicky odstraněná a je fyzicky odstraněna pouze při zavření poslední aplikací pomocí fronty.
- Neodstraní se, pokud je zavřena aplikací, která není autorizována k odstranění fronty, pokud uzavírací aplikace nevydala volání MQOPEN, které vytvořilo frontu. Kontroly autorizace se provádějí proti identifikátoru uživatele (nebo alternativního identifikátoru uživatele, pokud byl zadán parametr MQOO_ALTERNATE_USER_AUTHORITY), který byl použit k ověření platnosti odpovídajícího volání MQOPEN.
- Mohou být odstraněny stejným způsobem jako normální fronta.

Použití dynamických front

Můžete použít dynamické fronty pro:

- Aplikace, které nevyžadují uchování front po ukončení aplikace.
- Aplikace, které vyžadují odpovědi na zprávy, které mají být zpracovány jinou aplikací. Takové aplikace mohou dynamicky vytvářet frontu pro odpověď tím, že otevřou modelovou frontu. Klientská aplikace může například:
 1. Vytvořit dynamickou frontu.
 2. Zadejte jeho název do pole **ReplyToQ** struktury deskriptoru zpráv ve zprávě požadavku.
 3. Umístěte požadavek do fronty zpracovávané serverem.

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

Aspekty použití dynamických front

Při použití dynamických front zvažte následující body:

- V modelu klient-server musí každý klient vytvořit a používat svou vlastní dynamickou odpověď-do fronty. Je-li dynamická fronta odpovědi na frontu sdílána mezi více než jedním klientem, může být odstranění fronty pro odpovědi zpožděno, protože ve frontě není nevyřízena nepotvrzená aktivita vůči frontě, nebo protože fronta je používána jiným klientem. Kromě toho může být fronta označena jako logicky odstraněná a nepřístupná pro následné požadavky rozhraní API (jiné než MQCLOSE).
- Pokud prostředí aplikace vyžaduje sdílení dynamických front mezi aplikacemi, ujistěte se, že je tato fronta uzavřena (s volbou odstranění), pokud byla potvrzena veškerá aktivita vůči frontě. Toto by měl být posledním uživatelem. Tím je zajištěno, že odstranění fronty nebude zpožděno, a minimalizuje období, kdy je fronta nedostupná, protože byla označena jako logicky odstraněná.

Modelové fronty

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

Lokální frontu lze vytvořit dynamicky z programu IBM MQ a pojmenováváte frontu modelu, kterou chcete použít jako šablonu pro atributy fronty. V tomto okamžiku můžete změnit některé atributy nové fronty. Nemůžete však změnit **DefinitionType**. Pokud například vyžadujete trvalou frontu, vyberte modelovou frontu s typem definice nastaveným na hodnotu permanent (trvalé). Některé konverzační aplikace mohou používat dynamické fronty k uchování odpovědi na své dotazy, protože pravděpodobně nepotřebují tyto fronty udržovat poté, co zpracovaly odpovědi.

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

Můžete zadat název (plný) dynamické fronty nebo kmen názvu (například ABC) a nechat správce front přidat k této jedinečné části, nebo můžete nechat správce front, aby pro vás přiřadila úplný jedinečný název. Pokud název správce front přiřadí název, vloží jej do struktury MQOD.

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

MQSET a MQINQ nelze vydat pro modelovou frontu. Otevření modelové fronty s výsledky MQOO_INQUIRE nebo MQOO_SET v následných voláních MQINQ a MQSET vytvářených proti dynamicky vytvořené frontě.

Atributy modelové fronty jsou podmnožinou objektů z lokální fronty. Podrobný popis naleznete v tématu [Atributy pro fronty](#).

Způsoby definování front

Frontu můžete vytvořit pomocí příkazů IBM MQ (MQSC), PCF nebo pomocí rozhraní specifických pro platformu.

Fronty můžete definovat do produktu IBM MQ pomocí příkazu MQSC [DEFINE](#) nebo PCF [Vytvořit frontu](#) . Příkazy určují typ fronty a její atributy. Např. lokální objekt fronty má atributy, které určují, co se stane, když aplikace odkazují na frontu v voláních MQI. Příklady atributů jsou:

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

K dispozici jsou také různá rozhraní specifická pro platformu, která můžete použít k definování front. Odkazy na další informace naleznete v tématu [Související informace](#).

Související pojmy

[Administrace pomocí příkazů MQSC](#)

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

Související úlohy

[Automatizace administrace pomocí příkazů PCF](#)

[Vytváření a konfigurování správců front a objektů s produktem MQ Explorer](#)

[Správa produktu IBM MQ for IBM i pomocí příkazů jazyka CL](#)

[Vydávání příkazů do produktu IBM MQ for z/OS](#)

Fronty použité pro specifické účely produktem IBM MQ

Produkt IBM MQ používá některé lokální fronty pro specifické účely související s její operací.

Tyto fronty je třeba definovat dříve, než je produkt IBM MQ může použít.

Inicializační fronty

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

Má-li správce front použít spouštěcí program, musí být pro daného správce front definována alespoň jedna inicializační fronta. Viz [Správa objektů pro spouštění](#), [runmqtrma](#) [Spuštění aplikací produktu IBM MQ pomocí spouštěčů](#)

Přenosové fronty

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

Každý správce front může mít výchozí přenosovou frontu. Pokud správce front, který není součástí klastru, vloží zprávu do vzdálené fronty, použije se výchozí akce pro použití výchozí přenosové fronty. Existuje-li přenosová fronta se stejným názvem jako správce cílové fronty, bude zpráva vložena do této přenosové fronty. Pokud se jedná o definici aliasu správce front, ve které parametr **RQMNAME** odpovídá cílovému správci front, a je zadán parametr **XMITQ** , zpráva se umístí do přenosové fronty pojmenované **XMITQ**. Pokud parametr **XMITQ** neexistuje, 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 se při definování správce front vytvářejí při výchozím nastavení. Je-li atribut správce front **DEFCLXQ** nastaven na hodnotu **CHANNEL**, je

pro každý vytvářený odesílací kanál klastru automaticky vytvořena trvalá dynamická přenosová fronta klastru. Fronty se nazývají `SYSTEM.CLUSTER.TRANSMIT.ChanneName`. Přenosové fronty klastru můžete také definovat ručně.

Správce front, který je součástí klastru, odesílá zprávy v jedné z těchto front do jiných správců front, kteří jsou ve stejném klastru.

Během rozlišování názvů má přednost přenosové fronty klastru přednost před výchozí přenosovou frontou a určitá přenosová fronta klastru má přednost před `SYSTEM.CLUSTER.TRANSMIT.QUEUE`.

Fronty nedoručených zpráv

Fronta nedoručených zpráv (nedoručená zpráva) je fronta, do níž jsou ukládány zprávy, které nelze směřovat na jejich správné cíle. Zprávu nelze směřovat, je-li například plná cílová fronta. Dodaná fronta nedoručených zpráv se nazývá `SYSTEM.DEAD.LETTER.QUEUE`.

V případě distribuovaných front definujte frontu nedoručených zpráv pro všechny zúčastněné správce front.

Fronty příkazů

Fronta příkazů, `SYSTEM.ADMIN.COMMAND.QUEUE`, je lokální fronta, do které mohou vhodné autorizované aplikace odesílat příkazy MQSC pro zpracování. Tyto příkazy jsou poté načteny komponentou produktu IBM MQ, která se nazývá příkazový server. Příkazový server ověří příkazy, předá platné příkazy pro zpracování správcem front a vrátí všechny odpovědi na příslušnou frontu pro odpověď.

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 zpět zprávu s odpovědí na odesílající aplikaci. Tato zpráva je vložena do fronty s názvem odpovědi na frontu, která je obvykle lokální frontou pro odesílající aplikaci. Název fronty pro odpověď je určen odesílající aplikací jako součást deskriptoru zpráv.

fronty událostí

Údálosti přípravy nástrojů lze použít k monitorování správců front nezávisle na aplikacích MQI.

Když dojde k události instrumentace, správce front vloží zprávu události do fronty událostí. Tuto zprávu pak může číst aplikace monitorování, která může informovat administrátora nebo zahájit nápravnou akci, pokud událost označuje problém.

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

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

IBM MQ Správci front

Úvod do *správce front* a do front služeb, které poskytují aplikacím.

Program musí mít k dispozici připojení ke správci front, než bude moci používat služby tohoto správce front. Program může toto spojení explicitně vyvolat (pomocí volání `MQCONN` nebo `MQCONNX`), nebo může být připojení implicitně vytvořeno (závisí na platformě a prostředí, ve kterém je program spuštěn).

Správci front poskytují aplikacím zařazování služeb do front a spravují fronty, které k nim patří. Správce front zajišťuje následující akce:

- Atributy objektu se mění v souladu s přijatými příkazy.
- Speciální události, jako jsou události spouštěče nebo události přípravy nástrojů, jsou generovány, když jsou splněny příslušné podmínky.
- Zprávy jsou vloženy do správné fronty, jak je požadováno aplikací, která provádí volání `MQPUT`. Aplikace je informována, pokud to nelze provést, a je dán příslušný kód příčiny.

Každá fronta patří do jednoho správce front a říká se, že se jedná o *lokální frontu* pro daného správce front. Správce front, ke kterému je aplikace připojena, je údajně *lokální správce front* pro danou aplikaci. Pro aplikaci jsou fronty, které patří do svého lokálního správce front, lokální fronty.


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

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


Atributy správců front

Přidruženo ke každému správci front je sada atributů (nebo vlastností), které definují své charakteristiky. Některé atributy správce front jsou při jeho vytvoření opraveny; můžete je změnit pomocí příkazů IBM MQ. Můžete se dotázat na hodnoty všech atributů s výjimkou těch, které se používají pro šifrování TLS (Transport Layer Security), pomocí volání MQINQ.

Mezi pevné atributy patří následující:

- Název správce front
- Platforma, na které je správce front spuštěn (například, Windows)
- Úroveň příkazů řízení systému, které správce front podporuje.
- Maximální priorita, kterou můžete přiřadit ke zprávám zpracovaným správcem front
- Název fronty, do níž programy mohou odesílat příkazy IBM MQ
- Maximální délka zpráv, které správce front může zpracovat  (neměnné pouze v produktu IBM MQ for z/OS)
- Určuje, zda správce front podporuje při vkládání zpráv a získávání zpráv synchronizační body.

Mezi atributy *changeable* patří:

- 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
-  Časový interval, který správce front používá k určení, jak často se mají fronty skenovat kvůli vypršelym zprávám (pouze IBM MQ for z/OS).
- Název fronty nedoručených zpráv správce front (nedoručená zpráva)
- Název výchozí přenosové fronty správce front
- Maximální počet otevřených popisovačů pro libovolné připojení
- Povolení a zakázání různých kategorií sestav událostí
- Maximální počet nepotvrzených zpráv v rámci jednotky práce

Správci front a správa pracovní zátěže

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

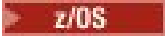
Definice procesů

Objekty definic procesů umožňují spouštění aplikací bez nutnosti zásahu operátora definováním atributů aplikace, které má správce front používat.

Objekt definice procesu definuje aplikaci, která se spustí jako odezva na událost spouštěče ve správci front IBM MQ. Mezi atributy definice procesu patří ID aplikace, typ aplikace a data specifická pro aplikaci.

Další informace najdete v tématu *Inicializační fronty* v příručce “[Fronty použité pro specifické účely produktem IBM MQ](#)” na stránce 21.

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

Atribut **ProcessName** je pevný při vytvoření objektu. Ostatní atributy však můžete měnit pomocí příkazů IBM MQ .  Případně můžete na serveru z/OS použít operace IBM MQ for z/OS a ovládací panely.

Můžete se dotázat na hodnoty všech atributů pomocí [MQINQ-Dotaz](#) na atributy objektu.

Související odkazy


[Atributy pro definice procesu](#)

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 jiných objektů produktu IBM MQ . Obvykle se seznamy názvů používají v aplikacích, jako jsou monitory spouštěčů, k identifikaci skupiny front. Výhodou používání seznamu názvů je, že je udržován nezávisle na aplikacích; lze jej aktualizovat bez zastavení aplikací, které jej používají. Také pokud jedna aplikace selže, seznam názvů není ovlivněn a jiné aplikace jej mohou nadále používat.

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

Můžete definovat a upravit seznamy názvů pomocí příkazů MQSC.  Alternativně můžete na serveru z/OS použít operace IBM MQ for z/OS a ovládací panely.

Programy mohou prostřednictvím rozhraní MQI zjistit, které fronty jsou obsaženy v těchto seznamech názvů. Uspořádání názvů je zodpovědností návrháře aplikací a administrátora systému.

Související odkazy

[Atributy pro seznamy názvů](#)

[DEFINOVAT SEZNAM NÁZVŮ](#)

Objekty ověřovacích informací

Objekt ověřovacích informací poskytuje definice požadované k provedení kontroly odvolání certifikátů.

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

Chcete-li definovat objekt ověřovacích informací, můžete použít příkaz MQSC **DEFINE AUTHINFO** . Další informace o attributech objektů ověřovacích informací najdete v tématu [DEFINE AUTHINFO](#).

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

- [setmqaut](#) (udělit nebo odvolat oprávnění)
- [dspmqaut](#) (oprávnění k zobrazení objektu)
- [dmpmqaut](#) (autorizace výpisu)
- [rcrmqobj](#) (znovu vytvořit objekt)
- [rcdmqing](#) (obraz média záznamu)
- [dspmqfls](#) (zobrazit názvy souborů)

Přehled zabezpečení TLS a použití objektů ověřovacích informací najdete v tématu [Protokoly zabezpečení TLS v produktu IBM MQ](#).

Související pojmy

[Koncepte zabezpečení přenosové vrstvy \(TLS\)](#)

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).

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 vytváření objektu COMMINFO naleznete v tématu [Začínáme s výběrovým vysíláním](#).

Související pojmy

[“IBM MQ Výběrové vysílání” na stránce 101](#)

Výběrové vysílání produktu IBM MQ nabízí spolehlivé výběrové vysílání zpráv s nízkou latencí a vysokým stupněm větvení.

Kanály

Kanál je logický komunikační spoj používaný distribuovanými správci front, mezi serverem IBM MQ MQI client a serverem IBM MQ nebo mezi dvěma servery IBM MQ.

Kanály jsou objekty, které poskytují komunikační cestu z jednoho správce front do jiného. Kanály se používají k přesouvání zpráv od jednoho správce front do jiného a zašití aplikace ze základních komunikačních protokolů. Správci front mohou existovat na stejném systému nebo na různých systémech na stejné platformě nebo na různých platformách. Odeslaná zpráva může pocházet z mnoha míst:

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

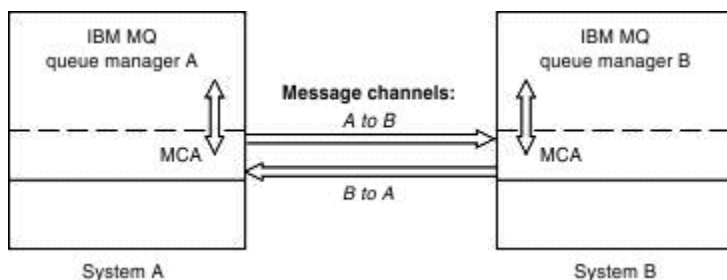
Kanál má dvě definice: jednu na každém konci připojení. Aby mohli správci front vzájemně komunikovat, je třeba definovat jeden objekt kanálu ve správci front, který má odesílat zprávy, a další komplementární zprávy ve správci front, který je má přijmout. Musí být použit stejný *název kanálu* na každém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

V produktu IBM MQ jsou k dispozici tři kategorie kanálu s různými typy kanálů v rámci těchto kategorií:

- Kanály zpráv, které jsou jednosměrné, a přenášejí zprávy z jednoho správce front do jiného.
- Kanály MQI, které jsou obousměrné, přenášejí volání MQI z produktu IBM MQ MQI client na správce front a odpovědi ze správce front na klienta IBM MQ.
- Kanály AMQP, které jsou obousměrné a připojují klienta AMQP ke správci front na serverovém počítači. IBM MQ používá kanály AMQP pro přenos 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 zpráv klienta.



Obrázek 2. Kanály zpráv mezi dvěma správci front

Kanál zpráv je jednosměrný odkaz. Pokud chcete, aby vzdálený správce front odpovídal na zprávy odeslané lokálním správcem front, je třeba nastavit druhý kanál k odeslání odpovědi zpět do lokálního správce front.

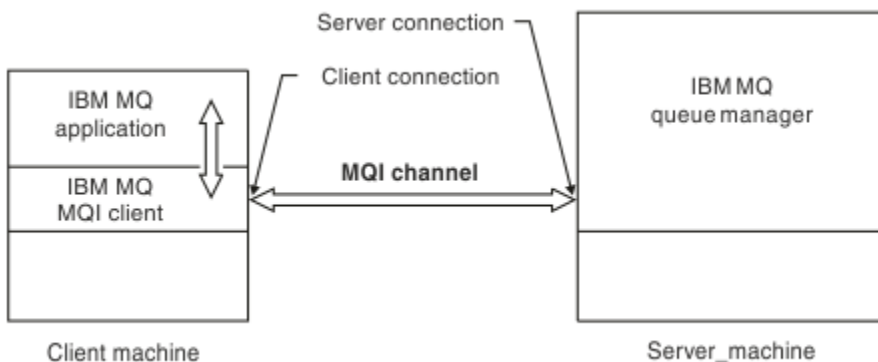
Kanál zpráv propojuje dva správce front s použitím *agentů kanálů zpráv* (MCA). Na každém konci kanálu se nachází agent kanálu zpráv. Chcete-li povolit přenos zpráv pomocí více podprocesů, můžete programu MCA povolit. Tento proces je znám jako *pipelining*. Potrubí umožňuje agentovi MCA efektivněji přenášet zprávy a zvyšovat výkon kanálů. Další informace o funkci pipelining naleznete v tématu [Atributy kanálů](#).

Další informace o kanálech naleznete v části [Channel-exit calls and data structures](#) “[Distribuované komponenty front](#)” na stránce 43.

Kanály MQI

Kanál MQI (Message Queue Interface) připojí produkt IBM MQ MQI client ke správci front v počítači se serverem a je ustanovený, když vydáte volání MQCONN nebo MQCONNX z aplikace IBM MQ MQI client .

Jedná se o dvoucestný odkaz a používá se pouze pro přenos volání a odpovědí MQI, včetně volání MQPUT obsahujících data zpráv a volání MQGET , které mají za následek vrácení dat zprávy. Existují různé způsoby vytvoření a použití definic kanálů (viz [Definování kanálů MQI](#)).



Obrázek 3. Klient-připojení a připojení k serveru pro kanál MQI

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

Pro definice kanálů MQI jsou k dispozici dva typy kanálů. Definují obousměrný kanál MQI.

Kanál připojení klienta

Tento typ je určen 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 produktu IBM MQ MQI client .

Kanály AMQP

Multi

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.

Definice kanálů

Popisy jednotlivých typů kanálů viz [“Definice kanálů”](#) na stránce 27 .

Související pojmy

[“Distribuované fronty a klastry”](#) na stránce 39

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

[Přehled rozhraní fronty zpráv](#)

Související úlohy

[Správa vzdálených objektů produktu IBM MQ](#)

[Zastavení kanálů MQI](#)

Související odkazy

[Volání uživatelských procedur kanálů a datové struktury](#)

[“Komunikace”](#) na stránce 30

IBM MQ MQI clients používají kanály MQI ke komunikaci se serverem.

Definice kanálů

Tabulky popisující různé typy kanálů zpráv a kanály 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 něž 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.

Typ definice kanálu zpráv	Popis
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.
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 CLUSSDR 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.

Typ definice kanálu zpráv	Popis
Příjemce klastru	Definice přijímacího kanálu klastru (CLUSRCVR) definuje přijímací 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:


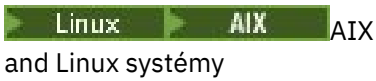


Typ kanálu MQI	Popis
Připojení serveru	Kanál připojení serveru je obousměrný kanál MQI, který je používán pro připojení klienta IBM MQ k serveru IBM MQ. Kanál připojení serveru odpovídá konci kanálu v serveru.
Připojení klienta	Kanál připojení klienta je obousměrný kanál MQI, který je používán pro připojení 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.

Komunikace

IBM MQ MQI clients používají kanály MQI ke komunikaci se serverem.

Definice kanálu musí být vytvořena jak na serveru IBM MQ MQI client, tak na serveru připojení. Postup při vytváření definic kanálů je vysvětlen v tématu [Definování kanálů MQI](#).

Možné protokoly přenosu jsou zobrazeny v následující tabulce:

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

Poznámka:

1.  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 uvádí možné kombinace IBM MQ MQI client a platform serverů pomocí těchto přenosových protokolů.

Aplikace IBM MQ na serveru IBM MQ MQI client může používat všechna volání MQI stejným způsobem, jako když je správce front lokální. Produkt **MQCONN** nebo **MQCONNX** přidruží aplikaci IBM MQ k vybranému správci front vytvořením *manipulátoru připojení*. Další volání pomocí tohoto manipulátoru připojení pak budou zpracovány připojeným správcem front. Komunikace IBM MQ MQI client vyžaduje aktivní spojení mezi klientem a serverem, a to na rozdíl od komunikace mezi správci front, který je nezávislý na připojení a nezávislý na čase.

Přenosový protokol je zadán s použitím definice kanálu a neovlivňuje aplikaci. Například aplikace Windows se může připojit k jednomu správci front prostřednictvím protokolu TCP/IP a do jiného správce front prostřednictvím protokolu NetBIOS.

Aspekty výkonu

Přenosový protokol, který používáte, může mít vliv na výkon klienta a serveru IBM MQ . Pro telefonickou podporu po pomalé telefonní lince může být doporučeno použít kompresi kanálu IBM MQ .

Kanály připojení klienta

Kanály připojení klienta jsou objekty, které poskytují komunikační cestu z produktu IBM MQ MQI client ke správci front.

Kanály připojení klienta se používají v distribuovaných frontách k přesouvání zpráv mezi správcem front a klientem. Odstíňují aplikace ze základních komunikačních protokolů. Klient může existovat na stejné nebo jiné platformě než správce front.

Související úlohy

[Konfigurace připojení mezi serverem a klientem](#)

Paměťové třídy

Paměťová třída mapuje jednu nebo více front na sadu stránek.

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

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

Další informace o třídách úložiště naleznete v příručce [Plánování v z/OS](#).


Moduly listener

Listenery jsou procesy, které přijímají požadavky na síť od jiných správců front nebo klientské aplikace a spouštějí přidružené kanály.

Procesy modulu listener lze spustit pomocí řídicího příkazu **runmq1sr** .

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

- Konfigurujete proces modulu listener.
- Určete, zda má být proces modulu listener automaticky spuštěn a zastaven při spuštění a zastavení správce front.

Důležité:  Objekty modulu listener nejsou v produktu 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 na systému z/OS” na stránce 154](#).

Související odkazy

[runmq1sr](#) (spuštění modulu listener)

Služby

Objekty *Služba* 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 určený jako SERVER. Objekt služby serveru je definice programu, který bude proveden, když je spuštěn uvedený správce front. Souběžně může být provedena pouze jedna instance procesu serveru. Během zpracování může být stav procesu serveru monitorován pomocí příkazu MQSC, DISPLAY SVSTATUS. Objekty služby serveru jsou obvykle definice programů, jako jsou obslužné rutiny zablokovaných dopisů nebo monitory spouštěčů, avšak programy, které lze spustit, nejsou omezeny na ty, které jsou dodávány s produktem IBM MQ. Navíc lze definovat

objekt služby serveru tak, aby zahrnoval příkaz, který bude spuštěn, když je ukončen běh uvedeného správce front, aby mohl ukončit program.

Příkazy

Příkaz je objekt služby, který má parametr SERVTYPE zadaný jako COMMAND. Objekt služby příkazu je definice programu, který bude proveden, když je spuštěn nebo zastaven zadaný správcem front. Souběžně může být provedeno více instancí procesu. Objekty služby příkazu se liší od objektů služby serveru, v tom okamžiku, kdy je proveden program, správce front nebude program monitorovat. Příkazové servisní objekty jsou obvykle definicemi programů, které jsou krátké, a budou provádět specifickou úlohu, jako je například spuštění jedné nebo více jiných úloh.

Důležité:  Objekty služby nejsou v produktu IBM MQ for z/OS podporovány.

Související pojmy

[Práce se službami](#)

Objekty tématu

Objekt tématu je objekt produktu IBM MQ, který umožňuje přiřazení specifických, jiných než výchozích atributů k tématům.

Téma je definováno aplikací publikováním nebo přihlášeným k odběru konkrétního *řetězce tématu*. Řetězec tématu může určovat hierarchii témat tak, že je oddělíte dopředným lomítkem (/). Tento obrázek lze vizualizovat *stromem témat*. Je-li například aplikace publikována na řetězce témat /Sport/American Football a /Sport/Soccer, vytvoří se strom témat s nadřazeným uzlem Sport se dvěma podřízenými 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, pak všechna témata dědí své atributy ze základního objektu tématu, SYSTEM.BASE.TOPIC.

Objekt tématu můžete vytvořit v libovolném uzlu stromu témat tak, že určíte řetězec témat daného uzlu v atributu TOPICSTR daného objektu tématu. Pro uzel administrativního tématu můžete také definovat další atributy. Další informace o těchto attributech najdete v tématu [Příkazy MQSC](#) nebo [Automatizace administrace pomocí příkazů PCF](#). Každý objekt tématu standardně dědí své atributy od svého nejbližšího nadřazeného uzlu administrativního tématu.

Objekty tématu lze také použít ke skrytí celého stromu témat od vývojářů aplikací. Je-li vytvořen objekt tématu s názvem FOOTBALL.US pro téma /Sport/American Football, může aplikace publikovat nebo odebírat objekt s názvem FOOTBALL.US namísto řetězce /Sport/American Football se stejným výsledkem.

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

Další informace o objektech tématu viz [“Publikování/odběr zpráv”](#) na stránce 58.

Pojmenování objektů IBM MQ


Konvence pojmenování přijatá pro objekty IBM MQ závisí na objektu. Název počítačů a ID uživatelů, které používáte s IBM MQ, jsou také předmětem některých omezení pojmenování.

Každá instance správce front je známá svým názvem. Tento název musí být jedinečný v rámci sítě vzájemně propojených správců front, aby mohl jeden správce front jednoznačně identifikovat cílového správce front, kterému byla odeslána veškerá daná zpráva.

Pro ostatní typy objektů má každý objekt k sobě přidružený název a může být na něj odkazováno tímto názvem. Tyto názvy musí být jedinečné v rámci jednoho správce front a typu objektu. Například, můžete mít frontu a proces se stejným názvem, ale nemůžete mít dvě fronty se stejným názvem.

V produktu IBM MQ názvy mohou mít maximálně 48 znaků, s výjimkou *kanálů*, jejichž maximální délka je 20 znaků. Další informace o pojmenování objektů IBM MQ viz [“Pravidla pro pojmenování objektů IBM MQ”](#) na stránce 33.

Název počítačů a ID uživatelů, které používáte spolu s IBM MQ , jsou také předmětem některých omezení pojmenování:

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

Související pojmy

[“IBM MQ NÁZVY SOUBORŮ”](#) na stránce 36

Každý správce front IBM MQ , fronta, definice procesu, seznam názvů, kanál, modul listener připojení klienta, služba listener, služba a objekt ověřovacích informací je reprezentován souborem. Vzhledem k tomu, že názvy objektů nejsou nutně platné názvy souborů, převede správce front v případě potřeby název objektu do platného názvu souboru.

Související odkazy

[“Pravidla pro pojmenování objektů IBM MQ”](#) na stránce 33

Názvy objektů IBM MQ mají maximální délky 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 délky 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 objektu IBM MQ a objekty z každého typu mohou mít stejný název, protože existují v samostatných prostorech jmen objektů: Například lokální fronta a odesílací kanál mohou mít oba stejný název. Nicméně objekt nemůže mít stejný název jako jiný objekt ve stejném oboru názvů: např. lokální fronta nemůže mít stejný název jako modelová fronta a kanál odesílatele nemůže mít stejný název jako kanál příjemce.

V samostatných oborech názvů objektů existují následující objekty produktu IBM MQ :

- Ověřovací informace
- Kanál
- Kanál klienta
- Modul listener
- Seznam názvů
- Proces
- Fronta
- Služba
- Paměťová třída
- Předplatné
- Téma

Délka znaku jména objektů

Obecně lze říci, že názvy objektů IBM MQ mohou být až 48 znaků dlouhé. Toto pravidlo se vztahuje k následujícím objektům:

- Ověřovací informace
- Klastr

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

Existují omezení:

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

Znaky v názvech objektů

Platné znaky pro názvy objektů produktu IBM MQ jsou:

Znaky	Omezení
Velká písmena A-Z	<ul style="list-style-type: none"> • Není
Malá písmena a-z	<ul style="list-style-type: none"> • V skriptech MQSC musí být názvy s malými písmeny uzavřeny v jednoduchých uvozovkách. Tím se zabrání přeložení malých písmen na velká písmena. • Systémy, které používají EBCDIC Katakana, nemohou v názvech objektů používat znaky malých písmen a-z. • z/OS V případě použití malých znaků v systémech z/OS mohou existovat omezení, například názvy správců front nemohou obsahovat malá písmena. • IBM i V systémech IBM i při použití CL příkazů musí být názvy s malými písmeny uzavřeny v jednoduchých uvozovkách. Tím se zabrání přeložení malých písmen na velká písmena.
Číslice 0-9	<ul style="list-style-type: none"> • Není
Tečka (.)	<ul style="list-style-type: none"> • Není

Znaky	Omezení
Podtržítka (_)	<ul style="list-style-type: none"> ▶ Multi Není ▶ z/OS Vyvarujte se použití názvů s úvodními nebo koncovými podtržítka, protože to nemohou být ošetřeny operacemi IBM MQ for z/OS a řídicími panely.
Lomítka (/)	<ul style="list-style-type: none"> ▶ Windows Na systémech Windows nesmí být první znak názvu správce front dopředným lomítkem. ▶ IBM i V systémech IBM i při použití CL příkazů musí být názvy obsahující dopředné lomítka uzavřeny v jednoduchých uvozovkách. ▶ z/OS Není
Procento (%)	<ul style="list-style-type: none"> ▶ ALW Není ▶ z/OS Pokud používáte produkt RACF jako externí správce zabezpečení pro produkt IBM MQ for z/OS, nepoužívejte v názvech objektů %, protože názvy nejsou při použití generických profilů produktu RACF zahrnuty do kontrol zabezpečení. ▶ IBM i V systémech IBM i při použití CL příkazů musí být názvy obsahující znak procent uzavřeny v jednoduchých uvozovkách.

Jsou zde 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 doplněn vpravo mezerami. Všechny krátké názvy, které jsou vráceny správcem front, jsou vždy vyplněny vpravo a prázdné.

Názvy front

Název fronty má dvě části:

- Název správce front
- Lokální název fronty, jak je znám pro daného správce front

Každá část názvu fronty je 48 znaků dlouhá.

Chcete-li se odkázat na lokální frontu, můžete vynechat název správce front (nahrazením mezerových znaků nebo použitím znaku null). Všechny názvy front navrácené do programu produktem IBM MQ však obsahují název správce front.


▶ **z/OS** Sdílená fronta, která je přístupná libovolnému správci front ve své skupině sdílení front, nemůže mít stejný název jako libovolná nesdílená lokální fronta ve stejné skupině sdílení front. Toto omezení zabraňuje tomu, aby aplikace omylem otevřela sdílenou frontu, když má otevřít lokální frontu, nebo naopak. Sdílené fronty a skupiny sdílení front jsou k dispozici pouze v produktu IBM MQ for z/OS.

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

Když aplikace používá název fronty, může být název buď název lokální fronty (nebo alias pro jeden), nebo název lokální definice vzdálené fronty, ale aplikace nemusí vědět, která z nich, pokud nepotřebuje získat zprávu z fronty (kdy musí být fronta lokální). Když aplikace otevře objekt fronty, volání MQOPEN provede funkci rozpoznání názvu a určí, která fronta má provést následující operace. Významnost této skutečnosti spočívá v tom, že aplikace nemá žádnou vestavěnou závislost na konkrétních frontách, které jsou definovány na konkrétních místech 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 řetězcem SYSTEM. , jsou vyhrazeny pro objekty definované správcem front. Příkazy **Alter**, **Define** a **Replace** můžete použít ke změně definic těchto objektů tak, aby vyhovovaly vaší instalaci. Názvy, které jsou definovány pro IBM MQ , jsou uvedeny v plném rozsahu v části [Názvy front](#).

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




Související pojmy

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

IBM MQ NÁZVY SOUBORŮ

Každý správce front IBM MQ , fronta, definice procesu, seznam názvů, kanál, modul listener připojení klienta, služba listener, služba a objekt ověřovacích informací je reprezentován souborem. Vzhledem k tomu, že názvy objektů nejsou nutně platné názvy souborů, převede správce front v případě potřeby název objektu do platného názvu souboru.

Výchozí cesta k adresáři správce front je následující:

- Předpona, která je definována v informacích o konfiguraci produktu IBM MQ :
 -   V systému AIX and Linux je výchozí předpona /var/mqm. Tato hodnota je konfigurována ve stanze DefaultPrefix konfiguračního souboru mqs.ini .
 -  Na 32bitových systémech Windows je výchozí předpona C:\Program Files (x86)\IBM\WebSphere MQ. V 64bitových systémech Windows je výchozí předpona C:\Program Files\IBM\MQ. Pro 32bitové i 64bitové instalace jsou datové adresáře nainstalovány do produktu C:\ProgramData\IBM\MQ. Tato hodnota je konfigurována ve stanze DefaultPrefix konfiguračního souboru mqs.ini .

Where available, the prefix can be changed using the IBM MQ properties page in the IBM MQ Explorer, otherwise edit the mqs.ini configuration file manually.

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

```
queue.manager
```

by byly reprezentovány jako:

```
queue!manager
```

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

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

Mohli byste například pojmenovat správce front:

QUEUE.MANAGER.ACCOUNTING.SERVICES

Každý správce front je však reprezentován souborem a existují omezení v maximální délce názvu souboru a na znacích, které lze v názvu použít. V důsledku toho se názvy souborů představujících objekty automaticky transformují za účelem splnění požadavků systému souborů.

Pravidla upravující transformaci názvu správce front jsou následující:

1. Transformovat jednotlivé znaky:
 - Od. na!
 - Od/do &
2. Pokud název stále není platný:
 - a. Oříznout ji na osm znaků
 - b. Připojit číselnou příponu o třech znacích

Předpokládejme například, že výchozí předpona a správce front s názvem queue.manager:

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

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

- **Windows** V systému Windows s FAT se název správce front stane takto:

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

- **Linux** **AIX** V systému AIX and Linux se název správce front stane následujícím:

```
/var/mqm/mqgrs/queue!manager
```

Transformační algoritmus také rozlišuje mezi názvy, které se liší pouze velikostí písmen v systémech souborů, které nejsou citlivé na velikost písmen.

Transformace názvu objektu

Názvy objektů nejsou nutně platné názvy systémů souborů. Je možné, že budete muset transformovat názvy objektů. Použitá metoda se liší od názvů pro názvy správců front, protože v každém počítači existuje pouze několik názvů správců front. Pro každého správce front může být k dispozici velký počet dalších objektů. Fronty, definice procesů, seznamy názvů, kanály, kanály připojení klienta, listenery, služby a objekty ověřovacích informací jsou reprezentovány v systému souborů.

Když je vygenerován nový název transformačním procesem, neexistuje žádný jednoduchý vztah s původním názvem objektu. Příkaz **dspmqls** lze použít k převodu mezi skutečnými a transformovanými názvy objektů.

Související odkazy

dspmqls (zobrazit jména souborů)

IBM i Názvy objektů v systému IBM i

Správce front má přidruženou knihovnu správce front, která má jedinečný název. Názvy správce front a názvy objektů mohou být nutně transformovat tak, aby splňovaly požadavky produktu IBM i Integrated File System (IFS).

Když je vytvořen správce front, IBM MQ přidruží knihovnu správce front k této knihovně. Této knihovně správce front je přidělen jedinečný název, který není delší než 10 znaků, a to převážně na základě jména

správce front definovaného uživatelem. Jak správce front, tak knihovna správce front jsou umístěny do adresáře, který je také založen na názvu správce front s předponou /QIBM/UserData/mqm. Níže je uveden 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 správce front jsou zapsány do oddílů v souboru /QIBM/UserData/mqm/mqs.ini.

IBM MQ Adresáře a soubory IFS

Systém IBM i Integrated File System (IFS) je rozsáhle používán produktem IBM MQ k ukládání dat. Další informace o IFS najdete v publikaci *Integrated File System Introduction*.

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

Cesta k adresáři správce front je vytvořena z následujících možností:

- Předpona, která je definována v konfiguračním souboru správce front qm.ini. Standardní předpona je /QIBM/UserData/mqm.
- Literál qmgrs.
- Kódovaný název správce front, který je názvem správce front, který je transformován do platného názvu adresáře. Například správce front queue/manager je představován produktem queue&manager.

Tento proces je označován jako transformace názvu.

Transformace názvu správce front IFS

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

Můžete například pojmenovat správce front QUEUE/MANAGER/ACCOUNTING/SERVICES. Stejným způsobem, jakým je vytvořena knihovna pro každého správce front, je každý správce front reprezentován také souborem. Vzhledem k variantním kódům kódu v EBCDIC existují omezení znaků, které lze v názvu použít. V důsledku toho se názvy souborů IFS, které reprezentují objekty, automaticky transformují, aby splňovaly požadavky systému souborů.

Při použití příkladu správce front s názvem queue/manager, transformace znaku / na hodnotu &a za předpokladu výchozí předpony bude název správce front v produktu IBM MQ for IBM i použit jako /QIBM/UserData/mqm/qmgrs/queue&manager.

Transformace názvu objektu

Názvy objektů nejsou nutně platné názvy systémů souborů, takže názvy objektů mohou být nutně transformovat. Použitá metoda se liší od názvů pro názvy správců front, protože pro každý počítač existuje pouze několik názvů správců front. Pro každého správce front může být k dispozici velký počet dalších objektů. V systému souborů jsou reprezentovány pouze definice procesu, fronty a seznamy názvů; kanály nejsou těmito pokyny ovlivněny.

Když je vygenerován nový název transformačním procesem, neexistuje žádný jednoduchý vztah s původním názvem objektu. K zobrazení transformovaných názvů pro objekty produktu IBM MQ můžete použít příkaz DSPMQOBJN.

Atributy objektu

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

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

V produktu IBM MQ existují dva způsoby odkazování na atribut:

- Použijte se název PCF, například **MaxMsgLength**.
- Použijte jeho název příkazu MQSC, například MAXMSGL.

z/OS Skupiny sdílení front

Správci front, kteří mohou přistupovat ke stejné sadě sdílených front, tvoří skupinu s názvem *skupina sdílení front* (QSG) a vzájemně spolu komunikují prostřednictvím prostředku CF (coupling facility), který ukládá sdílené fronty.

Sdílená fronta je typem lokální fronty se zprávami, ke kterým může přistupovat jeden nebo více správců front, kteří se nacházejí ve skupině sdílení front. To není stejné jako fronta, která je sdílena více než jednou aplikací, a to pomocí 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.

Skupiny sdílení front nejsou striktně objekty, ale jsou zde uvedeny pro vaše pohodlí.

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

Související pojmy

[“Sdílené fronty a skupiny sdílení front” na stránce 158](#)

Můžete použít sdílené fronty a skupiny sdílení front za účelem implementace vysoké dostupnosti prostředků produktu IBM MQ. Sdílené fronty a skupiny sdílení front jsou funkce jedinečné pro produkt IBM MQ for z/OS na platformě z/OS.

Systémové výchozí objekty

Systémové výchozí objekty jsou sady definic objektů, které jsou vytvářeny automaticky při každém vytvoření správce front.

Můžete zkopírovat a upravit libovolnou z těchto definic objektů pro použití v aplikacích při instalaci.

Výchozí názvy objektů mají cílový systém souborů; například výchozí lokální fronta je SYSTEM.DEFAULT.LOCAL.QUEUE a výchozí kanál příjemce je SYSTEM.DEF.RECEIVER. Tyto objekty nemůžete přejmenovat; jsou požadovány výchozí objekty těchto názvů.

Definujete-li objekt, všechny atributy, které nezadáte explicitně, jsou zkopírovány z odpovídajícího výchozího objektu. Pokud například definujete lokální frontu, tyto atributy, které nspecifikujete, jsou převzaty z výchozí fronty SYSTEM.DEFAULT.LOCAL.QUEUE.

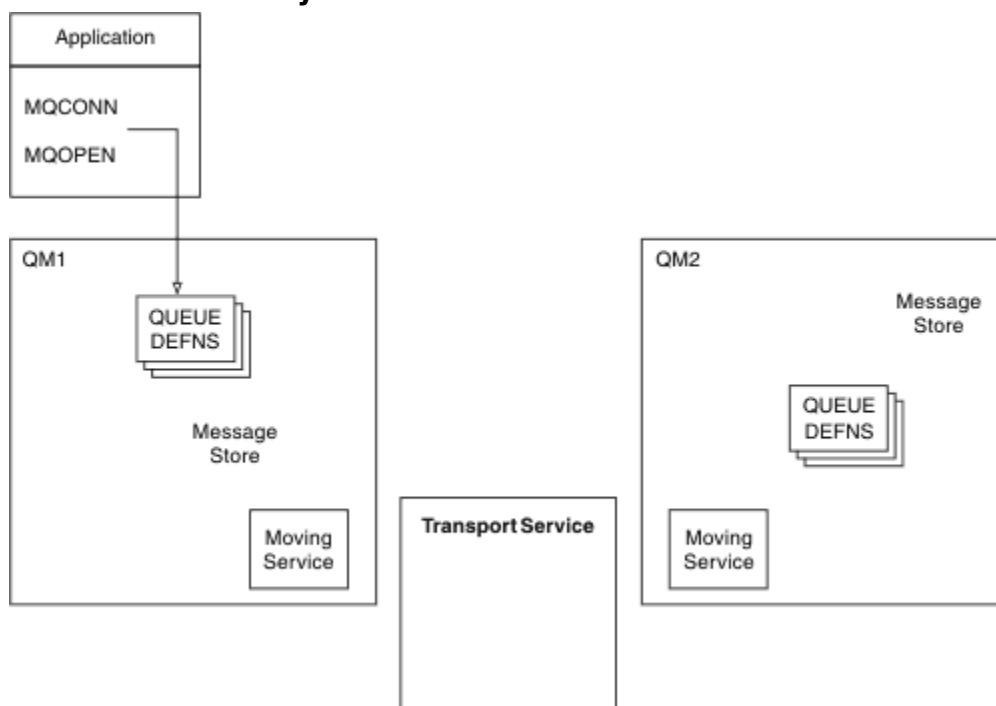
Související odkazy

[Systémové a výchozí objekty](#)

Distribuované fronty a klastry

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

distribuované fronty



Obrázek 4. Přehled komponent distribuovaných front

Na předchozím obrázku:

- Aplikace používá volání MQCONN pro připojení ke správci front. Aplikace pak použije volání MQOPEN k otevření fronty, aby mohla vkládat zprávy do fronty.
- Každý správce front má definici pro každou z jeho front. Může obsahovat definice *lokálních front* (tj. jejichž hostitelem je tento správce front), a definice *vzdálených front* (tj. jejichž hostitelem jsou jiní správci front).
- Jsou-li zprávy určeny pro vzdálenou frontu, lokální správce front je uchovává v *přenosové frontě*, která je trvale uchovává v úložišti zpráv, dokud je nelze předat vzdálenému správci front.
- Každý správce front obsahuje komunikační software známý jako *stěhovaná služba*, který správce front používá ke komunikaci s ostatními správci front.
- *Služba přenosu* je nezávislá na správci front a může být jakákoli z následujících možností (v závislosti na platformě):
 - Program Systems Network Architecture Advanced Program-to Program Communication (SNA APPC)
 - Transmission Control Protocol/Internet Protocol (TCP/IP)
 - Network Basic Input/Output System (NetBIOS) (základní systém vstupu/výstupu)
 - Sequenced Packet Exchange (SPX)

Komponenty potřebné k odeslání zprávy

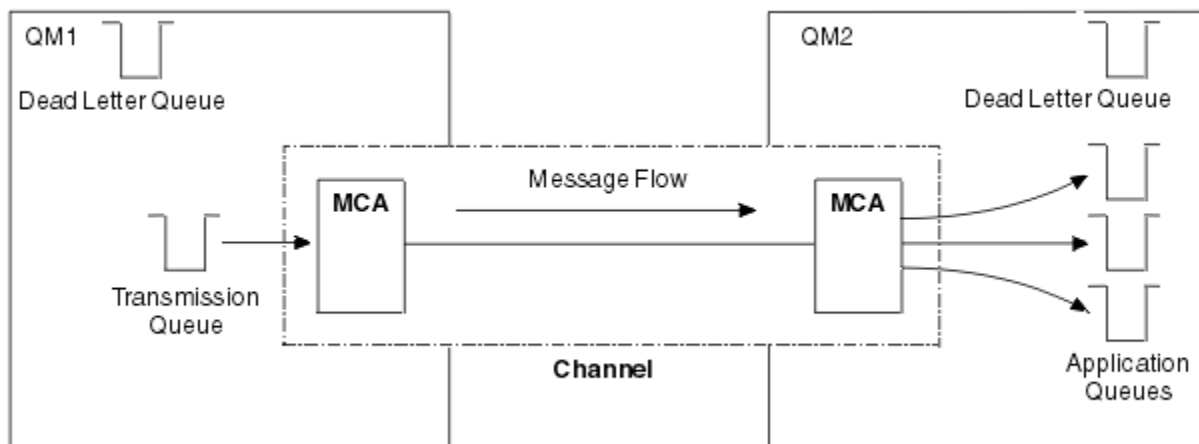
Pokud má být zpráva odeslána do vzdáleného správce front, potřebuje lokální správce front definice pro *přenosovou frontu* a *kanál*. Kanál je jednosměrný komunikační spoj mezi dvěma správci front. Může přenášet zprávy určené pro libovolný počet front ve vzdáleném správci front.

Každý konec kanálu má samostatnou definici, definující ji, například jako odesílající nebo přijímající konec. Jednoduchý kanál se skládá z definice kanálu *odesílatele* na lokálním správci front a z definice kanálu *příjemce* ve vzdáleném správci front. Tyto dvě definice musí mít stejný název a dohromady tvoří jeden kanál.

Software, který zpracovává odesílání a příjem zpráv, se nazývá *Agent kanálu zpráv* (MCA). Na každém konci kanálu je k dispozici *agent kanálu zpráv* (MCA).

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

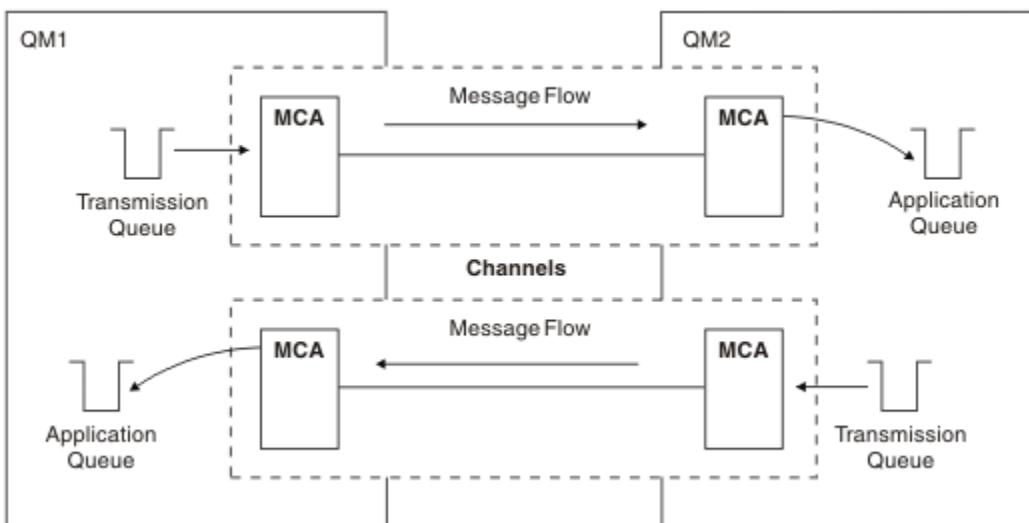
Následující obrázek ukazuje vztah mezi správcí front, přenosovou frontou, kanály a MCA:



Obrázek 5. Odesílání zpráv

Komponenty potřebné k vrácení zprávy

Pokud vaše aplikace vyžaduje, aby byly zprávy vráceny ze vzdáleného správce front, je třeba definovat jiný kanál tak, aby se spouštěl v opačném směru mezi správcí front, jak je znázorněno na následujícím obrázku:



Obrázek 6. Odesílání zpráv v obou směrech

Klastry

Místo ručního definování všech připojení v distribuovaném prostředí s frontami můžete seskupit sadu správců front v klastru. Pokud to provedete, mohou správci front vytvořit fronty, které jsou hostiteli dostupné jiným správcům front v klastru, bez nutnosti explicitních definic kanálů, definic vzdálených front nebo přenosových front pro každý cíl. 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 kanál příjemce klastru a jeden odesílací kanál klastru, přičemž všechny další kanály jsou automaticky spravovány klastrem.

Klient produktu IBM MQ se může připojit ke správcí front, který je součástí klastru, stejně jako se může připojit k libovolnému jinému správcí front. Stejně jako v případě ručně nakonfigurovaných

distribuovaných front zpráv použijete volání MQPUT k vložení zprávy 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 distribuce do distribuovaných front a také pomocí klastrů nebo nemusíte používat.

Výhody použití klastrů

Klastrování poskytuje dva klíčové výhody:

- Klastry zjednodušují administraci sítí IBM MQ, které obvykle vyžadují konfiguraci mnoha definic objektů pro kanály, vysílací fronty a vzdálené fronty. Tato situace je zvláště pravdivá ve velkých, potenciálně měnících se sítích, v nichž je třeba vzájemně propojit mnoho správců front. Tato architektura je obzvláště těžká pro konfiguraci a aktivní údržbu.
- Klastry lze použít k rozdělení pracovní zátěže provozu zpráv mezi frontami a správci front v klastru. Taková distribuce umožňuje rozdělení pracovní zátěže zpráv jedné fronty na ekvivalentní instance této fronty umístěné ve více správcích front. Toto rozdělení pracovní zátěže lze použít k dosažení větší odolnosti vůči selháním systému a ke zlepšení výkonu škálování u zvláště aktivních toků zpráv v systému. V takovém prostředí má každá z instancí distribuovaných front aplikace, které zpracovávají zprávy. Další informace naleznete v tématu [Použití klastrů pro správu pracovní zátěže](#).

Způsob směřování zpráv v klastru

Klaster je možné považovat za síť správců front udržovaných administrátorem svědomí. Kdykoli definujete frontu klastru, administrátor systému automaticky vytvoří odpovídající definice vzdálených front podle potřeby na ostatních správcích front.

Nemusíte vytvářet definice přenosových front, protože produkt IBM MQ poskytuje přenosovou frontu na každém správci front v klastru. Tato jediná přenosová fronta může být použita pro přenos zpráv do libovolného jiného správce front v klastru. Nejste omezeni na použití jedné přenosové fronty. Správce front může pomocí více přenosových front oddělit zprávy od jednotlivých správců front v klastru. Obvykle správce front používá jednu přenosovou frontu klastru. Atribut správce front DEFCLXQMůžete změnit tak, aby správce front používal jinou přenosovou frontu klastru pro každého správce front v klastru. Přenosové fronty klastru můžete také definovat ručně.

Všichni správci front, kteří se připojují ke klastru, souhlasí s tím, že pracují tímto způsobem. Vysílají informace o sobě a o frontách, které hostují, a přijímají informace o ostatních členech klastru.

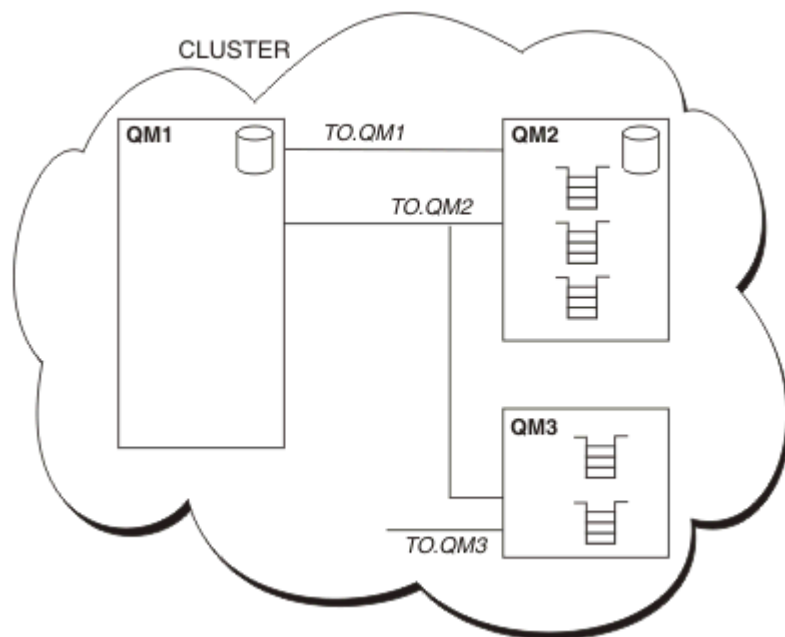
Chcete-li zajistit, aby nedošlo ke ztrátě informací při nedostupnosti správce front, určete dva správce front v klastru, aby se mohli chovat jako *úplná úložiště*. Tito správci front ukládají úplnou sadu informací o všech správcích front a frontách v klastru. Všichni ostatní správci front v klastru ukládají pouze informace o těchto správcích front a frontách, s nimiž si vyměňují zprávy. Tito správci front jsou známí jako *částečná úložiště*. Další informace viz ["Úložiště klastru"](#) na stránce 52.

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

- Kanál odesílatele klastru je komunikační kanál, jako je odesílací kanál. Chcete-li jej připojit k úplnému úložišti, které je již členem klastru, musíte ručně vytvořit jeden odesílací kanál klastru ve správci front.
- Přijímací kanál klastru je komunikační kanál, jako je přijímací kanál. Je třeba ručně vytvořit jeden kanál příjemce klastru. Kanál se chová jako mechanismus pro příjem klastrové komunikace správcem front.

Všechny ostatní kanály, které jsou 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 správce front

- CLUSTER obsahuje tři správce front, QM1, QM2, a QM3.
- Hostitelské systémy QM1 a QM2 budou používat úplná úložiště informací o správčích front a frontách v klastř.
- QM2 a QM3 jsou hostiteli některých front klastř, tj. front, které jsou přístupné pro jakéhokoli jiného správce front v klastř.
- Každý správce front má kanál příjemce klastř s názvem TO.qmgr , na kterém může přijímat zprávy.
- Každý správce front má také odesílací kanál klastř, na kterém může odesílat informace do jednoho ze správce front úložiště.
- QM1 a QM3 se odešlou do úložiště na QM2 a QM2 odešle do úložiště na QM1.

Distribuované komponenty front

Komponenty distribuovaných front jsou kanály zpráv, agenti kanálu zpráv, přenosové fronty, inicializátory kanálů a moduly listener a programy uživatelské procedury kanálu. Definice každého konce kanálu zpráv může být jedna z několika typů.

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

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

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

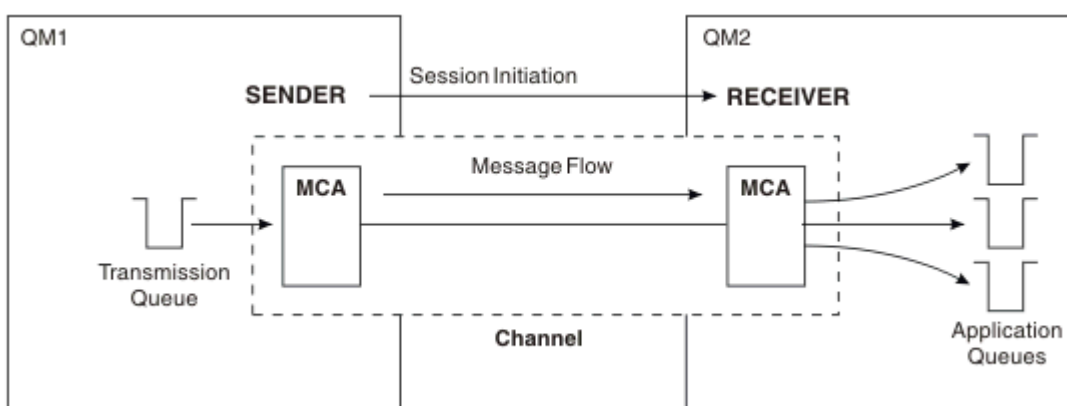
Kanál zpráv je definován pomocí jednoho z těchto typů definovaných na jednom konci a kompatibilního typu na druhém konci. Možné kombinace jsou:

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

Podrobné pokyny pro vytvoření přijímacího kanálu odesílatele jsou obsaženy v tématu [Definování kanálů](#). Příklady parametrů potřebných k nastavení kanálů příjemce odesílatele naleznete v tématu [Příklad informací o konfiguraci](#) platných pro vaši platformu. Informace o parametrech potřebných k definování kanálu libovolného typu naleznete v tématu [DEFINE CHANNEL](#).

Odesílací-přijímací kanály

Odesílatel v jednom systému spustí kanál tak, aby mohl odesílat zprávy do jiného systému. Odesílatel si vyžádá spuštění příjemce na druhém konci kanálu. Odesílatel odešle zprávy ze své přenosové fronty do přijímače. Přijemce vloží zprávy do cílové fronty. [Obrázek 8 na stránce 44](#) ilustruje toto.

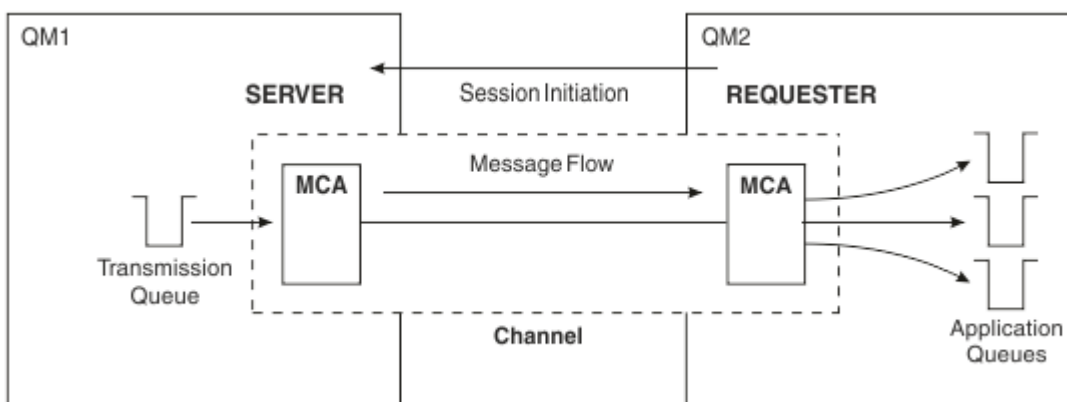


Obrázek 8. Přijímací kanál odesílatele

Kanály žadatele-serveru

Klient v jednom systému spouští kanál tak, aby mohl přijímat zprávy z druhého systému. Klient požádá server o spuštění na druhém konci kanálu. Server odešle zprávy žadateli z přenosové fronty definované ve své definici kanálu.

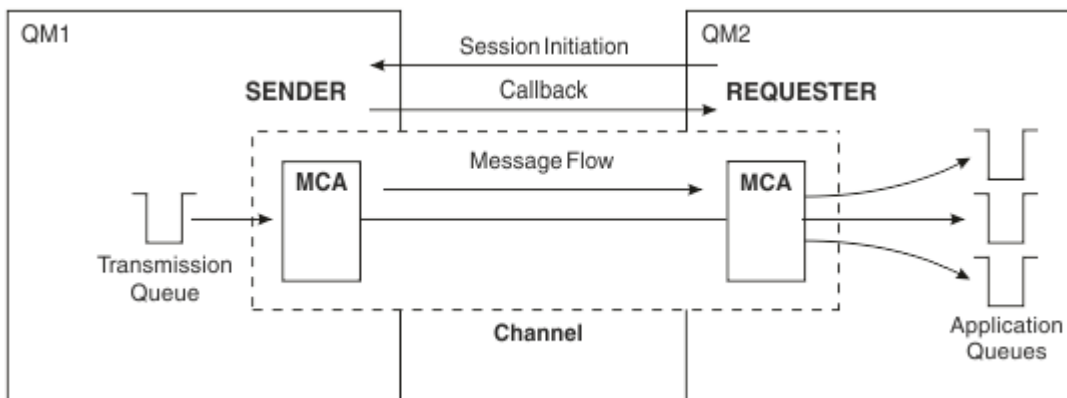
Kanál serveru může také iniciovat komunikaci a odesílat zprávy žadateli. To platí pouze pro *plně kvalifikované* servery, které jsou kanály serveru, které mají název připojení partnera uvedeného v definici kanálu. Plně kvalifikovaný server může být buď spuštěn žadatelem, nebo může iniciovat komunikaci s žadatelem.



Obrázek 9. Žadatelský kanál serveru

Kanály žadatele-odesílající kanály

Klient spustí kanál a odesílatel ukončí volání. Odesílatel pak restartuje komunikaci podle informací v příslušné definici kanálu (známé jako *callback*). Odesílá zprávy z přenosové fronty na žadatele.



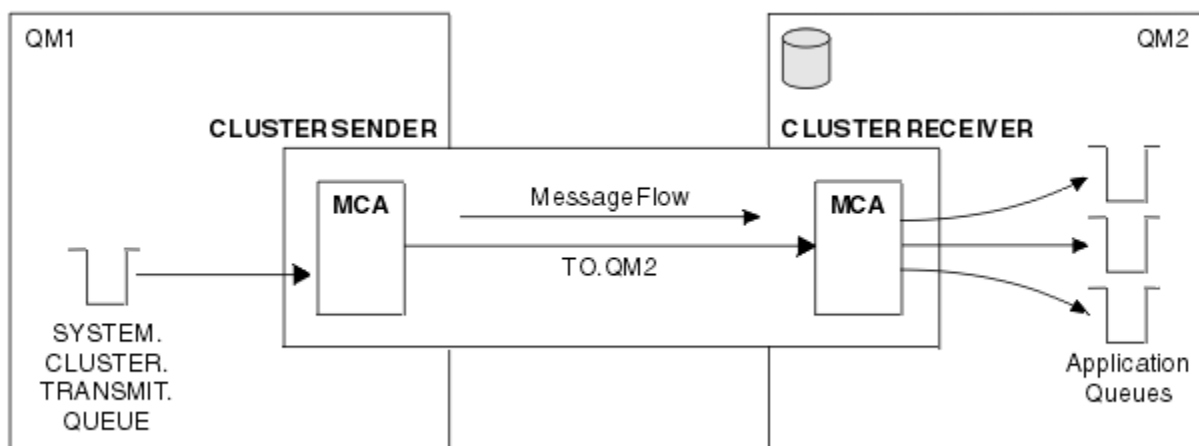
Obrázek 10. Žadatel-odesílací kanál

Kanály příjemce serveru

To je jako příjemce odesílatele, ale vztahuje se pouze na *plně kvalifikované* servery, které jsou kanály serveru, které mají název připojení partnera uvedeného v definici kanálu. Spuštění kanálu musí být iniciováno na konci serveru propojení. Ilustrace této části je jako ilustrace v části [Obrázek 8](#) na stránce 44.

Odesílací kanály klastru

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



Obrázek 11. Kanál odesílatele klastru

Přijímací kanály klastru

V klastru má každý správce front kanál příjemce klastru, na kterém může přijímat zprávy a informace o klastru. Ilustrace této části je jako ilustrace v části [Obrázek 11](#) na stránce 45.

Fronty nedoručených zpráv

Fronta nedoručených zpráv (nebo fronta nedoručených zpráv) je fronta, do níž jsou odesílány zprávy, pokud je nelze směřovat na jejich správné místo určení. Pro každého správce front je obvykle fronta nedoručených zpráv.

Fronta nedoručených zpráv (DLQ), někdy označovaná jako *fronta nedoručených zpráv*, je zadržující fronta pro zprávy, které nelze doručit do cílových front, například proto, že fronta neexistuje, nebo proto, že je plná. Fronty nedoručených zpráv se používají také 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 nelze doručit na jejich správné místo určení, lze uložit pro pozdější načtení.

Zprávy lze vkládat do fronty DLQ správci front, agenty kanálů zpráv (MCA) a aplikací. Všechny zprávy na DLQ musí mít předponu ve struktuře *dead-letter header*, MQDLH. Pole *Příčina* struktury MQDLH obsahuje kód příčiny, který identifikuje důvod, proč je zpráva na DLQ.

Obvykle byste měli definovat frontu nedoručených zpráv pro každého správce front. Pokud tomu tak není, a agent MCA nemůže vložit zprávu, je ponechán v přenosové frontě a kanál je zastaven. Také v případě rychlých přechodných zpráv (viz [Rychlé, přechodné zprávy](#)), nelze dodat, a v cílovém systému neexistuje žádná fronta nedoručených zpráv, tyto zprávy budou vyřazeny.

Použití smrtících front však může ovlivnit pořadí doručování zpráv, a proto se můžete rozhodnout, že nebudete tyto zprávy používat.

Definice vzdálených front

Definice vzdálených front jsou definice pro fronty, které jsou vlastněny jiným správcem front.

Zatímco aplikace mohou načítat zprávy pouze z lokálních front, mohou vkládat zprávy do lokálních front nebo vzdálených front. Proto může mít správce front, stejně jako definice pro každou z jeho lokálních front, *definice vzdálených front*. Výhoda definic vzdálených front je taková, že umožňují aplikaci vložit zprávu do vzdálené fronty, aniž by bylo nutné zadat název vzdálené fronty nebo vzdáleného správce front, nebo název přenosové fronty. Definice vzdálených front vám poskytují nezávislost umístění.

Existují jiná použití pro definice vzdálených front, které jsou popsány později.

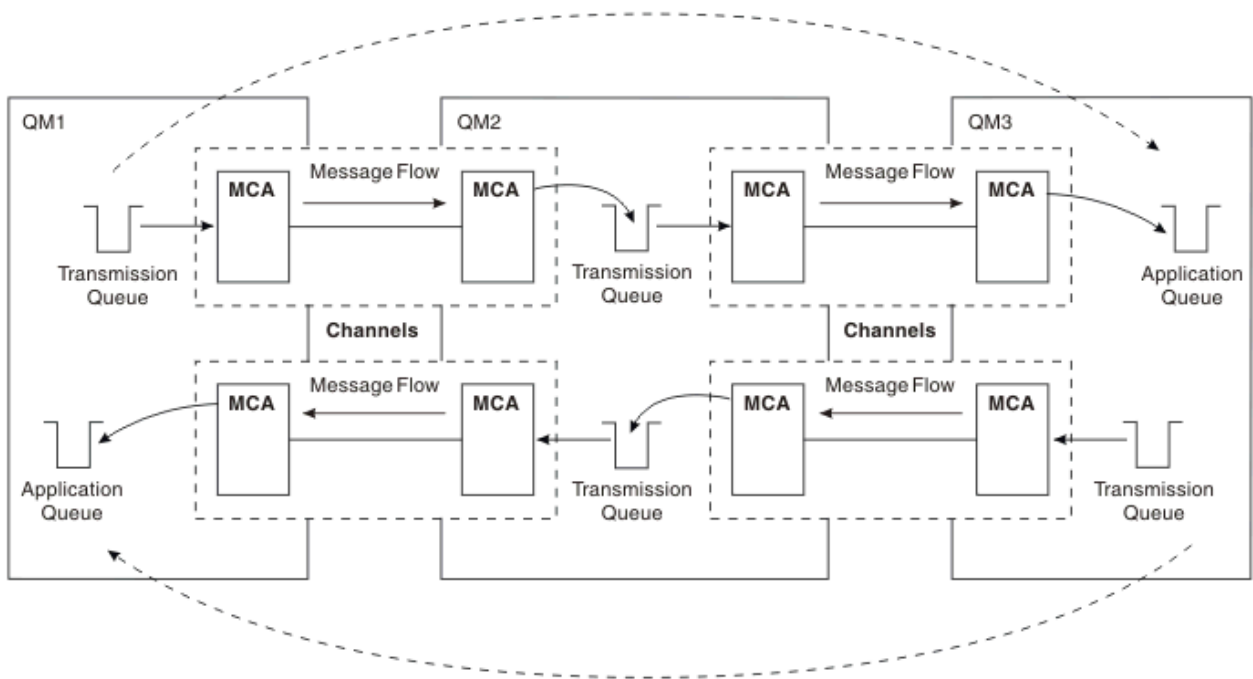
Jak se dostat ke vzdálenému správci front

Mezi jednotlivými zdrojovými a cílovými správci front nemusí být vždy k dispozici jeden kanál. Existuje řada dalších způsobů propojení mezi dvěma, včetně více-hopping, sdílení kanálů, použití různých kanálů a klastrování.

Vícenásobný přeskokování

Pokud mezi správcem zdrojové fronty a cílovým správcem front neexistuje žádné přímé komunikační spojení, je možné projít jedním nebo více *intermediačními správci front* na cestě k cílovému správcem front. To je známé jako *multi-hop*.

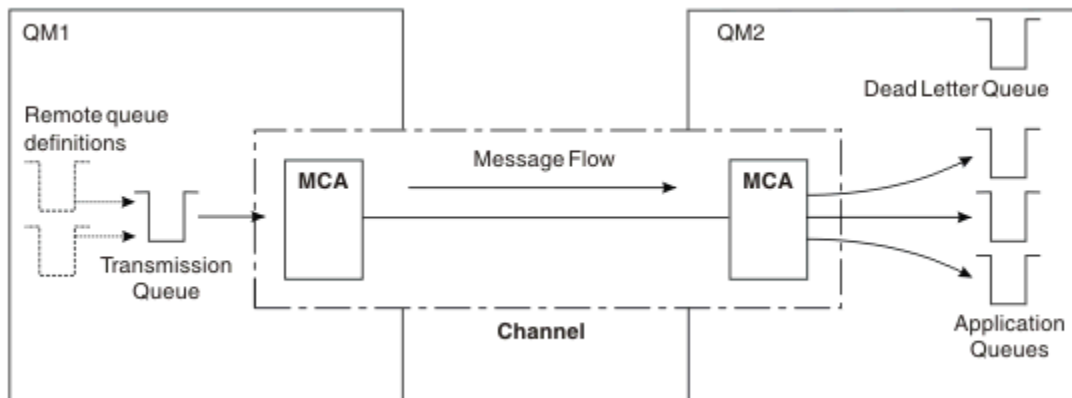
Je třeba definovat kanály mezi všemi správci front a přenosové fronty u zprostředkujících správců front. To je zobrazeno v části [Obrázek 12 na stránce 47](#).



Obrázek 12. Předání zprostředkujícími správci front

Sdílení kanálů

Jako návrhář aplikací máte možnost donutit aplikace k zadání názvu vzdáleného správce front spolu s názvem fronty nebo vytvoření *definice vzdálené fronty* pro každou vzdálenou frontu. Tato definice uchovává název vzdáleného správce front, název fronty a název přenosové fronty. Každopádně všechny zprávy ze všech aplikací adresující fronty ve stejném vzdáleném systému mají své zprávy odeslané prostřednictvím stejné přenosové fronty. To je zobrazeno v části Obrázek 13 na stránce 47.



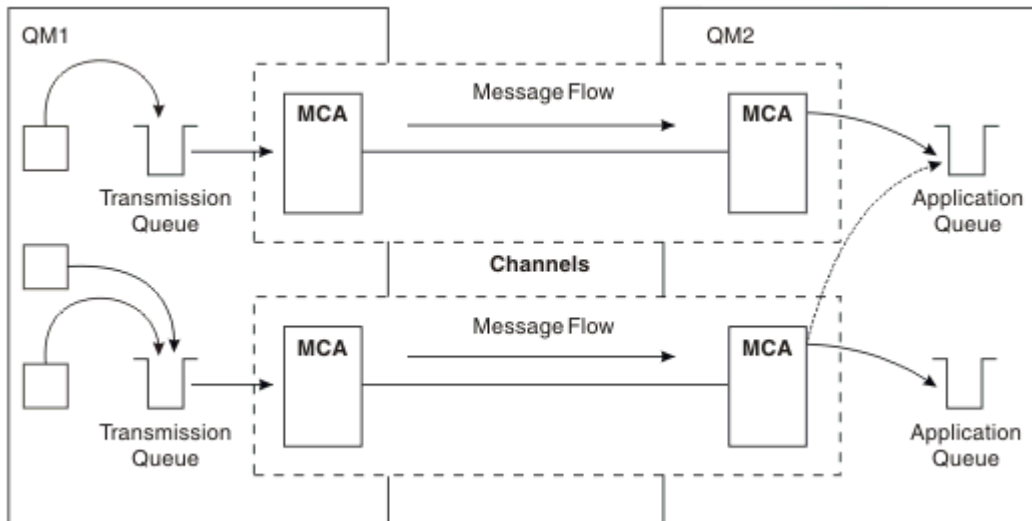
Obrázek 13. Sdílení přenosové fronty

Obrázek 13 na stránce 47 ilustruje, že zprávy z více aplikací do více vzdálených front mohou používat stejný kanál.

Použití různých kanálů

Pokud máte zprávy různých typů k odeslání mezi dvěma správci front, můžete mezi těmito dvěma kanály definovat více než jeden kanál. Existují okamžiky, kdy potřebujete alternativní kanály, možná pro účely zabezpečení, nebo pro výměnu rychlosti dodávek proti velké hromadné přepravě zpráv.

Chcete-li nastavit druhý kanál, je třeba definovat další kanál a další přenosovou frontu a vytvořit definici vzdálené fronty, která bude určovat umístění a název přenosové fronty. Aplikace pak mohou používat kterýkoli kanál, ale zprávy jsou stále doručovány do stejných cílových front. To je zobrazeno v části Obrázek 14 na stránce 48.



Obrázek 14. Použití více kanálů

Když použijete definice vzdálených front k určení přenosové fronty, vaše aplikace musí **neurčovat** umístění (tj. správce cílové fronty) sami. Pokud tak učiní, správce front nepoužívá definice vzdálených front. Definice vzdálených front vám poskytují nezávislost umístění. Aplikace mohou vkládat zprávy do *logické* fronty, aniž by věděli, kde je fronta umístěna, a můžete změnit *fyzickou* frontu, aniž byste museli měnit své aplikace.

Použití klastrování

Každý správce front v rámci klastru definuje přijímací kanál klastru. Pokud chce jiný správce front odeslat zprávu tomuto správci front, bude automaticky definovat odpovídající odesílací kanál klastru. Je-li například v klastru více než jedna instance fronty, může být kanál odesílatele klastru definován pro libovolného správce front, který je hostitelem dané fronty. IBM MQ používá algoritmus správy pracovní zátěže, který používá rutinu round-robin k výběru dostupného správce front, do kterého má být odeslána zpráva. Další informace viz [Klastry](#).

Informace o adresování

Když aplikace vkládá zprávy, které jsou určeny pro vzdáleného správce front, přidá je lokální správce front k nim, než je umístí do přenosové fronty. Toto záhlaví obsahuje název cílové fronty a správce front, tj. *adresovací informace*.

V prostředí s jedním správcem front je adresa cílové fronty vytvořena při otevření fronty pro vložení zpráv do fronty. Vzhledem k tomu, že cílová fronta se nachází ve stejném správci front, není třeba žádné informace o adresování.

V distribuovaném prostředí řazení do fronty musí správce front znát nejen název cílové fronty, ale také umístění této fronty (tj. název správce front) a směrování do tohoto vzdáleného umístění (tedy přenosová fronta). Tyto informace o adresování jsou obsaženy v záhlaví přenosu. Přijímací kanál odebere záhlaví přenosu a použije informace z něj k vyhledání cílové fronty.

Pokud používáte definici vzdálené fronty, můžete zabránit tomu, aby aplikace určujete název cílového správce front. Tato definice uvádí název vzdálené fronty, jméno vzdáleného správce front, do kterého jsou zprávy určeny, a jméno přenosové fronty použité pro přenos zpráv.

Co jsou aliasy?

Alias se používají k poskytování kvalitní služby pro zprávy. Alias správce front umožňuje administrátorovi systému změnit název cílového správce front bez toho, že byste museli měnit své aplikace. Umožňuje také administrátorovi systému změnit trasu na cílového správce front nebo nastavit přenosovou cestu, která zahrnuje předání několika dalších správců front (více přechodů do fronty). Alias fronty pro odpovědi poskytuje kvalitu služby pro odpovědi.

Alias správce front a aliasy fronty pro odpověď jsou vytvářeny s použitím definice vzdálené fronty s prázdným názvem RNAME. Tyto definice nedefinují skutečné fronty; používají správce front k interpretaci názvů fyzických front, názvů správců front a přenosových front.

Definice aliasů se vyznačují tím, že mají prázdný název RNAME.

Rozlišení názvu fronty

K rozlišování názvů fronty dochází při každém otevření fronty v každém správci front. Jeho účelem je identifikovat cílovou frontu, cílového správce front (který může být lokální) a přenosovou cestu k tomuto správci front (což může být null). Vyřešený název má tři části: název správce front, název fronty a, je-li správce front vzdálený, přenosová fronta.

Pokud existuje definice vzdálené fronty, nejsou odkazovány žádné definice alias. Název fronty zadaný aplikací je interpretován jako název cílové fronty, vzdáleného správce front a přenosové fronty určené v definici vzdálené fronty. Podrobnější informace o řešení názvů front naleznete v tématu [Řešení názvu fronty](#).

Pokud neexistuje žádná definice vzdálené fronty a je zadán název správce front nebo pokud je tento název rozpoznán službou názvů, bude správce front hledat v případě, že existuje definice aliasu správce front, která odpovídá zadanému názvu správce front. Pokud existují, jsou informace v ní použity k vyřešení názvu správce front na název cílového správce front. Definici aliasu správce front lze také použít k určení přenosové fronty k cílovému správci front.

Není-li vyřešený název fronty lokální frontou, je název správce front i název fronty zahrnutý v záhlaví přenosu každé zprávy vkládané aplikací do přenosové fronty.

Použitá přenosová fronta má obvykle stejný název jako vyřešený správce front, pokud není změněn definicí vzdálené fronty nebo definicí alias správce front. Pokud jste takovou přenosovou frontu nedefinovali, ale jste definovali výchozí přenosovou frontu, pak se použije.

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

Definice aliasů správce front

Definice aliasů správce front jsou použity, když aplikace, která otevírá frontu k 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 pozměňující nebo uvedení přenosové fronty
- Při příjmu zpráv určujícím, zda je lokální správce front pro tyto zprávy zamýšleným cílem

Odchozí zprávy-přemapování názvu správce front

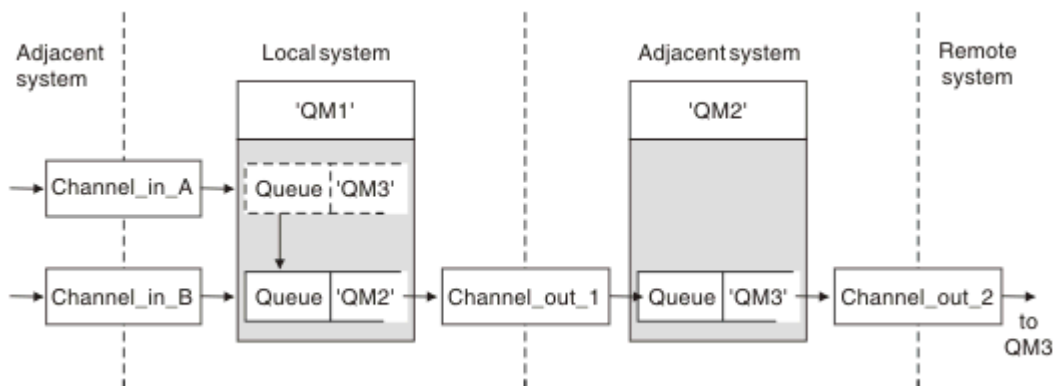
Definice aliasů správce front lze použít k přemapování názvu správce front určeného v rámci volání MQOPEN. Volání MQOPEN například určuje název fronty produktu THISQ a název správce front produktu YOURQM. V lokálním správci front existuje definice aliasu správce front jako v následujícím příkladu:

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

To ukazuje, že skutečný správce front, který má být použit, je-li aplikací umístěje zprávy do správce front YOURQM, je REALQM. Je-li lokální správce front REALQM, vloží zprávy do fronty THISQ, což je lokální fronta. Není-li lokální správce front nazván REALQM, směruje zprávu do přenosové fronty nazvané REALQM. Správce front změní záhlaví přenosu tak, aby bylo možné říci REALQM místo YOURQM.

Odchozí zprávy-změna nebo uvedení přenosové fronty

Produkt [Obrázek 15](#) na stránce 50 zobrazuje scénář, kdy zprávy dorazí do správce front QM1 se záhlavími přenosů zobrazujících názvy front ve správci front QM3. V tomto scénáři je produkt QM3 dostupný pro více přechodů přes QM2.



Obrázek 15. Alias správce front

Všechny zprávy pro QM3 jsou zachyceny v 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 jsou zprávy s různými destinacemi shromažďovány do společné přenosové fronty k odpovídajícímu sousednímu systému, pro další přenos do jejich míst určení.

Příchozí zprávy-určení cíle

Přijímající agent MCA otevře frontu, na kterou se odkazuje 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 hodnotou 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 aliasů fronty odpovědi

Definice alias fronty pro odpověď určuje alternativní názvy pro informace odpovědi v deskriptoru zpráv. Výhodou tohoto je, že můžete změnit název fronty nebo správce front, aniž byste museli měnit své aplikace.

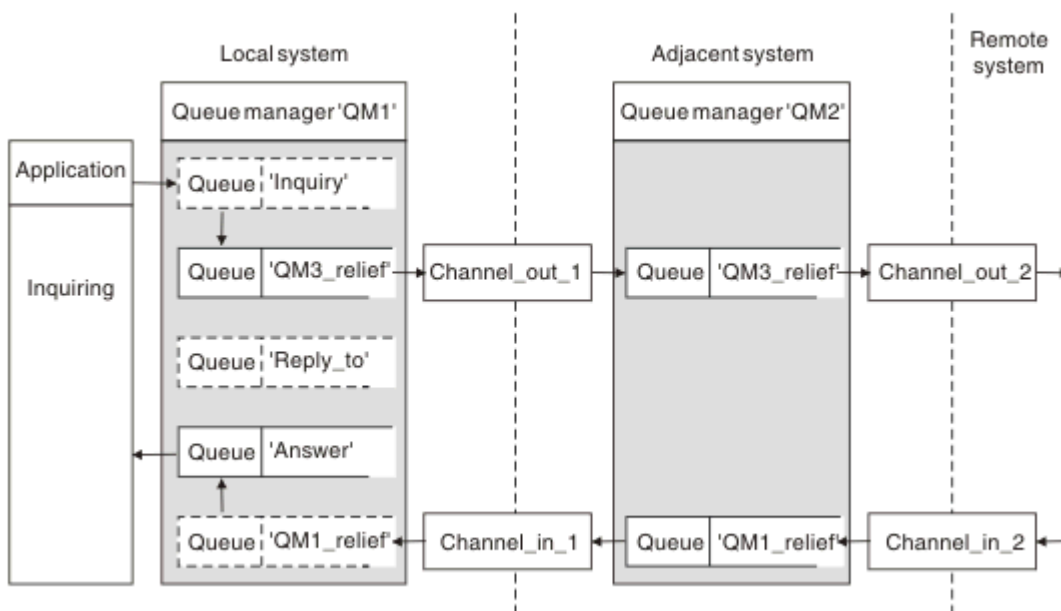
Rozlišení názvu fronty

Když aplikace odpoví na zprávu, použije data v *deskriptoru zprávy* přijaté zprávy, aby zjistila jméno fronty, na kterou se má odpovědět. Odesílající aplikace označuje, kam se odesílají odpovědi a připojují tyto informace ke svým zprávám. Tento koncept musí být koordinován jako součást návrhu vaší aplikace.

Rozpoznání názvu fronty probíhá na odesílajícím konci aplikace před tím, než je zpráva vložena do fronty. K vyřešení názvu fronty proto dochází před interakcí se vzdálenou aplikací, do které je zpráva odesílána. To je jediná situace, kdy rozlišení názvů probíhá v době, kdy se fronta neotevře.

Rozpoznání názvu fronty pomocí aliasu správce front

Obvykle aplikace určuje frontu pro odpověď a ponechá název správce front pro odpovědi prázdný. Správce front dokončí svůj vlastní název v čase vložení. Tato metoda funguje dobře, kromě případů, kdy má být použit alternativní kanál pro odpovědi, například kanál, který používá přenosovou frontu `QM1_relief` namísto výchozího kanálu zpětného přenosu, který používá přenosovou frontu `QM1`. V této situaci názvy správců front uvedené v záhlavích přenosových front neodpovídají názvům správce front "real", ale znovu jsou určeny pomocí definic aliasů správce front. Chcete-li vrátit odpovědi po alternativních trasách, je nutné namapovat také data fronty pro odpověď s použitím alias fronty pro odpověď na alias fronty.



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

V příkladu v souboru [Obrázek 16](#) na stránce 51:

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

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

`ReplyToQMgr` musí být prázdný, aby bylo možné použít alias fronty pro odpověď.

2. Vytvoříte definici alias 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 se odesílají s deskriptorem zprávy zobrazujícím ReplyToQ= ' Answer ' a ReplyToQMgr= 'QM1_relief'.
4. Specifikace aplikace musí zahrnovat informace, které mají být ve frontě nalezeny ve frontě Answer , nikoli v produktu Reply_to.

Chcete-li se připravit na odpovědi, musíte vytvořit paralelní návratový kanál, který definuje:

- Ve frontě QM2, přenosová fronta s názvem QM1_relief

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

- Při QM1alias správce front QM1_relief

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

Tento alias správce front ukončuje řetězec paralelních návratových kanálů a zachycuje zprávy pro QM1.

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

Název fronty pro odpověď

S pojmenováním odpovědí na fronty je třeba věnovat pozornost. Důvodem, proč aplikace vkládá do zprávy název fronty pro odpověď, je to, že může určit frontu, do které jsou odesílány její odpovědi. Při vytváření definice alias fronty pro odpověď s tímto názvem nemůžete mít skutečnou frontu pro odpověď (to znamená lokální definici fronty) se stejným názvem. Proto musí definice alias fronty pro odpověď obsahovat nový název fronty a také název správce front a specifikace aplikace musí obsahovat informace, které jsou její odpovědi nalezeny v této jiné frontě.

Aplikace nyní musí načítat zprávy z jiné fronty z toho, které pojmenovali jako odpověď na frontu, když vložili původní zprávu.

Komponenty klastru

Klustry se skládají z správců front, klastrovaných úložišť, kanálů klastru a front klastru.

Informace o jednotlivých komponentách klastru najdete v následujících dílčích tématech:

Související pojmy

[Porovnání klastrování a distribuovaných front](#)



Související úlohy

[Konfigurace klastru správce front](#)

[Nastavení nového klastru](#)

Úložiště klastru

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

Informace o úložišti obsahují názvy správců front, jejich umístění, kanály, které hostují, a další informace. Informace se ukládají ve formě zpráv ve frontě s názvem SYSTEM . CLUSTER . REPOSITORY . QUEUE. Tato fronta je jedním z výchozích objektů.  V produktu [Multiplatforms](#) je definován při vytváření správce front produktu IBM MQ.  V produktu IBM MQ for z/OS je tento parametr definován jako součást přizpůsobení správce front.

Úplné úložiště a částečné úložiště

Dva správci front v klastru obvykle obsahují úplné úložiště. Zbývající správci front mají všechny zadržené dílčí úložiště.

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

Částečné úložiště obsahuje informace o pouze těch správcích front, s nimiž správce front potřebuje vyměňovat zprávy. Správci front žádají o aktualizace informací, které potřebují, takže pokud se změní, správce front úplného úložiště jim odešle nové informace. Pro většinu času obsahuje dílčí úložiště všechny informace, které správce front potřebuje 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, migrujte úplná úložiště před částečnými úložišti. Důvodem je skutečnost, že starší úložiště nemůže ukládat novější atributy zavedené v novější verzi. Toleruje je, ale neuchovává 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 v rámci všech klastrů, jejichž je členem.

Správce front klastru může být hostitelem front, které inzeruje k ostatním správcům front v klastru. K tomu však není třeba. Místo toho může odesílat zprávy do front hostovaných na jiném místě v klastru a přijímat pouze odpovědi, které jsou přímo adresovány tomuto uživateli.

 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 nezávislí. Mají plnou 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. Mají kompletní sadu všech definic, pro použití v případě potřeby. Klaster je federace správců front.

Po vytvoření nebo změně definice ve správci front klastru se tyto informace odešlou do správce front úplného úložiště. Ostatní 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 dva nebo více správců front úplného úložiště v každém klastru. Správci front úplného úložiště obdrží informace odeslané ostatními správcí front v klastru a aktualizují svá úložiště. Posílají si navzájem zprávy, aby si byli jisti, že jsou oba udržovány v aktuálním stavu s novými informacemi o klastru.

Správci front a úložiště

Každý klaster má alespoň jeden (nejlépe dva) správce front, v nichž jsou umístěna úplná úložiště informací o správcích front, frontách a kanálech v klastru. Tato úložiště také obsahují požadavky od ostatních správců front v klastru pro aktualizace informací.

Ostatní správci front, kteří obsahují dílčí úložiště, obsahují informace o podmnožině front a správců front, s nimiž potřebují komunikovat. Správci front vytvářejí svá dílčí úložiště tím, že provádějí dotazy v případě, že nejprve potřebují přistupovat k jiné frontě nebo správci front. Požadují, aby byly upozorněny na všechny nové informace týkající se této fronty nebo správce front.

Každý správce front ukládá své informace o úložišti ve zprávách 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 spojuje klastr, definuje odesílatele klastru, CLUSSDR, kanál na jedno z úložišť. Okamžitě se dozví, které ostatní správci front v klastru mají úplná úložiště. Od té doby může správce front požadovat informace z některého z úložišť. Odešle-li správce front informace do vybraného úložiště, odešle také informace do jiného úložiště (je-li k dispozici).


Úplné úložiště je aktualizováno, když správce front, který je hostitelem, obdrží nové informace od jednoho z správců front, který je k němu připojen. Nové informace jsou také odeslány do jiného úložiště, aby se snížilo riziko zpoždění v případě, že správce front úložiště není ve službě. Jelikož jsou všechny informace odeslány dvakrát, úložiště musí zahodit duplikáty. Každá položka informací nese pořadové číslo, které úložiště používají k identifikaci duplikátů. Všechna úložiště jsou postupně mezi sebou ve výměně 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 z nich mohou být vhodnější pro váš systém.

Náklady iniciátoru kanálu

Ve frontách klastru jsou zprávy odesílány kanály, což umožňuje kromě nákladů aplikací také náklady iniciátoru kanálu. V síti jsou náklady, protože kanály mohou získávat a vkládat zprávy. Tyto náklady se neprezentují se sdílenými frontami, které proto při přesouvání zpráv mezi správci front ve skupině sdílení front používají méně výpočetního výkonu než fronty klastru.

Dostupnost zpráv

Při vkládání do fronty odesílají fronty klastru zprávu do jednoho ze správců front s aktivními kanály připojenými ke správci front. Pokud v vzdáleném správci front nejsou aplikace používané ke zpracování zpráv funkční, zprávy nebudou zpracovány a počkejte, až se aplikace spustí. Podobně platí, že pokud je správce front vypnut, nebudou všechny zprávy v daném správci front zpřístupněny, dokud se správce front nerestartuje. Tyto instance zobrazují nižší dostupnost zpráv než při použití sdílených front.

Při použití sdílených front může každá aplikace ve skupině sdílení front odesílat zprávy, které jsou odeslány. Pokud vypnete jednoho správce front ve skupině sdílení front, budou k dispozici zprávy ostatním správcům front a poskytují vyšší dostupnost zpráv než při použití front klastru.

kapacita

Prostředek CF je dražší než disk; proto jsou náklady na ukládání 1.000.000 zpráv v lokální frontě nižší než zařízení pro spojení s dostatečnou kapacitou pro uložení stejného počtu zpráv.

Odesílání jiným správcům front

Zprávy ve sdílené frontě jsou k dispozici pouze v rámci skupiny sdílení front. Chcete-li používat správce front mimo skupinu sdílení front, je třeba použít kanály. Klastrování můžete použít k vyrovnávání pracovní zátěže mezi více vzdálenými distribuovanými správci front.

Vyrovňování zátěže

Klastrování můžete použít k poskytnutí váhy odesílaným kanálům a správcům front, které získají podíl odeslaných zpráv. Můžete například odeslat 60% zpráv jednomu správci front a 40% zpráv do jiného správce front. Tato instance nezávisí na schopnosti vzdáleného správce front zpracovat práci. Systém s prvním správcem front může být přetížen a systém s druhým správcem front může být nečinný, ale většina zpráv stále přejde k prvnímu správci front.

Se sdílenými frontami mohou mít dva systémy CICS zprávy. Je-li jeden systém přetížen, druhý systém převezme většinu zátěže.

Kanály klastru

V každém úplném úložišti ručně definujete přijímací kanál klastru a sadu odesílacích kanálů klastru pro připojení ke všem ostatním úplným úložištím v klastru. Když přidáte částečné úložiště, ručně definujete kanál příjemce klastru a jeden odesílací kanál klastru, který se připojuje k jednomu z úplných úložišť. Další odesílací kanály klastru jsou podle potřeby automaticky definovány klastrem. Automaticky definované kanály odesílatele klastru získávají své atributy z příslušné 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 daném klastru.

Pro každého správce front klastru je třeba definovat alespoň jeden kanál CLUSRCVR . Definováním kanálu CLUSRCVR správce front zobrazí ostatní správce front klastru, kterého je k dispozici pro příjem zpráv.

Definice kanálu CLUSRCVR také umožňuje jiným správcům front automaticky definovat odpovídající definice odesílacího kanálu klastru. Viz část [“Automaticky definované kanály odesílatele klastru”](#) na stránce 56 v tomto článku.

Kanál odesílatele klastru: CLUSSDR

Kanál CLUSSDR můžete ručně definovat ze všech správců front úplného úložiště do všech ostatních správců front úplného úložiště v klastru. Všechny aktualizace vyměňované v úplných úložištích jsou přenášeny výhradně na tyto kanály. Pokud tyto kanály definujete ručně, budete přímo řídit síť úplných úložišť.

Přidáte-li do klastru správce front dílčího úložiště, ručně definujte jeden kanál CLUSSDR pro připojení k jednomu z úplných úložišť. Je příliš malý rozdíl v tom, jaké úplné úložiště si zvolíte, protože po provedení počátečního kontaktu jsou další objekty správce front klastru pro vašeho správce front, včetně kanálů CLUSSDR , definovány automaticky, jak je nezbytné. To umožní správci front odesílat informace o klastru do libovolného úplného úložiště a odesílat zprávy libovolnému správci front v klastru.

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

Kanály CLUSSDR byste měli definovat pouze ručně z výše uvedených důvodů. To znamená, že se na počátku připojí částečné úložiště k úplnému úložišti, nebo k připojení dvou úplných úložišť dohromady. Ruční konfigurace kanálu CLUSSDR , který se připojuje k částečnému úložišti nebo ke správci front, který není v klastru, způsobí, že budou vydány chybové zprávy jako AMQ9427 a AMQ9428 . Ačkoli to může být někdy nevyhnutelné jako dočasná situace, například když upravujete umístění úplného úložiště, měla by být definice ručně odstraněna co nejdříve.

Automaticky definované kanály odesílatele klastru

Obvykle když do klastru přidáte správce front s částečným úložištěm, definujete ve správci front pouze dva kanály klastru:

- Odesílatel klastru (CLUSSDR) kanálu do úplného správce front úložiště pro klastr.
- Přijímač klastru (CLUSRCVR) kanálu.

Kanál CLUSSDR , který definujete, umožňuje správci front navázat počáteční kontakt s klastrem. Po počátečním kontaktu jsou další kanály CLUSSDR definovány automaticky podle klastru, je-li to potřeba.

Automaticky definovaný kanál CLUSSDR má své atributy z příslušné definice kanálu CLUSRCVR v přijímajícím správci front. I v případě, že existuje ručně definovaný kanál CLUSSDR , použijí se atributy z automaticky definovaného kanálu CLUSSDR . Předpokládáme například, že definujete kanál CLUSRCVR , aniž byste zadali číslo portu v parametru **CONNNAME** , a ručně definujete kanál CLUSSDR , který určuje číslo portu. Když automaticky definovaný kanál CLUSSDR nahradí ručně definovaný kanál, bude číslo portu (převzaté ze kanálu CLUSRCVR) prázdné. Použijte se výchozí číslo portu a kanál selže.

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

Abyste se vyhnuli nejasnostem, pokud možno dodržte následující pokyny:

- Kanály CLUSSDR lze pouze ručně definovat tak, aby ukazovaly na úplná úložiště.
- Pokud máte ručně definované kanály CLUSSDR , nakonfigurujte je tak, aby identické odpovídaly odpovídající definici kanálu CLUSRCVR v přijímajícím správci front.

Viz též [Working with auto-defined channels](#).

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 odběru na ostatních správci 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 správci 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.

Informace o konfiguraci témat klastru pro použití *přímého směřování* nebo *směřování hostitele* *témata* informace o dědičnosti klastrovaných témat a předplatných zástupných znaků naleznete v tématu 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

ZOBRAZIT TÉMA

ZOBRAZIT STAV TPSTATUS

ZOBRAZIT POD

Výchozí objekty klastru

 Na platformách Multiplatforms jsou výchozí objekty klastru zahrnuty do sady výchozích objektů automaticky vytvořených při definování správce front.  V produktu z/OS lze výchozí definice objektů klastru nalézt ve vzorcích přizpůsobení.

Poznámka: Výchozí definice kanálů můžete změnit stejným způsobem jako v libovolné jiné definici kanálu spuštěním příkazů MQSC nebo PCF. Neměňte výchozí definice fronty, kromě 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 všechny nové nebo změněné informace o správci front nebo o požadavcích na informace o ostatních správcích front. SYSTEM . CLUSTER . COMMAND . QUEUE je normálně prázdný.

SYSTEM . CLUSTER . HISTORY . QUEUE

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

Ve výchozím nastavení objektu je parametr SYSTEM . CLUSTER . HISTORY . QUEUE nastaven na hodnotu PUT (ENABLED). Chcete-li potlačit shromažďování historie, změňte nastavení na hodnotu 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 ukládání všech úplných informací o 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 do všech front a správců front, kteří jsou v klastrech. Výchozí přenosovou frontu pro každý odesílací kanál klastru můžete změnit na hodnotu SYSTEM . CLUSTER . TRANSMIT . *ChannelName* změnou atributu správce front DEFCLXQ . Produkt SYSTEM . CLUSTER . TRANSMIT . QUEUE lze odstranit. Používá se také k definování kontroly autorizace, zda je použita výchozí přenosová fronta, která se používá, SYSTEM . CLUSTER . TRANSMIT . QUEUE nebo SYSTEM . CLUSTER . TRANSMIT . *ChannelName*.

SYSTEM . DEF . CLUSRCVR

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

Související pojmy

Práce s výchozími objekty klastru

Publikování/odběr zpráv

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

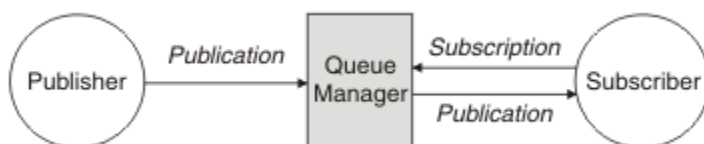
Před tím, než může aplikace IBM MQ mezi dvěma body poslat zprávu do jiné aplikace, musí o této aplikaci vědět něco. Například potřebuje znát název fronty, do které se mají odeslat informace, a může také uvést název správce front.

IBM MQ publish/subscribe odstraní potřebu vaší aplikace vědět cokoliv o cílové aplikaci. To vše, co má odesílající aplikace dělat, je toto:

- Založit zprávu IBM MQ obsahující informace, které aplikace chce.
- Přiřadíte zprávu k tématu, které označuje předmět informací.
- Nechte IBM MQ zpracovat distribuci těchto informací.

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

Následující obrázek ukazuje nejjednodušší systém publikování/odběr. K dispozici je jeden vydavatel, jeden správce front a jeden odběratel. Odběr je vytvořen odběratelem ve správci front, publikování je odesláno z vydavatele do správce front a publikování je poté předáno odběrateli do odběratele.



Obrázek 17. Konfigurace jednoduchého 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 jak vydavatel, tak i odběratel.

Další významný rozdíl mezi systémem zpráv publikování/odběru a dvoubodovým systémem je taková, že zpráva odeslaná do fronty zpráv typu point-to-point je zpracována pouze jednou aplikací s možností spotřeby. Každý zainteresovaný odběratel zpracovává zprávu publikovanou v tématu publikování/odběru, kde je registrován zájem více než jeden odběratel.

Publikování/odběr komponent

Publikování/odběr je mechanismus, podle kterého mohou odběratelé přijímat informace ve formě zpráv od vydavatelů. Interakce mezi vydavatelem a odběrateli jsou řízeny správci front pomocí standardních mechanismů produktu 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 jak vydavatel, tak i odběratel.

Poskytovatel informací se nazývá *vydavatel*. Vydavatelé poskytují informace o předmětu, aniž by potřebovali vědět cokoliv o aplikacích, které mají o tyto informace zájem. Vydavatelé generují tyto informace ve formě zpráv s názvem *publications* , které chtějí publikovat a definovat téma těchto zpráv.

Příjemce informací se nazývají *odběratel*. Odběratelé vytvářejí *odběry* popisující téma, o které se odběratel zajímá. Tento odběr určuje, které publikace jsou postoupeny odběrateli. Odběratelé mohou vytvářet více odběrů a mohou přijímat informace od mnoha vydavatelů.

Publikované informace jsou odeslány ve zprávě IBM MQ a předmět informací je identifikován svým *tématem*. Vydavatel určuje téma při publikování informací a odběratel určuje témata, o kterých chce přijímat publikování. Odběratel je odeslán informace o pouze těch tématech, k nimž se přihlašuje.

Jedná se o existenci témat, která umožňují oddělit poskytovatele a spotřebitele informací v systému zpráv publikování/odběru tím, že odebere potřebu zahrnutí určitého místa určení do každé zprávy, jak je požadováno 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 (do rozsahu témat). Úloha správce front slouží ke směřování publikovaných zpráv na odběratele, kteří zaregistrovali zájem o dané téma zprávy.

Standardní zařízení produktu IBM MQ se používají k distribuci zpráv, takže aplikace mohou využívat všechny funkce, které jsou k dispozici pro existující aplikace produktu IBM MQ. To znamená, že můžete použít trvalé zprávy, abyste získali once-only zajištěné doručení, a že vaše zprávy mohou být součástí transakční jednotky práce, abyste zajistili, že zprávy budou doručeny odběrateli pouze tehdy, jsou-li potvrzeny vydavatelem.

Vydavatelé a publikování

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

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

Kromě rozdělení publikací na všechny lokální odběratele, kteří mají odpovídající odběry, může správce front také distribuovat publikování všem ostatním správcům front, kteří jsou k němu připojeni, buď přímo, nebo prostřednictvím sítě správců front, kteří mají odběratele k tomuto tématu.

V síti typu IBM MQ publish/subscribe může být vydavatelská aplikace také odběratelem.

Publikace pod synchronizačním bodem

Vydavatelé mohou volat volání MQPUT nebo MQPUT1 v synchronizačním bodu, aby zahrnul všechny zprávy doručené odběratelům v pracovní jednotce. Je-li zadána volba MQPMO_RETAIN nebo volby doručení tématu NPMMSGDLV nebo PMSGDLV s hodnotami ALL nebo ALLDUR, správce front použije interní volání MQPUT nebo MQPUT1 v synchronizačním bodě v rámci rozsahu volání MQPUT vydavatele nebo volání MQPUT1 vydavatele.

Informace o stavu a událostech

Publikace mohou být kategorizovány buď jako stavová publikace, jako např. aktuální cena akcie nebo publikování událostí, jako je například obchod s touto zásobami.

Publikování stavu

Státní publikace obsahují informace o aktuálním stavu něčeho, jako je například cena akcie nebo aktuální skóre fotbalového utkání. 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 odesílat nové informace pokaždé, když se stav změní.

Pokud publikace obsahuje informace o stavu, je často publikována jako zachovaná publikace. Nový odběratel obvykle okamžitě chce informace o aktuálním stavu okamžitě; odběratel nechce čekat na událost, která způsobí opětovné publikování informací. Odběratelé automaticky obdrží zachované publikování tématu, pokud se přihlásí k odběru, pokud odběratel nepoužije volby MQSO_PUBLICATIONS_ON_REQUEST nebo MQSO_NEW_PUBLICATIONS_ONLY.

Publikování událostí

Publikování událostí obsahují informace o jednotlivých událostech, k nimž dochází, jako je například obchod s některými akciemi nebo hodnocení konkrétního cíle. Každá událost je na ostatních událostech nezávislá.

Odběratel bude chtít přijímat informace o událostech, jak se stávají.

Zachovaná publikování

Ve výchozím nastavení je po odeslání všem zainteresovaným odběratelům vyřazen. Vydavatel však může určit, že kopie publikace bude zachována, aby mohla být odeslána na budoucí odběratele, kteří se zajímají o dané téma.

Odstranění publikací poté, co byly odeslány všem zainteresovaným odběratelům, je vhodné pro informace o události, ale není vždy vhodné pro informace o stavu. Pokud zachováte zprávu, nové odběratele nemusí čekat, až budou informace znovu publikovány, dříve než obdrží počáteční informace o stavu. Například odběratel s odběrem na akciovou cenu by dostal aktuální cenu rovnou, aniž by čekal na změnu ceny akcií (a tedy i znovu zveřejnit).

Správce front může zachovat pouze jedno publikování pro každé téma, takže existující zachované publikování tématu je při doručení nového zachovaného publikování ve správci front odstraněno. Avšak odstranění existující publikace se nemusí vyskytnout synchronně s příchodem nové zachované publikace. Proto pokud je to možné, neodesílá více vydavatelů žádná publikování v libovolném tématu.

Odběratelé mohou určit, že nechtějí přijímat zachovaná publikování, pomocí volby odběru MQSO_NEW_PUBLICATIONS_ONLY. Existující odběratelé mohou požádat o odeslání duplicitní kopie zachovaných publikování.

Existují okamžiky, kdy si možná nebudete chtít uchovat publikace, dokonce i pro informace o stavu:

- Pokud jsou všechny odběry tématu provedeny před provedením jakýchkoli publikování v daném tématu a neočekáváte nebo nepovolíte nové odběry, není třeba uchovávat žádné publikace, protože jsou doručeny kompletním množstvím odběratelů při prvním publikování těchto odběrů.
- Pokud k publikování dochází často, jako např. za sekundu, obdrží nový odběratel (nebo odběratel zotavující se z selhání) aktuální stav téměř okamžitě po jejich počátečním odběru, takže tyto publikace není třeba uchovávat.
- Jsou-li publikace velké, můžete pro každé téma uložit zachované publikování, které potřebuje značnou velikost úložného prostoru. V prostředí s více správci front jsou zachovaná publikování uložena všemi správci front v síti, které mají odpovídající odběr.

Při rozhodování o tom, zda mají být použity zachované publikace, zvažte možnost zotavení aplikací z odběru ze selhání. Pokud vydavatel nepoužívá zachovaná publikování, může být aplikace odběratele lokálně uložena do svého aktuálního stavu.

Chcete-li se ujistit, že je publikace zachována, použijte volbu vložení zprávy MQPMO_RETAIN. Je-li tato volba použita a publikování nelze zadržet, zpráva se nepublikuje a volání selže s hodnotou MQRC_PUT_NOT_RETAILED.

Je-li zpráva zachovaná publikování, je tato zpráva označena vlastností zprávy MQIsRetained . Perzistence zprávy je tak, jak byla, když byla původně publikována.

Související pojmy

Aspekty návrhu pro zachovaná publikování v klastrech publikování/odběru

Publikace pod synchronizačním bodem

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

Vydavatelé používají synchronizační bod, když vydají volání MQPUT/MQPUT1 s volbou MQPMO_SYNCPOINT. Všechny zprávy doručené odběratelům se započítávají do maximálního počtu nepotvrzených zpráv v jednotce work.The Atribut správce front MAXUMSGS uvádí

tento limit. Je-li dosaženo limitu, vydavatel obdrží kód příčiny 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_REACHED.

Když vydavatel vydá volání MQPUT/MQPUT1 pomocí volby MQPMO_NO_SYNCPOINT s volbou MQPMO_RETAIN nebo s volbami doručení tématu NPMSGDLV/PMSGDLV s hodnotami ALL nebo ALLDUR, správce front použije vnitřní synchronizační body k zaručení toho, že jsou zprávy doručeny podle požadavku. Vydavatel může přijmout kód příčiny 2024 (07E8) (RC2024): MQRC_SYNCPOINT_LIMIT_REACHED, je-li dosažen limit v rozsahu volání MQPUT/MQPUT1 vydavatele.

Odběratelé a odběry

V produktu IBM MQ publish/subscribe je odběratel aplikací, který vyžaduje informace o specifickém tématu ze správce front v síti typu publikování/odběr. Odběratel může přijímat zprávy o stejných nebo různých tématech, a to od více než jednoho vydavatele.

Odběry mohou být vytvořeny ručně pomocí příkazu MQSC nebo aplikací. Tyto odběry jsou vydávány pro lokálního správce front a obsahují informace o publikacích, které chce odběratel přijmout:

- Téma, o které se odběratel zajímá; lze jej přeložit na více témat, jsou-li použity zástupné znaky.
- Volitelný výběrový řetězec, který má být použit na publikované zprávy.
- Manipulátor s frontou (označovanou jako *fronta odběratele*), na které mají být umístěny vybrané publikace, a volitelné CorrelId.

Lokální správce front ukládá informace o odběru a poté, co obdrží publikování, zkontroluje informace a určí, zda existuje odběr, který odpovídá tématu publikování a výběru řetězce. Pro každý odpovídající odběr správce front směřuje publikování do fronty odběratele odběratele. Informace, které správce front ukládá o odběrech, lze zobrazit pomocí příkazů služby DIS SUB a DIS SBSTATUS.

Odběr se odstraní pouze v případě, že dojde k jedné z následujících událostí:

- Odběratel se nepřihlašuje k odběru pomocí volání MQCLOSE (pokud byl odběr proveden nedurivě).
- Platnost odběru vyprší.
- Přihlášení je odstraněno administrátorem systému pomocí příkazu DELETE SUB.
- Aplikace odběratele se ukončí (pokud byl odběr proveden nedurivě).
- Správce front je zastaven nebo znovu spuštěn (pokud byl odběr proveden nedurivě).

Při získávání zpráv použijte příslušné volby na volání MQGET. Pokud vaše aplikace zpracovává pouze zprávy pro jeden odběr, měli byste použít `get-by-correlid`, jak je předvedeno v ukázkovém programu `C amqssbxa.c` a v nespravovaném odběrateli MQ. Hodnota **CorrelId** pro použití je vrácena z MQSUB v MQSD. Pole **SubCorrelId**.

Související pojmy

Klonované a sdílené odběry

Související odkazy

Příklady toho, jak definovat vlastnost sharedSubscription

Spravované fronty a publikování/odběr

Při vytváření odběru můžete zvolit způsob použití spravovaných front. Pokud použijete spravované fronty fronty odběru, bude automaticky vytvořena při vytvoření odběru. Spravované fronty se tišují automaticky v souladu se stálostí odběru. Použití spravovaných front znamená, že se nemusíte starat o vytváření front pro příjem publikování a všechny nespotebované publikování budou automaticky odebrány z front odběratele, je-li uzavřeno netrvalé připojení k odběru.

Pokud aplikace nemá potřebu používat určitou frontu jako její frontu odběratele, cíl pro publikování, které přijímá, může využívat *spravované odběry* pomocí volby odběru MQSO_MANAGED. Pokud vytvoříte spravovaný odběr, správce front vrátí obslužnou rutinu objektu pro odběratele pro frontu odběratele, kterou správce front vytvoří tam, kde budou přijaty publikování. Důvodem je to, že *spravované odběry* jsou takové, kde IBM MQ zpracovává odběr. Bude vrácen popisovač objektu fronty, který vám umožní procházet, získat nebo dotázat se na frontu (není možné vkládat nebo nastavovat atributy spravované fronty, pokud jste explicitně neposkytl přístup k dočasným dynamickým frontám).

Trvalost odběru určuje, zda je spravovaná fronta i nadále po přerušení připojení aplikace odběru ke správci front po přerušení.

Spravované odběry jsou obzvláště užitečné při použití s netrvalnými odběry, protože při ukončení připojení aplikace zůstanou nespoteřované zprávy ve frontě odběratele nadále zabrané ve správci front neomezeně dlouho. Pokud používáte spravovaný odběr, bude spravovaná fronta dočasnou dynamickou frontou a jako taková bude odstraněna spolu s libovolnými nespoteřovanými zprávami při přerušení připojení z následujících důvodů:

- Je použit příkaz MQCLOSE s MQCO_REMOVE_SUB a spravovaný objekt Hobj je uzavřen.
- dojde ke ztrátě připojení k aplikaci s použitím trvalého odběru (MQSO_NON_DURABLE).
- Odběr byl odebrán, protože jeho platnost vypršela a spravovaný objekt Hobj je uzavřen.

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

Chcete-li používat trvalou dynamickou spravovanou frontu, můžete nastavit vypršení platnosti, takže ačkoli fronta bude stále existovat i po přerušení spojení, nebude nadále existovat po neomezenou dobu.

Pokud odstraníte spravovanou frontu, obdržíte chybovou zprávu.

Spravované fronty, které jsou vytvořeny, jsou pojmenovány s čísly na konci (časová razítka) tak, aby byla každá 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 nespotřebovaným 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, nespotřebované 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](#)

Výběrové řetězce

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

Když se přihlásíte k odběru, kromě uvedení tématu můžete zadat řetězec výběru pro výběr publikací podle jejich vlastností zprávy.

Výběrový řetězec je vyhodnocen proti zprávě, jak ji vložil vydavatel, před tím, než je upraven pro doručení každému odběrateli. Buďte opatrní při použití polí v řetězci výběru, který může být upraven jako součást operace publikování. Např. pole MQMD UserIdentifier, MsgIda CorrelId.

Výběrové řetězce by se neměly odkazovat na žádné z polí vlastností zprávy přidáných správcem front jako součást operace publikování (viz téma [Vlastnosti publikování/odběru zpráv](#)), kromě vlastnosti zprávy MQTopicString, která obsahuje řetězec tématu pro publikování.

Související pojmy

[Pravidla řetězce výběru a omezení](#)

Témata

Téma je předmětem informací, které jsou publikovány ve zprávě publikování/odběru.

Zprávy v systému PPP se posílají na určitou cílovou adresu. Zprávy v systémech publikování/odběru založené na předmětu jsou odeslány odběratelům na základě předmětu, který popisuje obsah zprávy. V systémech založených na obsahu se zprávy odesílají odběratelům na základě obsahu samotné zprávy.

Systém publikování/odběru IBM MQ je systém založený na předmětu a systém odběru. Vydavatel vytvoří zprávu a publikuje ji spolu s řetězcem tématu, který nejlépe odpovídá předmětu publikování. Chcete-li přijímat publikování, odběratel vytvoří odběr s použitím řetězce tématu shodujícího se na vzorek, který bude vybírat témata publikování. Správce front doručuje publikování odběratelům, kteří mají odběry, které odpovídají tématu publikování, a jsou autorizováni k přijetí těchto publikování. Článek, "[Řetězce tématu](#)" na stránce 64, popisuje syntaxi řetězců témat, které identifikují předmět publikace. Odběratelé také vytvářejí řetězce témat k výběru témat, která mají být zasílána. Řetězce témat, které odběratelé mohou vytvořit, mohou obsahovat dva alternativní schéma zástupných znaků pro shodu vzoru vůči řetězcům témat v příručkách. Porovnávání se vzory je popsáno v "[Režimy zástupných znaků](#)" na stránce 65.

V publikovat/subscribe-based publish/subscribe, vydavatelé nebo administrátoři jsou zodpovědní za klasifikaci předmětů do témat. Typicky jsou témata uspořádána hierarchicky, do stromu témat pomocí

znaku ' / ' pro vytvoření podtémat v řetězci tématu. Příklady stromů témat viz [“Stromy témat”](#) na stránce 71 . Témata jsou uzly ve stromu témat. Témata mohou být listové uzly bez dalších dílčích témat nebo zprostředkujících uzlů s dílčími tématy.

Paralelně s uspořádáním předmětů do hierarchického stromu témat můžete asociovat témata s objekty administrativních témat. Atributy přiřazujete k tématu, jako je například informace o tom, zda je téma rozděleno do klastru, a to tak, že jej přidružíte k objektu tématu administrace. Přidružení se provádí pojmenováním tématu pomocí atributu TOPICSTR objektu administrativního tématu. Pokud k tématu explicitně nepřidružíte objekt administrativního tématu, téma zdědí atributy nejbližšího předchůdce ve stromu témat, který máte přidružený k objektu administrativního tématu. Pokud jste nedefinovali žádná nadřazená témata vůbec, zdědí se z SYSTEM . BASE . TOPIC . Objekty administrativního tématu jsou popsány v tématu [“Objekty administrativního tématu”](#) na stránce 72.

Poznámka: I v případě, že dědíte všechny atributy tématu z SYSTEM . BASE . TOPIC , definujte kořenové téma pro svá témata, která přímo dědí z SYSTEM . BASE . TOPIC . Například v prostoru témat amerických států, USA/Alabama USA/Alaskaatd. USA je kořenové téma téma. Hlavním účelem kořenového tématu je vytvořit diskretní, nepřekrývající se prostory témat, aby se zabránilo publikování shodujících se s chybnými odběry. To také znamená, že můžete změnit atributy vašeho kořenového tématu, aby se ovlivnilo celý prostor tématu. Můžete například nastavit název atributu **CLUSTER** .

Když se odkazujete na téma jako vydavatel nebo odběratel, máte možnost zadat řetězec tématu nebo odkazovat na objekt tématu. Nebo můžete provést obojí a v tom případě zadaný řetězec tématu definuje podtéma objektu tématu. Správce front identifikuje téma připojením řetězce tématu k předponě řetězce tématu uvedeným v objektu tématu a vkládá mezi tyto dva řetězce témat další prvek ' / ' , například *string/string string/object string* . [“Kombinování řetězců témat”](#) na stránce 69 popisuje toto dále. Výsledný řetězec tématu se používá k identifikaci tématu a k jeho přidružení k objektu administrativního tématu. Objekt administrativního tématu nemusí být nutně totožný s objektem tématu jako objekt tématu odpovídající hlavnímu tématu.

Při publikování/odběru založeného na obsahu definujete, které zprávy chcete přijímat, poskytnutím řetězců výběru, které prohledávají obsah každé zprávy. Produkt IBM MQ poskytuje zprostředkující formu publikování/odběru založeného na obsahu pomocí selektorů zpráv, které skenují vlastnosti zpráv, a ne úplný obsah zprávy, viz [Selektory](#) . Archetypní použití selektorů zpráv má být přihlášené k odběru tématu a poté kvalifikovat výběr pomocí číselné vlastnosti. Selektor vám umožňuje uvést, že se zajímáte o hodnoty pouze v určitém rozsahu; něco, co nemůžete provést pomocí zástupných znaků nebo zástupných znaků založených na tématu. Pokud potřebujete filtrovat na základě úplného obsahu zprávy, je třeba použít produkt IBM Integration Bus.

Řetězce tématu

Informace o jmenovkách, které publikujete jako téma pomocí řetězce tématu. Přihlaste se k odběru skupin témat pomocí řetězců témat zástupných znaků založených na znacích nebo tématu.

Témata

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



Tři znaky mají speciální význam v publikaci IBM WebSphere MQ 7 publish/subscribe. Jsou povoleny kdekoli v řetězci tématu, ale používají je s opatrností. Použití speciálních znaků je vysvětleno v části [“Schéma zástupných znaků založené na tématu”](#) na stránce 65.

dopředné lomítko (/)

Oddělovač úrovně témat. Chcete-li strukturovat téma do stromu témat, použijte znak ' / ' .

Vyhňte se prázdným úrovním tématu, ' / ' , pokud můžete. Tyto odpovídají uzlům v hierarchii témat bez řetězce tématu. Vedoucí nebo koncový znak ' / ' v řetězci tématu odpovídá počátečnímu nebo koncovému prázdnému uzlu a je třeba se mu vyhnout.

Znak křížku (#)

Používá se v kombinaci s ' / ' k vytvoření zástupného znaku více úrovní v odběrech. Dávejte pozor na ' # ' sousedící s ' / ' v řetězcích témat použitých k pojmenování publikovaných témat. [“Příklady řetězců témat”](#) na stránce 65 ukazuje citlivé použití ' # ' .

Řetězce ' . . . / # / . . . ' , ' # / . . . ' a ' . . . / # ' mají speciální význam v řetězcích témat odběru. Řetězce se shodují se všemi tématy na jedné nebo více úrovních v hierarchii témat. Takže pokud jste vytvořili téma s jednou z těchto posloupností, nelze se k němu přihlásit, aniž byste se také přihlásili k odběru všech témat na více úrovních v hierarchii témat.

Znaménko plus (+)

Používá se v kombinaci s ' / ' k vytvoření zástupného znaku single-level v odběrech. Dávejte pozor na ' + ' sousedící s ' / ' v řetězcích témat použitých k pojmenování publikovaných témat.

Řetězce ' . . . / + / . . . ' , ' + / . . . ' a ' . . . / + ' mají speciální význam v řetězcích témat odběru. Řetězce se shodují se všemi tématy na jedné úrovni v hierarchii témat. Takže pokud jste vytvořili téma s jednou z těchto posloupností, nemůžete se přihlásit k odběru, aniž byste se také 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](#)

Režimy zástupných znaků

K odběru více témat se používají dva režimy zástupných znaků. Volba schématu je volba odběru.

MQSO_WILDCARD_TOPIC

Chcete-li se přihlásit k odběru pomocí schématu zástupných znaků založeného na tématu, vyberte témata.

Toto je výchozí, pokud není explicitně vybráno žádné schéma zástupného znaku.

MQSO_WILDCARD_CHAR

Chcete-li se přihlásit k odběru pomocí schématu zástupných znaků pro charakterizaci, vyberte témata.

Nastavte buď 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 zástupných znaků s použitím znaků.

Příklady

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

Schéma zástupných znaků založené na tématu

Zástupné znaky založené na tématu umožňují odběratelům přihlásit se k odběru více než jednoho tématu současně.

Zástupné znaky založené na tématu jsou mocnou funkcí systému témat v publikaci 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 zástupných znaků založené na tématu vám umožňuje vybrat publikace seskupené podle úrovně témat. Můžete zvolit, pro každou úroveň v hierarchii témat, zda řetězec v odběru pro danou úroveň tématu se musí přesně shodovat s řetězcem v publikování, nebo ne. Například odběr IBM/+/Results vybírá všechna témata.

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

Existují dva typy zástupných znaků.

Zástupný znak více úrovní

- Zástupný znak více úrovní se používá v předplatných. Při použití v publikaci se s ním zachází jako s literálem.
- Zástupný znak více úrovní '#' se používá ke shodě s libovolným počtem úrovní v rámci tématu. For example, using the example topic tree, if you subscribe to 'USA/Alaska/#', you receive messages on topics 'USA/Alaska' and 'USA/Alaska/Juneau'.
- Zástupný znak více úrovní může představovat nula nebo více úrovní. Proto 'USA/#' může také odpovídat singulární 'USA', kde '#' představuje nulové úrovně. Oddělovač úrovně témat v tomto kontextu postrádá smysl, protože nejsou žádné úrovně k oddělení.
- Zástupný znak více úrovní je platný pouze v případě, že je zadán samostatně nebo vedle znaku oddělovače úrovně tématu. Proto platí, že '#' a 'USA/#' jsou platná témata, kde se s znakem '#' zachází jako se zástupným znakem. Ačkoli však 'USA#' je také platným řetězcem tématu, znak '#' se nepovažuje za zástupný znak a nemá žádný speciální význam. Další informace viz [“Pokud zástupné znaky založené na tématu nejsou zástupné”](#) na stránce 68.

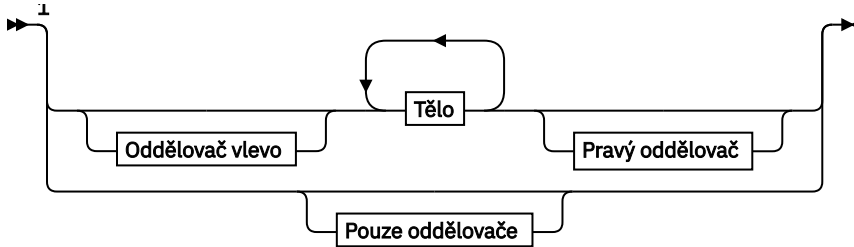
Zástupný znak jedné úrovně

- Jeden zástupný znak se používá v odběrech. Při použití v publikaci se s ním zachází jako s literálem.
- Zástupný znak single-level '+' odpovídá jednomu, a pouze jedné úrovni tématu. Například, 'USA/+' odpovídá 'USA/Alabama', ale ne 'USA/Alabama/Auburn'. Protože zástupný znak single-level odpovídá pouze jedné úrovni, 'USA/+' se neshoduje s 'USA'.
- Zástupný znak jednotného místa lze použít na libovolné úrovni stromu témat a ve spojení s víceúrovňového zástupným znakem. Zástupný znak jednotného místa musí být zadán vedle oddělovače úrovně témat, s výjimkou případů, kdy je tento oddělovač zadán samostatně. Proto platí, že '+' a 'USA/+' jsou platná témata, kde se s znakem '+' zachází jako se zástupným znakem. Ačkoli však 'USA+' je také platným řetězcem tématu, znak '+' se nepovažuje za zástupný znak a nemá žádný speciální význam. Další informace viz [“Pokud zástupné znaky založené na tématu nejsou zástupné”](#) na stránce 68.

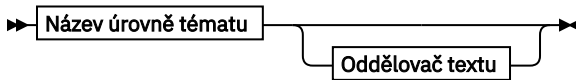
Syntaxe pro schéma zástupných znaků založené na tématu nemá žádné řídicí znaky. Zda jsou '#' a '+' považovány za zástupné znaky nebo ne, závisí na jejich kontextu. Další informace viz [“Pokud zástupné znaky založené na tématu nejsou zástupné”](#) na stránce 68.

Poznámka: Počátek a konec řetězce tématu je zpracováván speciálním způsobem. Using '\$' označuje konec řetězce, pak '\$#/...' je víceúrovňový zástupný znak a '\$#/...' je prázdný uzel v kořenovém adresáři, následovaný víceúrovňovým zástupným znakem.

Řetězec zástupných znaků založený na tématu



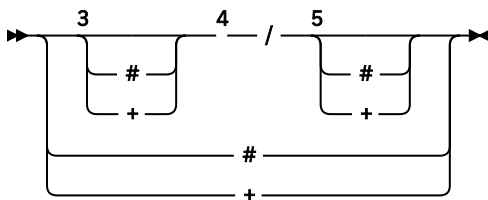
Tělo



Název úrovně tématu



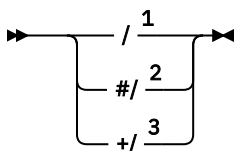
Pouze oddělovače



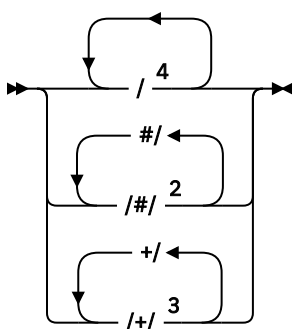
Poznámky:

- 1 Řetězec tématu s hodnotou Null nebo s nulovou délkou je neplatný.
- 2 Doporučuje se nepoužívat žádný z řetězců názvu produktu *, ?, % pro kompatibilitu s použitím zástupných schémat založených na řetězci a schématech založených na tématu.
- 3 Tyto případy jsou ekvivalentní se vzorem *left delimiter* .
- 4 / bez zástupných znaků se shoduje s jedním prázdným tématem.
- 5 Tyto případy jsou ekvivalentní ke vzoru *right delimiter* .
- 6 Shoda se všemi tématy.
- 7 Shoda každé téma tam, kde je pouze jedna úroveň.

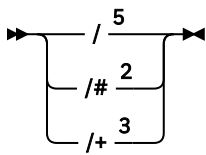
Levý oddělovač



Oddělovač textu



Pravý oddělovač



Poznámky:

- 1 Řetězec tématu začíná prázdným tématem
- 2 Odpovídá nule nebo více úrovním. Vícenásobný řetězec shody na úrovni více úrovní má stejný účinek jako jeden řetězec s více shodnými řetězci.
- 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

Zástupné znaky '+' a '#' nemají žádný speciální význam, jsou-li směřovány s ostatními znaky (včetně sebe) na úrovni tématu.

To znamená, že témata, která obsahují '+' nebo '#' společně s ostatními znaky na úrovni témat, mohou být publikována.

Vezměme si 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, který má být publikován, ale je platný v odběru.

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

Příklady

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

Schéma zástupných znaků založených na znacích

Schéma zástupného znaku na základě znaků vám umožňuje vybrat témata založená na tradiční shodě znaků.

Můžete vybrat všechna témata na více úrovních v hierarchii témat pomocí řetězce '*'. Použití produktu '*' ve schématu zástupných znaků s použitím znaků je ekvivalentní použití řetězce zástupného znaku '#' založeného na tématu.

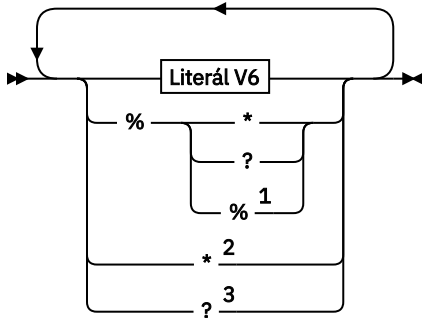
'x*/y' je ekvivalentní příkazu 'x#/y' v rámci schématu založeném na tématu a vybírá všechna témata v hierarchii témat mezi úrovněmi 'x a y', kde 'x' a 'y' jsou názvy témat, které nejsou v sadě úrovní vrácených zástupným znakem.

'/+' v rámci schématu založeného na tématu nemá v rámci schématu na základě znaků žádný přesný ekvivalent. 'IBM*/Results' by také vybral 'IBM/Patents/Software/Results'. Pouze v případě, že je sada názvů témat na každé úrovni hierarchie jedinečná, můžete vždy vytvářet dotazy se dvěma schématy, které vedou identické shody.

Obecně platí, že '*' a '?' v systému založeném na popisu nemají žádné ekvivalenty v rámci schématu založeném na tématu. Schéma založené na tématu neprovádí částečné shody s použitím zástupných znaků. Odběr zástupného znaku založený na znacích 'IBM/*ware/Results' neobsahuje žádný ekvivalent na základě témat.

Poznámka: Shody používající zástupné znaky zástupných znaků jsou pomalejší než shody s použitím odběrů založených na tématu.

Řetězec zástupných znaků založený na znacích



Literál V6

► Jakýkoliv znak Unicode kromě *,? a% ◄

Poznámky:

- ¹ Znamená únik následujícího znaku, takže se s ním zachází jako s literálem. Za '%' musí následovat buď '*', '?', nebo '%'. Viz "Příklady řetězců témat" na stránce 65.
- ² Neodpovídá žádnému nebo více znakům v odběru.
- ³ Prostředky odpovídají přesně jednomu znaku v odběru.

Příklady

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

Kombinování řetězců témat

Při vytváření odběrů nebo otevírání témat, abyste pro ně mohli publikovat zprávy, může být řetězec tématu vytvořen kombinací dvou samostatných řetězců dílčích témat nebo "dílčích témat". Jeden dílčí téma je poskytováno aplikací nebo administrativním příkazem jako řetězec tématu a druhým je řetězec tématu přidružený k objektu tématu. Jako řetězec tématu můžete použít buď dílčí téma jako řetězec tématu, nebo je zkombinovat a vytvořit nový název tématu.

Pokud například definujete odběr pomocí příkazu MQSC **DEFINE SUB**, může příkaz provést buď **TOPICSTR** (řetězec tématu), nebo **TOPICOBJ** (objekt tématu) jako atribut, nebo oboje současně. Je-li zadán pouze příznak **TOPICOBJ**, bude řetězec tématu asociovaný s daným objektem tématu použit jako řetězec tématu. Je-li zadán pouze příznak **TOPICSTR**, bude použit jako řetězec tématu. Jsou-li zadány oba parametry, jsou zřetězeny tak, že tvoří jeden řetězec tématu ve tvaru **TOPICOBJ / TOPICSTR**, kde je řetězec tématu **TOPICOBJ** konfigurován vždy jako první a obě části řetězce jsou vždy odděleny znakem **"/"**.

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

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

Výsledný řetězec tématu se vrátí v parametru **ResObjectString**.

Tato pole se považují za přítomná, pokud první znak každého pole není prázdný znak nebo znak null a délka pole je větší než nula. Je-li přítomno pouze jedno z těchto polí, použije se nezměněno jako název tématu. Pokud pole nemá hodnotu, volání selže s kódem příčiny MQRC_UNKNOWN_OBJECT_NAME nebo MQRC_TOPIC_STRING_ERROR, je-li úplný název tématu neplatný.

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

Tabulka 2 na stránce 70 zobrazuje příklady zřetězení řetězce tématu:

<i>Tabulka 2. Příklady zřetězení řetězce tématu</i>			
TOPICSTR objektu tématu	Řetězec tématu poskytnutý aplikací nebo příkazem DEFINE SUB	Úplný název tématu	Komentář
Fotbal/Scores	' '	Fotbal/Scores	TOPICSTR objektu tématu se používá samostatně.
' '	Fotbal/Scores	Fotbal/Scores	Parametr ObjectString/ TOPICSTR se používá samostatně.
Fotbal	Skóre	Fotbal/Scores	Znak "/" je přidán do bodu zřetězení.
Fotbal	/Skóre	Fotbal//skóre	Mezi dvěma řetězci se vytváří 'prázdný uzel'. To se liší od položky "Fotbal/Skóre".
/Fotbal	Skóre	/Fotbal/Scores	Téma začíná znakem 'empty node'. To se liší od položky "Fotbal/Skóre".

Znak "/" je považován za speciální znak a poskytuje strukturu úplnému názvu tématu v produktu "Stromy témat" na stránce 71. Znak "/" nesmí být použit z žádného jiného důvodu, protože struktura stromu témat je ovlivněna. Téma "/Football" není stejné jako téma "Football".

Poznámka: Použijete-li při vytváření odběru objektu tématu, bude hodnota řetězce tématu objektů tématu v rámci odběru definována v době definování. Jakákoli další změna objektu tématu nemá vliv na řetězec tématu, na který je odběr definován.

Zástupné znaky v řetězcích témat

Následující zástupné znaky jsou speciální znaky:

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

Zástupné znaky mají při použití u odběru speciální význam. Tyto znaky se při použití jinde nepovažují za neplatné, ale musíte zajistit, abyste porozuměli tomu, jak jsou tyto znaky používány, a při publikování nebo definování objektů tématu můžete raději nepoužívat tyto znaky ve svých řetězcích témat.

Pokud publikujete na řetězec tématu s # nebo + smíchaným s ostatními znaky (včetně samotných) v rámci úrovně tématu, může být řetězec tématu přihlášen k odběru buď pomocí zástupných schémat.

Pokud publikujete na řetězec tématu s # nebo + jako jediný znak mezi dvěma znaky / , řetězec tématu nemůže být odebírán explicitně aplikací pomocí schématu zástupného znaku MQSO_WILDCARD_TOPIC. Tato situace vede k tomu, že aplikace bude dostávat více publikací, než se očekávalo.

V řetězci tématu definovaného objektu tématu byste neměli používat zástupný znak. Pokud tak učiníte, je tento znak považován za literálový znak, je-li objekt používán vydavatelem, a jako zástupný znak při použití odběru. To může vést k zmatku.

Příklad úseku kódu

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

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

Stromy témat

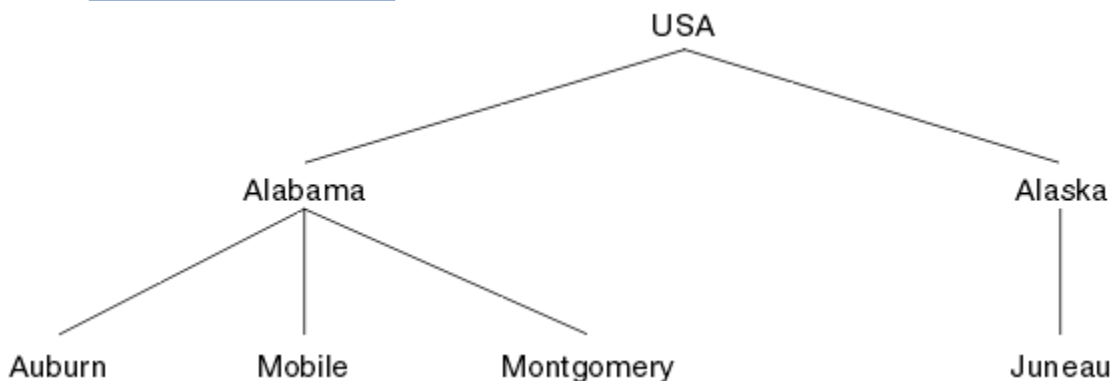
Každé téma, které definujete, je prvkem, neboli uzlem, stromu témat. Strom témat může být buď prázdný, aby mohl začínat nebo obsahovat témata, která byla definována dříve pomocí příkazů MQSC nebo PCF. Nové téma můžete definovat buď pomocí příkazů k vytvoření tématu, nebo zadáním tématu pro první publikování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

I když můžete použít libovolný řetězec znaků k definování řetězce tématu, je vhodné zvolit řetězec tématu, který se vejde do hierarchické stromové struktury. Promyšlený návrh struktur témat a stromů témat vám může pomoci s následujícími operacemi:

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

Ačkoli můžete vytvořit strom témat jako fixní, 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 [Zabezpečení publikování a odběru](#).

Příklad [Obrázek 18](#) na stránce 71 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 se vytváří agregací uzlů z jedné nebo více úrovní ve stromu témat. Úrovně jsou odděleny znakem "/". Formát plně specifikovaného řetězce témat je: "root/level2/level3".

Platná témata ve stromu témat zobrazeném v produktu [Obrázek 18](#) na stránce 71 jsou:

```
'USA'
"USA/Alabama"
"USA/Aljaška"
"USA/Alabama/Auburn"
"USA/Alabama/Mobile"
"USA/Alabama/Montgomery"
"USA/Aleska/Juneau"
```

Když navrhujete řetězce témat a stromy témat, nezapomeňte, že správce front není interpretován nebo se pokouší odvodit význam samotného řetězce tématu. 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 omezen počtem úrovní.
- Délka názvu ve stromu témat není nijak omezena délkou názvu.
- Může existovat libovolný počet "kořenových" uzlů; to znamená libovolný počet stromů témat.

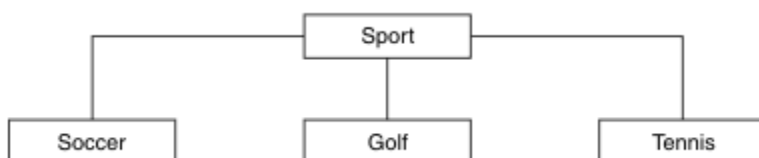
Související úlohy

Snížení počtu nežádoucích témat ve stromu témat

Objekty administrativního tématu

Pomocí objektu administrativního tématu můžete jednotlivým tématům přiřazovat specifické, nevýchozí atributy.

Obrázek 19 na stránce 72 ukazuje, jak lze vizualizovat téma vysoké úrovně Sport rozdělené do samostatných témat pokrývajících různé sporty jako strom témat:



Obrázek 19. Vizualizace stromu témat

Obrázek 20 na stránce 72 ukazuje, jak lze strom témat rozdělit dále, aby oddělil různé typy informací o každém sportu:



Obrázek 20. Rozšířený strom témat

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

Všimněte si, že není dobrým zvykem dávat oprávnění pro uživatele, kteří nejsou mqm, v kořenovém uzlu stromu témat, což je `SYSTEM.BASE.TOPIC`, protože oprávnění jsou děděna, ale nemohou být omezena. Tím, že udělavám orgánům na této úrovni, dáváte úřadům celý strom. Oprávnění by mělo být uděleno na nižší úrovni témat v hierarchii.

Administrační objekty témat lze použít k definování specifických atributů pro konkrétní uzly ve stromu témat. V následujícím příkladu je definován objekt administrativního tématu k nastavení vlastnosti trvalé odběry DURSUB na fotbalové téma na hodnotu NO:

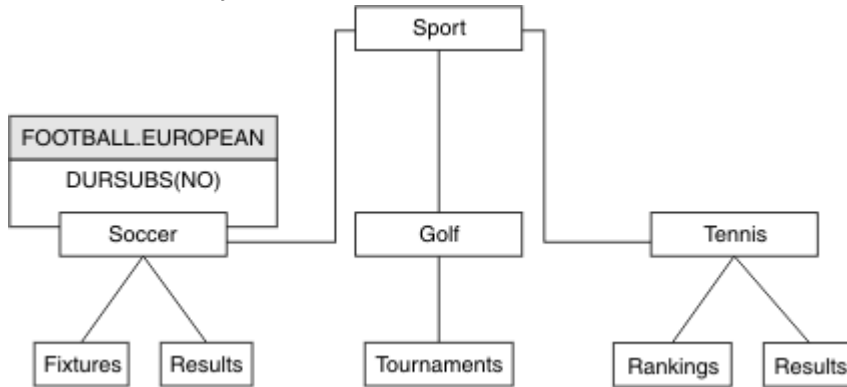
```

DEFINE TOPIC (FOOTBALL.EUROPEAN)
TOPICSTR ('Sport/Soccer')
  
```



```
DURSUB(NO)
DESCR('Administrative topic object to disallow durable subscriptions')
```

Strom témat lze nyní vizualizovat takto:



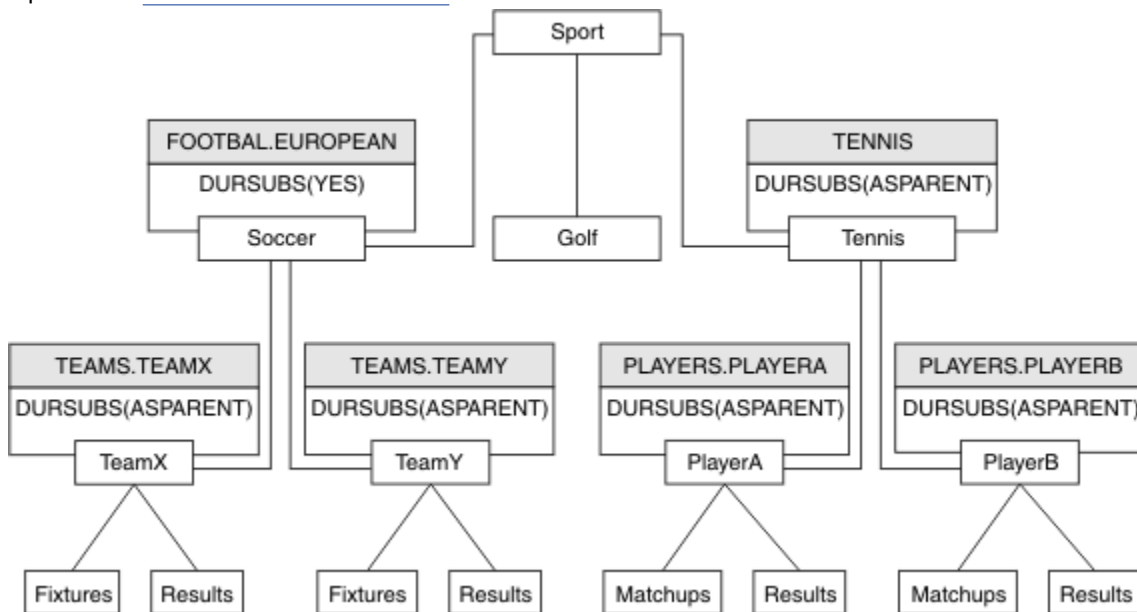
Obrázek 21. Vizualizace objektu administrativního tématu přidruženého k tématu Sport/Soccer

Všechny aplikace, které jsou přihlášeny k odběru témat pod Soccer ve stromu, stále mohou používat řetězce témat, které byly použity před přidáním objektu administrativního tématu. Nyní však může být aplikace zapsána k odběru s použitím názvu objektu FOOTBALL . EUROPEAN, namísto řetězce /Sport/Soccer. Chcete-li se například přihlásit k odběru /Sport/Soccer/Results, může aplikace zadat MQSD.ObjectName jako FOOTBALL . EUROPEAN a MQSD.ObjectString jako Results.

Pomocí této funkce můžete skrýt část stromu témat od vývojářů aplikací. Definujte objekt administrativních témat v konkrétním uzlu ve stromu témat a vývojáři aplikací pak mohou definovat vlastní témata jako podřízené prvky uzlu. Vývojáři musí vědět o nadřazeném tématu, ale ne o žádném jiném uzlu v nadřazeném stromu.

Zdědění atributů

Pokud má strom témat mnoho objektů administrativních témat, každý objekt tématu správy standardně dědí své atributy od nejbližšího nadřazeného administrativního tématu. Předchozí příklad byl rozšířen v produktu [Obrázek 22 na stránce 73](#):



Obrázek 22. Strom témat s několika objekty administrativního tématu

Pokud například použijete dědičnost k poskytnutí všech podřízených témat /Sport/Soccer, vlastnosti, které odběry jsou netrvalé. Změňte atribut DURSUB z FOOTBALL . EUROPEAN na NO.

Tento atribut lze nastavit pomocí následujícího příkazu:

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

Všechny objekty administračních témat podřízených témat produktu Sport/Soccer mají vlastnost DURSUB nastavenou na výchozí hodnotu ASPARENT. Po změně hodnoty vlastnosti DURSUB u FOOTBALL . EUROPEAN na NO podřízená témata Sport/Soccer dědí hodnotu vlastnosti DURSUB NO. Všechna podřízená témata Sport/Tennis dědí hodnotu od objektu DURSUB z objektu SYSTEM . BASE . TOPIC . SYSTEM . BASE . TOPIC má hodnotu YES.

Pokus o vytvoření trvalého odběru pro téma Sport/Soccer/TeamX/Results se nyní nezdaří. Pokus o vytvoření trvalého odběru pro produkt Sport/Tennis/PlayerB/Results by však byl úspěšný.

Řízení použití zástupného znaku s vlastností WILDCARD

Pomocí vlastnosti MQSC Topic WILDCARD nebo ekvivalentních vlastností PCF Topic WildcardOperation můžete řídit doručování příruček aplikacím, které používají řetězce názvů témat zástupných znaků. Vlastnost WILDCARD může mít jednu ze dvou možných hodnot:

WILDCARD

Chování odběrů používajících zástupné znaky s ohledem na toto téma.

PASSTHRU

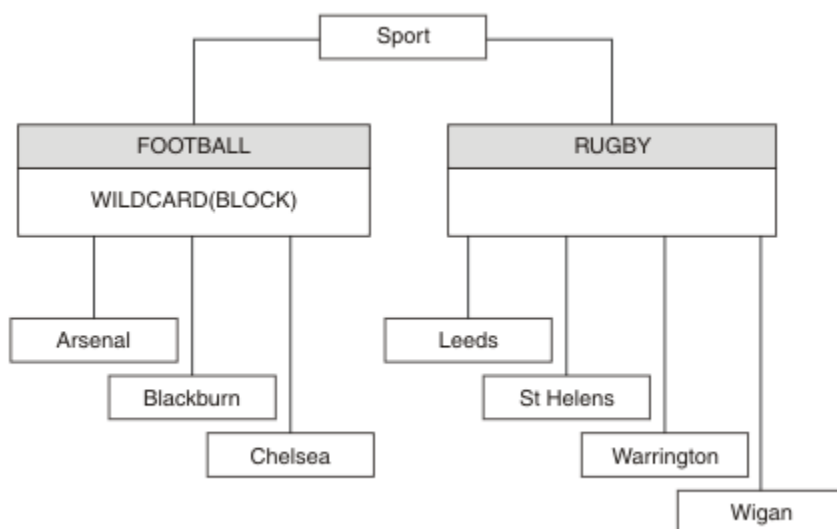
U odběrů registrovaných pro téma, které používá zástupné znaky a které je méně specifické než řetězec tématu v tomto objektu tématu, jsou poskytnuty publikace k tomuto tématu a k řetězcům tématu, které jsou specifičtější než toto téma.

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 pro publikování/odběr produktu Sport”](#) na stránce 78, můžete postupovat podle kroků k vytvoření stromové struktury témat zobrazené v [Obrázek 23](#) na stránce 74.



Obrázek 23. Strom témat, který používá vlastnost WILDCARD, BLOCK.

Odběratel používající řetězec tématu s zástupnými znaky # přijímá všechny publikace k tématu Sport a podstromu Sport/Rugby . Odběratel neobdrží žádné publikace do podstromu Sport/Football , protože hodnota vlastnosti WILDCARD u tématu Sport/Football je BLOCK.

Výchozí nastavení je PASSTHRU . Hodnotu vlastnosti ZÁSTUPNÝ ZNAK PASSTHRU můžete nastavit na uzly ve stromu Sport . Pokud uzly nemají hodnotu vlastnosti WILDCARD BLOCK, nastavení PASSTHRU nezmění chování pozorované odběratelem na uzlech ve stromu Sports .

V tomto příkladu vytvořte odběry a zjistěte, jak nastavení zástupných znaků ovlivňuje doručené publikace, viz Obrázek 27 na stránce 79. Spusťte publikační příkaz v produktu Obrázek 30 na stránce 80 a vytvořte některé publikace.

pub QMA

Obrázek 24. Publikovat na QMA

Výsledky jsou zobrazeny v Tabulka 3 na stránce 75. Všimněte si, jak nastavení hodnoty vlastnosti WILDCARD BLOCK brání odběrům se zástupnými znaky z příjmu publikací na témata v rámci rozsahu zástupného znaku.

Tabulka 3. Publikace přijaté v systému QMA			
Předplatně	Řetězec tématu	Přijaté publikace	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace ve fotbalovém podstromu blokovány uživatelem WILDCARD (BLOCK) v systému Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) v Sports/Football zabraňuje odběru zástupného znaku na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Předvolba WILDCARD na Sports/Rugby nezabrání odběru zástupného znaku na Leeds.

Poznámka:

Předpokládejme, že odběr má zástupný znak, který odpovídá objektu tématu s hodnotou vlastnosti WILDCARD BLOCK. Má-li odběr také řetězec tématu napravo od odpovídajícího zástupného znaku, odběr nikdy neobdrží publikování. Sada publikací, které nejsou blokovány, jsou publikace k tématům, která jsou nadřazenými prvky blokováného zástupného znaku. Publikování do témat, která jsou podřízenými prvky tématu s hodnotou vlastnosti produktu BLOCK , jsou blokovány zástupnými znaky. Proto řetězce témat odběru, které obsahují téma napravo od zástupného znaku, nikdy nepřijímají žádné publikace, které by se shodovaly.

Nastavení hodnoty vlastnosti WILDCARD na hodnotu BLOCK neznámá, že se nemůžete přihlásit k odběru pomocí řetězce tématu, který obsahuje zástupné znaky. Takové předplatně je normální. Odběr má explicitní téma, které odpovídá danému tématu s objektem tématu, který má hodnotu vlastnosti WILDCARD BLOCK. Používá zástupné znaky pro témata, která jsou nadřazenými nebo podřízenými prvky tématu s hodnotou vlastnosti WILDCARD BLOCK. V příkladu v produktu Obrázek 23 na stránce 74 může publikování, jako je 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 ve správci front. Proxy odběr je vytvořen ve všech

ostatních správci front v klastru. Odběry pomocí řetězců témat obsahujících zástupné znaky, kombinované s tématy klastru, mohou být obtížné předpovědět chování. Chování je vysvětleno v následujícím příkladu.

V klastru nastavovaný pro tento příklad, “Příklad: Vytvoření klastru pro publikování/odběr produktu Sport” na stránce 78, má QMB stejnou sadu odběrů jako QMA, ale QMB nepřijala žádné publikace poté, co vydavatel publikován na QMA, viz Obrázek 24 na stránce 75. Ačkoli témata Sports/Football a Sports/Rugby jsou témata klastru, odběry definované v souboru fullsubs.tst se neodkazují na téma klastru. Z produktu QMB do produktu QMA nejsou šířeny žádné odběry proxy. Bez odběrů proxy jsou do produktu QMB předávány žádné publikace QMA.

Některé odběry, jako např. Sports/#/Leeds, se mohou v tomto případě odkazovat na téma klastru Sports/Rugby. Odběr produktu Sports/#/Leeds je ve skutečnosti interpretováno jako objekt tématu SYSTEM.BASE.TOPIC.

Pravidlo pro vyřešení objektu tématu, na který se odkazuje odběr, jako je například Sports/#/Leeds, je následující. Oříznete řetězec tématu na první zástupný znak. Skenování bylo ponecháno přes řetězec tématu, který hledá první téma, které má přidružený objekt administrativního tématu. Objekt tématu může určovat název klastru, nebo definovat lokální objekt tématu. V příkladu, Sports/#/Leeds, řetězec tématu po oseknutí je Sports, který nemá žádný objekt tématu, a tak se Sports/#/Leeds dědí od SYSTEM.BASE.TOPIC, což je lokální objekt tématu.

Chcete-li zjistit, jak může přihlášení k odběru klastrovaných témat změnit způsob, jakým šíření zástupných znaků funguje, spusťte dávkový skript upsubs.bat. Skript vymaže fronty odběru a přidá odběry témat klastru do adresáře fullsubs.tst. Znovu spusťte soubor puba.bat a vytvořte dávku publikací; viz Obrázek 24 na stránce 75.

Produkt Tabulka 4 na stránce 76 zobrazuje výsledek přidání dvou nových odběrů do stejného správce front, ve kterém byly publikovány publikace. Výsledkem je podle očekávání, že nové odběry obdrží jedno publikování a počty publikování přijatých ostatními odběry budou nezměněny. Dojde k neočekávaným výsledkům na druhém správci front klastru, viz Tabulka 5 na stránce 77.

Tabulka 4. Publikace přijaté v systému QMA			
Předplatně	Řetězec tématu	Přijaté publikace	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace ve fotbalovém podstromu blokováné uživatelem WILDCARD (BLOCK) v systému Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) v Sports/Football zabraňuje odběru zástupného znaku na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Předvolba WILDCARD na Sports/Rugby nezabrání odběru zástupného znaku na Leeds.
FARSENAL	Sports/Football/ Arsenal	Sports/Football/ Arsenal	Produkt Arsenal obdrží publikování, protože odběr nemá zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikování v každém případě.

Produkt Tabulka 5 na stránce 77 zobrazuje výsledky přidání dvou nových odběrů v produktu QMB a publikování v produktu QMA. Připomeňme si, že produkt QMB nepřijal žádná publikování bez těchto dvou nových odběrů. Jak bylo očekáváno, tyto dva nové odběry přijímají publikace, protože Sports/Football a Sports/Rugby jsou obě témata klastru. QMB postoupil proxy odběry pro Sports/Football/Arsenal a Sports/Rugby/Leeds do QMA, které následně odeslaly publikace do QMB.

Neočekávaný výsledek je takový, že se dva odběry Sports/# a Sports/#/Leeds, které dříve neobdržely žádné publikace, nyní přijímají publikace. Důvodem je to, 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. V důsledku toho se odběry lokálních témat Sports/# a Sports/#/Leeds přijímají v publikaci Sports/Rugby/Leeds. Produkt Sports/#/Arsenal nadále neobdrží publikování, protože sport/fotbal má nastavenou hodnotu vlastnosti WILDCARD nastavenou na hodnotu BLOCK.

Tabulka 5. Publikace přijaté v systému QMB			
Předplatně	Řetězec tématu	Přijaté publikace	Notes
SPORTS	Sports/#	Sports/Rugby/Leeds	Všechny publikace ve fotbalovém podstromu blokovány produktem WILDCARD (BLOCK) v systému Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) v Sports/Football zabraňuje odběru zástupného znaku na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Výchozí hodnota WILDCARD na Sports/Rugby nezabrání odběru zástupných znaků v systému Leeds.
FARSENAL	Sports/Football/Arsenal	Sports/Football/Arsenal	Produkt Arsenal obdrží publikování, protože odběr nemá zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikování v každém případě.

Ve většině aplikací je nežádoucí pro jeden odběr ovlivnit chování jiného odběru. Jedno důležité použití vlastnosti WILDCARD s hodnotou BLOCK znamená, že se odběry ke stejnému řetězci tématu, které obsahují zástupné znaky, budou chovat jednotně. Údaj o tom, zda je odběr ve stejném správci front jako vydavatel nebo jiný, výsledky odběru jsou stejné.

Zástupné znaky a proudy

Pro novou aplikaci napsanou na rozhraní API pro publikování/odběr je výsledkem, že odběr z produktu * neobdrží žádné publikace. Chcete-li přijmout všechny sportovní publikace, musíte se přihlásit k odběru produktu Sports/*nebo produktu Sports/#a podobně jako publikace Business.

Chování existující aplikace publikování/odběru ve frontě se nezmění, pokud je zprostředkovatel publikování/odběru migrován do produktu IBM WebSphere MQ 7 a novějších verzí. Vlastnost **StreamName** v příkazech **Publish**, **Register Publisher**nebo **Subscriber** je mapována na název tématu, na který byl proud migrován.

Zástupné znaky a body odběru

U nové aplikace napsané pro rozhraní API pro publikování/odběr je výsledkem migrace skutečnost, že odběr produktu * přijímá žádné publikace. Chcete-li přijmout všechny sportovní publikace, musíte se přihlásit k odběru produktu Sports/*nebo produktu Sports/#a podobně jako publikace Business.

Chování existující aplikace publikování/odběru ve frontě se nezmění, pokud je zprostředkovatel publikování/odběru migrován do produktu IBM WebSphere MQ 7 a novějších verzí. Vlastnost **SubPoint** v příkazech **Publish**, **Register Publisher**nebo **Subscriber** je mapována na název tématu, na které byl odběr migrován.

Příklad: Vytvoření klastru pro publikování/odběr produktu Sport

Kroky následující po vytvoření klastru CL1se čtyřmi správci front: dvě úplná úložiště, CL1A a CL1Ba dvě dílčí úložiště, QMA a QMB. Úplná úložiště se používají k zadržení pouze definic klastru. QMA je označen jako hostitel tématu klastru. Trvalé odběry jsou definovány na serveru QMA i v produktu QMB.

Poznámka: Příklad je kódován pro Windows. Chcete-li nakonfigurovat a otestovat tento příklad na jiných platformách, musíte soubor `Create qmgrs.bat` a `create pub.bat` nakonfigurovat a otestovat.

1. Vytvořte skriptové soubory.
 - a. [Vytvoření topics.tst](#)
 - b. [Vytvořit wildsubs.tst](#)
 - c. [Vytvořit fullsubs.tst](#)
 - d. [Vytvořit qmgrs.bat](#)
 - e. [create pub.bat](#)
2. Chcete-li vytvořit konfiguraci, spusťte příkaz `Create qmgrs.bat`.

```
qmgrs
```

Vytvořte témata v produktu [Obrázek 23 na stránce 74](#). Skript na obrázku 5 vytvoří témata klastru Sports/Football a Sports/Rugby.

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

```
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. Odstraňte a vytvořte témata: topics.tst

Poznámka: Odstraňte témata, protože produkt REPLACE nebude nahrazovat řetězce témat.

Vytvoření odběrů se zástupnými znaky. Zástupné znaky odpovídají tématům s objekty témat v produktu [Obrázek 23 na stránce 74](#). Vytvořte frontu pro každý odběr. Fronty jsou vymazány a odběry odstraněny, když je skript spuštěn nebo znovu spuštěn.

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

```

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. Vytvoření zástupných znaků pro zástupný znak: wildsubs.tst

Vytvořte odběry, které odkazují na objekty tématu klastru.

Poznámka:

Oddělovač, /, se automaticky vkládá mezi řetězec tématu, na který odkazuje TOPICOBJ, a řetězec tématu definovaný parametrem TOPICSTR.

Definice DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) vytváří stejný odběr. TOPICOBJ se používá jako rychlý způsob, jak odkazovat na řetězec tématu, který jste již definovali. Když se odběr vytvoří, již se nebude odkazovat 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. Odstraňte a vytvořte odběry: fullsubs.tst

Vytvořte klastr se dvěma úložišti. Vytvořte dvě dílčí úložiště pro publikování a odběr. Znovu spusťte skript k odstranění všeho a začněte znovu. Skript také vytvoří hierarchii témat a počáteční zástupné znaky odběrů.

Poznámka:

Na ostatních platformách napište podobný skript nebo zadejte všechny příkazy. Pomocí skriptu lze rychle odstranit vše a začít znovu se stejnou 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. Vytvoření správců 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. Aktualizovat odběry: upsubs.bat

Spuštěním příkazu pub.bat se správcem front jako parametrem publikujete zprávy obsahující řetězec tématu publikování. Produkt Pub.bat používá vzorový 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 publiku/odběru. Ve frontě publikovat/odebírat proudy poskytují způsob, jak oddělit tok informací pro různá témata. Od produktu IBM WebSphere MQ 7.0 je proud implementován jako téma nejvyšší úrovně, které lze administrativně namapovat na jiný identifikátor tématu.

Výchozí proud SYSTEM.BROKER.DEFAULT.STREAM je nastaven automaticky pro všechny zprostředkovatele a správce front v síti a žádná další konfigurace není vyžadována pro použití výchozího proudu. Myslete na výchozí proud jako nepojmenovaný výchozí prostor tématu. Témata, která byla publikována do výchozího proudu, jsou okamžitě k dispozici všem připojeným správcům front od IBM WebSphere MQ 7.0, kdy je povoleno publikování/odběr ve frontě. Pojmenované proudy jsou jako samostatné pojmenované prostory témat. Pojmenovaný proud musí být definován na každém zprostředkovateli, kde se používá.

Jsou-li vydavatelé a odběratelé v různých správcích front, potom jsou po připojení zprostředkovatelů ve stejné hierarchii zprostředkovatele, není vyžadována žádná další konfigurace pro publikování a odběry mezi nimi. Stejná interoperabilita funguje i v obráceném pořadí.

Pojmenované proudy

Návrhář řešení, který pracuje s programovacím modelem publikování/odběru ve frontě, může rozhodnout o umístění všech sportovních publikací do pojmenovaného proudu nazvaného `Sport`. V produktu IBM WebSphere MQ 6.0 se proud často replikuje automaticky na jiné zprostředkovatele, kteří používají modelovou frontu `SYSTEM.BROKER.MODEL.STREAM`. Aby však byl proud dostupný správci front spuštěnému v produktu IBM WebSphere MQ 7.0 s povoleným publikováním/odběrem ve frontě, musí být tento proud přidán ručně.

Aplikace typu publikování/odběr zařazené do fronty, které se přihlásí k odběru `Soccer/Results` v proudu `Sport`, se bez změny nemění. Integrované aplikace typu publikování/odběr, které se přihlašují k odběru tématu `Sport` pomocí produktu `MQSUBa` dodávají řetězec tématu `Soccer/Results`, obdrží také stejné publikace.

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

1. Pokračujete v vytváření aplikací publikování/odběru ve frontě spuštěných v pozdějších správcích front verze, nikoli migrace aplikací do rozhraní modulu `MQI` publikování/odběru.
2. Výchozí mapování proudů na témata vede k "kolizi" v prostoru témat a publikace v proudu mají stejný řetězec tématu jako publikace odjinud.

Oprávnění

Ve výchozím nastavení je v kořenovém adresáři stromu témat více objektů témat: `SYSTEM.BASE.TOPIC`, `SYSTEM.BROKER.DEFAULT.STREAMa` `SYSTEM.BROKER.DEFAULT.SUBPOINT`. Oprávnění (například pro publikování nebo odběr) jsou určována autoritami na serveru `SYSTEM.BASE.TOPIC`; veškerá oprávnění k produktu `SYSTEM.BROKER.DEFAULT.STREAM` nebo `SYSTEM.BROKER.DEFAULT.SUBPOINT` jsou ignorována. Pokud je některý z `SYSTEM.BROKER.DEFAULT.STREAM` nebo `SYSTEM.BROKER.DEFAULT.SUBPOINT` odstraněn a znovu vytvořen s neprázdným řetězcem tématu, oprávnění definovaná na těchto objektech se používají stejným způsobem jako normální objekt tématu.

Mapování mezi proudy a tématy

Proud publikování/odběru ve frontě je napodoben v produktu IBM WebSphere MQ 7.0 tím, že vytváří frontu a dává jí stejný název jako proud. Někdy se fronta nazývá fronta proudu, protože taková je, jak se objevuje ve frontě aplikací typu publikování/odběr. Fronta je identifikována pro stroj publikování/odběru přidáním do speciálního seznamu názvů nazvaného `SYSTEM.QPUBSUB.QUEUE.NAMELIST`. Do seznamu názvů můžete přidat tolik proudů, kolik jich potřebujete, přidáním dalších speciálních front do seznamu názvů. Nakonec je třeba přidat témata se stejnými názvy jako proudy a stejné řetězce témat jako název proudu, takže můžete publikovat a odebírat témata.

Avšak za výjimečných okolností můžete udělit témata, která odpovídají proudům, všechny řetězce témat, které zvolíte při definování témat. Účelem řetězce tématu je poskytnout tématu jedinečný název v prostoru tématu. Typicky název proudu slouží tomuto účelu dokonale. Někdy se může název proudu a existující název tématu kolidovat. Chcete-li tento problém vyřešit, vyberte jiný řetězec tématu pro téma přidružené k proudu. Vyberte libovolný řetězec tématu a ujistěte se, že je jedinečný.

Řetězec tématu, který je definován v definici tématu, je uveden jako předpona k řetězci tématu poskytovaným vydavateli a odběrateli pomocí volání modulu `MQOPEN` nebo `MQSUB MQI`. Aplikace odkazující na témata používající objekty témat nejsou ovlivněny volbou předpony řetězce témat-a proto můžete zvolit libovolný řetězec tématu, který uchovává publikování jedinečné v prostoru tématu.

Přemapování různých proudů na různá témata se spoléhá na předpony použité pro řetězce témat, které jsou jedinečné, aby se oddělují jedna sada témat zcela od jiného. Musíte definovat konvenci pojmenování univerzálních témat, která bude přísně dodržována, aby mapování fungovalo.

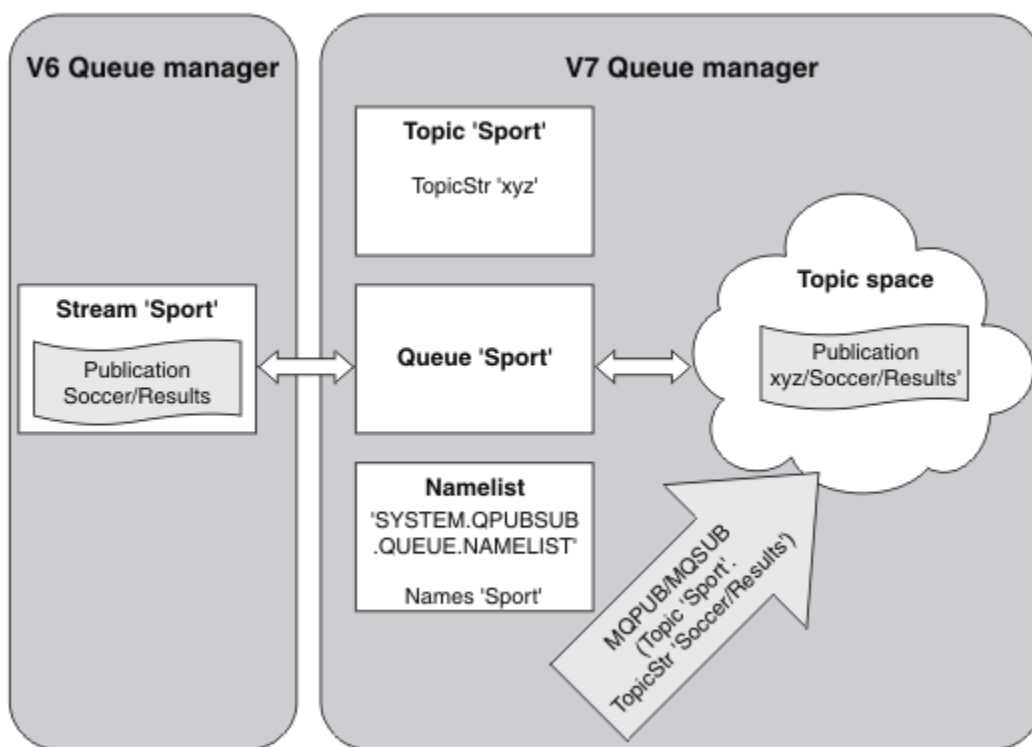
Pokud se v produktu IBM WebSphere MQ 7.0 kolidují řetězce témat, můžete použít proudy k oddělení prostorů témat.

Počínaje produktem IBM WebSphere MQ 7.0 je k přemapování řetězce tématu na jiné místo v prostoru tématu použit mechanismus předběžného určení.

Poznámka: Odstraní-li proud, odstraňte nejprve všechny odběry z proudu. Tato akce je nejdůležitější, pokud některý z odběrů pochází od jiných zprostředkovatelů v hierarchii zprostředkovatele.

Příklad

V produktu [Obrázek 31](#) na stránce 82 má téma 'Sport' řetězec tématu 'xyz', který má za následek, že publikace pocházející z proudu 'Sport' mají předponu řetězce 'xyz' v prostoru tématu správce front verze 7. Publikování nebo přihlášení k odběru verze 7 s předponami 'Sport' témat 'xyz' s řetězcem tématu. Pokud publikační tok přechází na odběratele produktu IBM WebSphere MQ 6, předpona 'xyz' je odebrána z publikace a je umístěna v proudu produktu 'Sport'. Naopak při publikování toků z produktu IBM WebSphere MQ 6 do produktu IBM WebSphere MQ 7z proudu 'Sport' na téma 'Sport' je předpona 'xyz' přidána do řetězce tématu.



Obrázek 31. Proudy IBM WebSphere MQ 6, které koexistují s tématy IBM WebSphere MQ 7

Body odběru a témata

Pojmenované body odběru jsou emulovány tématy a objekty témat.

Chcete-li přidat body odběru ručně, přečtěte si téma [Přidání bodu odběru](#).

Body odběru v produktu IBM MQ

Produkt IBM MQ mapuje body odběru na různé prostory témat ve stromu témat IBM MQ. Témata v příkazových zprávách bez bodu odběru se namapují nezměněny na kořen stromu témat IBM MQ a dědí vlastnosti z produktu SYSTEM.BASE.TOPIC.

Zprávy příkazů s bodem odběru se zpracovávají pomocí seznamu objektů témat v produktu SYSTEM.QPUBSUB.SUBPOINT.NAMELIST. Název bodu odběru ve zprávě příkazu je porovnán s řetězcem tématu pro každý objekt tématu v seznamu. Je-li nalezena shoda, je název bodu odběru předřazen do

řetězce tématu jako uzel tématu. Téma dědí své vlastnosti z přidruženého objektu tématu, který se nachází v produktu SYSTEM.QPUBSUB.SUBPOINT.NAMELIST.

Výsledkem použití bodů odběru je vytvoření samostatného prostoru tématu pro každý bod odběru. Prostor tématu je kořenem v tématu, které má stejný název jako bod odběru. Témata v každém prostoru tématu dědí vlastnosti z objektu tématu se stejným názvem, jako má bod odběru.

Jakékoli vlastnosti, které nejsou nastaveny ve vyhovujícím objektu tématu, jsou zděděny běžným způsobem z produktu SYSTEM.BASE.TOPIC.

Existující fronty publikování/odběru ve frontě pomocí záhlaví zpráv produktu MQRFH2 pokračují v práci nastavením vlastnosti **SubPoint** ve zprávách příkazu Publish nebo Register subscriber. Bod odběru je zkombinován s řetězcem tématu ve zprávě příkazu a výsledné téma je zpracováno stejně jako ostatní.

IBM WebSphere MQ 7.0nebo novější nejsou aplikace ovlivněny body odběru. Pokud aplikace používá téma, které dědí informace od jednoho z odpovídajících objektů témat, tato aplikace interoperauje s aplikací ve frontě s použitím odpovídajícího bodu odběru.

Příklad

Existující WebSphere Message Broker (nyní známé jako IBM Integration Bus) Aplikace publikování/ odběru, která byla migrována do produktu IBM MQ , vytvořila dva objekty témat, GBP a USD, s odpovídajícími řetězci témat ' GBP ' a ' USD ' .

Existující vydavatelé k tématu NYSE/IBM/SPOT, migrovaný ke spuštění na systému IBM MQ, který používá bod odběru USD , vytváří publikace na téma USD/NYSE/IBM/SPOT. Podobně existující odběratelé do produktu NYSE/IBM/SPOTs použitím bodu odběru USD vytvářejí odběry USD/NYSE/IBM/SPOT.

Přihlaste se k odběru ceny dolaru v produktu IBM WebSphere MQ 7.0nebo novějším, publish/subscribe program, a to voláním MQSUB. Vytvořte odběr pomocí objektu tématu USD a řetězce tématu ' NYSE/IBM/SPOT ' , jak je ilustrováno v části 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 témat USD a GBP na hostiteli témat klastru.
2. Odstraňte všechny kopie objektů tématu USD a GBP v ostatních správcích 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 pro jednu správce front

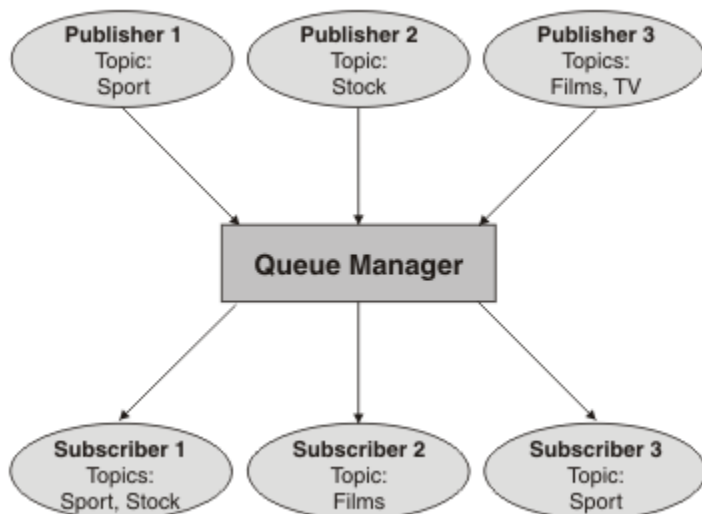
Obrázek 32 na stránce 84 ilustruje konfiguraci publikování/odběru základního jediného správce front. Příklad ukazuje konfiguraci služby news, kde jsou informace k dispozici od vydavatelů o několika tématech:

- Vydavatel 1 zveřejňuje informace o sportovních výsledcích pomocí tématu Sport
- Vydavatel 2 publikuje informace o cenách akcií za použití tématu Burzovní
- Vydavatel 3 zveřejňuje informace o filmových recenzích pomocí tématu Films, a o televizních filmech pomocí témat TV

Dva odběratelé zaregistrovali zájem o různá témata, takže správce front jim odešle informace, které se zajímají o:

- Odběratel 1 přijímá sportovní výsledky a ceny akcií
- Odběratel 2 obdrží filmové recenze
- Odběratel 3 přijímá sportovní výsledky

Žádný z účastníků nezaregistroval zájem o televizní výpisy, takže se nešíří.



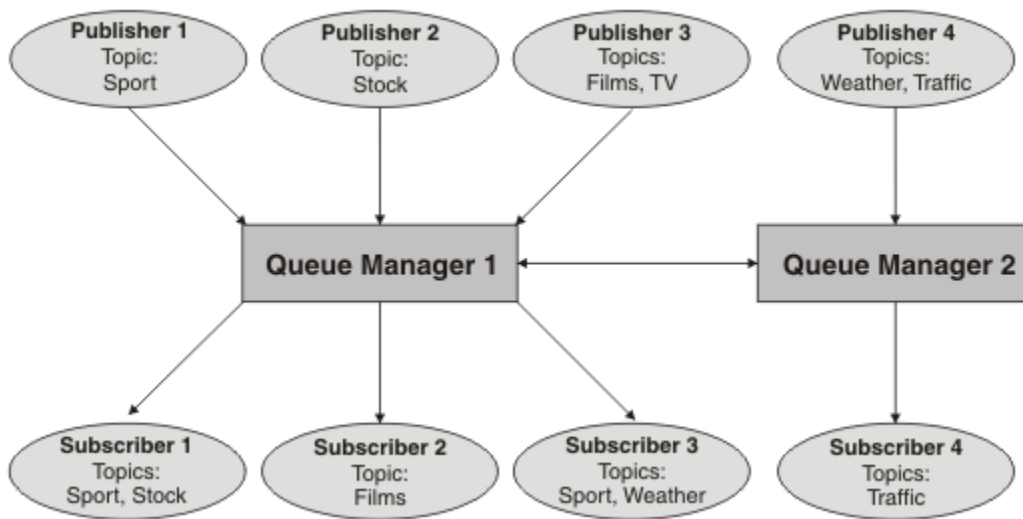
Obrázek 32. Příklad publikování/odběru jednoho správce front

Distribuované sítě typu publikování/odběr

Každý správce front odpovídá zprávám publikovaným v rámci tématu s lokálně vytvořenými odběry, které jsou k odběru daného tématu přihlášeni. Můžete konfigurovat síť správců front tak, aby zprávy publikované aplikací připojenými k jednomu správci front byly doručovány s odpovídajícími odběry vytvořenými v jiných správcích front v síti. To vyžaduje další konfiguraci u jednoduchých kanálů mezi správcí front.

Distribuovaná konfigurace publikování/odběru je sada správců front, které jsou vzájemně propojeny. Správci front se mohou všichni nacházet ve stejném fyzickém systému nebo mohou být distribuovány přes několik fyzických systémů. Při společném připojení správců front se mohou odběratelé přihlásit k odběru jednoho správce front a přijímat zprávy, které byly původně publikovány v jiném správci front. Pro ilustraci přidává následující obrázek druhého správce front do konfigurace popsané v tématu [“Příklad konfigurace publikování/odběru pro jednu správce front”](#) na stránce 83.

- Správce front 2 je používán produktem Publisher 4 k publikování informací o prognóze počasí pomocí tématu Počasí a informace o dopravních podmínkách na hlavních silnicích za použití tématu Provoz.
- Odběratel 4 také používá tohoto správce front a přihlašuje se k odběru informací o provozních podmínkách pomocí tématu Provoz.
- Odběratel 3 také odebírá informace o meteorologických podmínkách, i když používá jiného správce front od vydavatele. To je možné, protože jsou k sobě navzájem propojeny správce front.

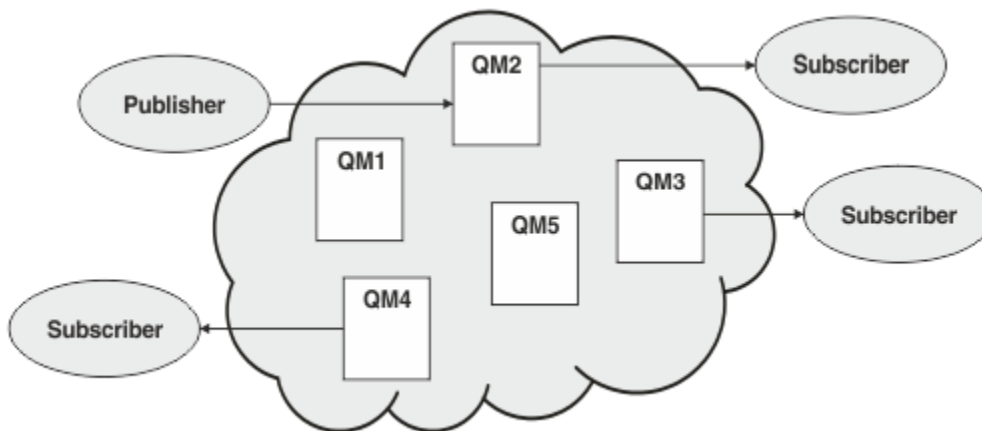


Obrázek 33. Příklad publikování/odběru se dvěma správci front

Správce front můžete ručně připojit v nadřazené a podřazené hierarchii, nebo můžete vytvořit klastr publikování/odběru a nechat prostor IBM MQ definovat velkou část podrobností o připojení. Můžete také použít obě topologie v kombinaci, například spojením několika klastrů dohromady v hierarchii.

Přehled klastrů publikování/odběru

Klastr typu publikování/odběr je standardním klastrem s jedním nebo více objekty tématu, které byly přidány do klastru. Definujete-li objekt administrativního tématu v libovolném správci front v klastru a tento objekt tématu zpřístupníte zadáním názvu klastru, budou vydavatelé a odběratelé daného tématu schopni se připojit k libovolnému správci front v klastru a publikované zprávy jsou směřovány na odběratele v rámci kanálů klastru mezi správci front.



Obrázek 34. klastr systému publikování/odběr

Existují dva způsoby, jak nakonfigurovat, jak jsou zprávy publikování/odběru směřovány do klastru:

- přímé směřování
- směřování hostitele témat

Konfigurujete-li klastrované téma s přímým směřováním, jsou zprávy publikované v rámci jednoho správce front odesílány přímo z tohoto správce front do všech odběrů v libovolném jiném správci front v daném klastru. To může poskytnout nejpřímější cestu k publikacím, ale má za následek, že všichni správci front v klastru budou informováni o všech ostatních správcích front, přičemž každý z nich může mít k dispozici kanály klastru, které jsou mezi nimi vytvořeny.

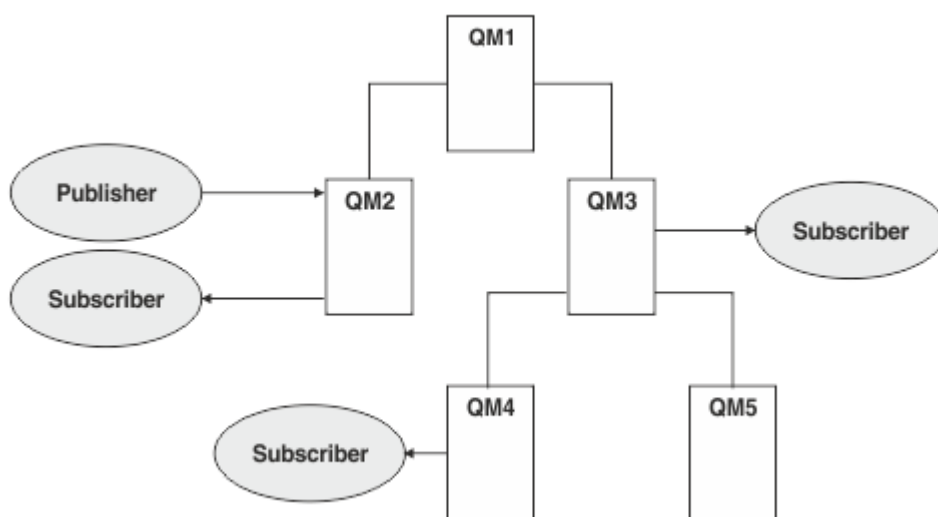
Při použití směrování hostitele témat jsou zprávy publikované v rámci jednoho správce front odesílány do správce front, který je hostitelem definice daného objektu tématu. Tento *správce front hostitele tématu* směřuje zprávu do každého odběru v každém dalším správci front v klastru. Pokud nejsou vydavatelé nebo odběratelé umístění na správci front hostitele tématu, bude to mít za následek delší trasu pro publikování. Výhodou je však to, že pouze správci front hostitele tématu se dozví o všech ostatních správčích front v klastru a mohou mít k sobě vytvořené kanály klastru.

Další informace viz [“Klastry publikování/odběru”](#) na stránce 87.

Přehled hierarchií publikování/odběru

Hierarchická hierarchie publikování/odběru je sada správců front propojených kanály do hierarchické struktury. Každý správce front identifikuje svého *nadřazeného* správce front, jak je popsáno v tématu [Připojení správce front k hierarchii publikování a odběru](#).

Vydavatelé a odběratelé k tématu se mohou připojit k libovolnému správci front v hierarchii a toky zpráv mezi nimi používají hierarchickou konektivitu správce front.



Obrázek 35. Hierarchie publikování/odběru

Na předchozím obrázku byly publikace doručené odběratelům na QM3 a QM4 směřovány z QM2 do QM1 a pak do QM3 a nakonec QM4.

Hierarchie vám dává přímou kontrolu nad vztahy mezi každým správcem front v hierarchii. To umožňuje přesné řízení směrování zpráv od vydavatelů k odběratelům a je obzvláště užitečné při směrování mezi sítěmi správců front s omezenou propojitelností. Měli byste pečlivě zvážit dostupnost a schopnost každého správce front, jehož prostřednictvím je zpráva směřována na její způsob od vydavatele k odběratelům.

Další informace viz [“Hierarchie publikování/odběru”](#) na stránce 89.

Distribuce publikací mezi správci front

Kromě voleb směrování jsou k dispozici dva přístupy k distribuci publikací v rámci sítě správců front:

- Zasílat pouze publikace z jednoho správce front do správců front, kteří jsou v současné době hostitelem odběru pro danou publikaci.
- Odešle každé publikování všem správcům front a umožní jejich porovnání s jejich odběry.

Dřívější výsledky jsou v publikačních zprávách odesílány pouze tam, kde je to nezbytné, ale vyžadují sdílení znalostí o odběru mezi správci front. Ta nevyžaduje sdílení znalostí o odběru, ale může vést ke zbytečným zprávám o publikování, které se posílají mezi správci front.

Produkt IBM MQ standardně používá původní metodu, v níž jsou publikace odesílány pouze správčům front, kteří mají odběry pro tyto účely. Znalost odběru je šířena mezi správci front ve formě *odběrů proxy*. Závisí na distribuci a životnosti odběrů a četnosti publikování, která je neefektivnější k použití v distribuované topologii publikování/odběru. Viz téma [Výkon odběru v sítích typu publikování/odběr](#).

Související pojmy

[“Stromy témat” na stránce 71](#)

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

Související úlohy

[Návrh klastrů publikování a odběru](#)

Související odkazy

[Scénáře hierarchie publikování/odběru](#)

Klastry publikování/odběru

Klaster publikování/odběru je standardním klastrem vzájemně propojených správců front, na kterém jsou publikace automaticky přesouvány z publikování aplikací do odběrů, které existují na některém správci front v klastru. Pro směrování publikací ve klastru publikování/odběru jsou k dispozici dvě volby: *přímé směrování* a *směrování hostitelů témat*. Výběr směrování závisí na velikosti a očekávaných vzorech aktivity pro váš klaster.

Klaster, který se používá pro systém zpráv publikování/odběru, se neliší od standardního klastru IBM MQ. V takovém případě mohou správci front v klastru publikování/odběru existovat na fyzicky oddělených počítačích a každý pár správců front je v případě potřeby automaticky připojen v rámci kanálů klastru. Další informace najdete v tématu [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 provést vytvoření tématu klastru, nakonfigurujte vlastnost **CLUSTER** s názvem klastru. Provedete-li to, všechny téma použité vydavatelem nebo odběratelem v tomto bodu nebo níže ve [stromu témat](#) budou sdíleny všemi správci front v klastru a zprávy publikované do klastrované větve stromu témat se automaticky směřují do odběrů u ostatních správců front v klastru.

Pouze jedna kopie každé zprávy je odeslána mezi správcem front vydavatele a každým z ostatních správců front, bez ohledu na počet odběratelů pro zprávu v cílovém správci front. Při příchodu do správce front s jedním nebo více odběry je zpráva duplikována přes všechny odběry.

Každý správce front, který se připojuje ke klastru, je automaticky informován o klastrovaných tématech a vydavatelé a odběratelé z tohoto správce front se automaticky účastní klastru.

Neklastrovaná aktivita publikování/odběru může být také umístěna v klastru publikování/odběru tím, že pracuje s řetězci tématu, které nespádají pod klastrovaný objekt tématu.

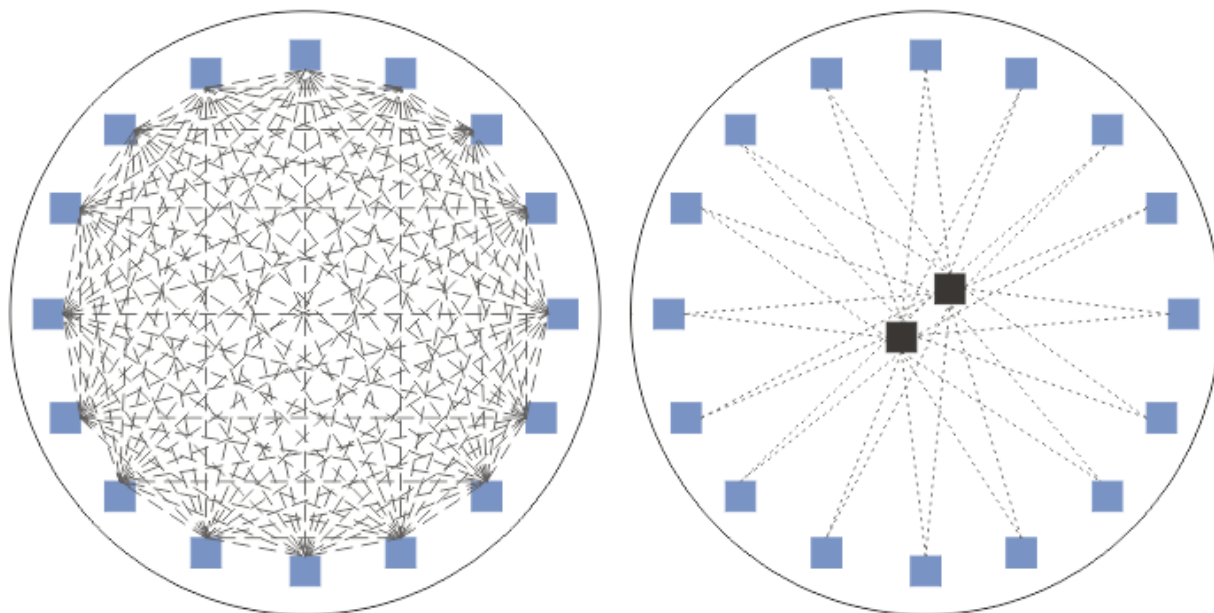
Pro směrování publikací ve klastru publikování/odběru jsou k dispozici dvě volby: *přímé směrování* a *směrování hostitelů témat*. Chcete-li zvolit směrování zpráv, které má být použito uvnitř klastru, nastavte vlastnost **CLROUTE** na administrovaném objektu tématu na jednu z následujících hodnot:

- **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 36. Přímé směrování a směrování hostitele témat

Přehled přímého směrování

Je-li spravovaný 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. Výběr správce front, na kterém je téma definováno, nemá vliv na chování systému zpráv publikování/odběru pro dané téma.

Každá zpráva směřuje přímo od správce front vydavatele ke každému odběru u ostatních správců front v klastru, nikoli prostředkujícími prostředkujícími správci front.

Při výchozím nastavení jsou zprávy odesílány pouze jiným správcům front v klastru, které jsou hostiteli jednoho nebo více odběrů.

- To závisí na každém správci front, který přímo informuje všechny ostatní správce front v klastru o všech tématech, která aktuálně mají k dispozici jeden nebo více odběrů. Výsledkem je skutečnost, že všichni správci front v klastru jsou si vědomi všech témat, na které je přihlášen odběr, a každý správce front, který je hostitelem odběru, který vytváří kanál ke všem ostatním správcům front. To je nezávislé na tom, zda má každý správce front vydavatele.
- Znalost jednotlivých odebíraných témat u všech správců front lze odebrat změnou na model odeslání všech publikací na všechny správce front v klastru bez ohledu na to, zda mají odběry. Tím se sníží znalost odběru informací o odběru, ale pravděpodobně se zvýší provoz publikování a počet kanálů, které každý správce front vytvoří. Viz téma [Výkon odběru v sítích typu publikování/odběr](#).

Toky zpráv publikování/odběru využívající přímo směrovaná klastrovaná témata se mohou rozprostírat do více klastrů publikování/odběru přidáním jednoho správce front z každého klastru do hierarchie publikování/odběru. Viz [Kombinování prostorů tématu s více klastry](#).

Podrobnější prozkoumání přímého směrování naleznete v tématu [Přímé směrování v klastrech publikování/odběru](#).

Přehled směrování hostitele témat

Je-li objekt spravovaného tématu konfigurován pro směrování hostitele témat, jsou publikováni ze správce front v klastru směrována prostřednictvím správce front, ve kterém je konfigurován objekt tématu ("hostitel tématu"), a odtud na správce front, ve kterém existují odběry.

- To závisí na každém správci front, který informuje všechny hostitele témat o každém tématu, které má v současné době k dispozici jeden nebo více odběrů. Každý správce front, který je hostitelem odběru, vytvoří kanál pro každého hostitele tématu pro téma, ke kterému se odběr vztahuje.
- Netéma hostování správců front není informované o jiných netématech, která jsou hostiteli správců front v klastru pro účely publikování a odběru, a nejsou mezi nimi vytvořeny kanály pro tento účel.
- Je-li aplikace publikování připojena ke správci front, který je hostitelem tématu, jsou publikované zprávy směrovány přímo na správce front, ve kterých byly vytvořeny odpovídající odběry, aniž by byl požadován další přechod 'přechod'. Podobně platí, že pokud jsou odpovídající odběry vytvořeny ve správci front, který je hostitelem daného tématu, jsou zprávy publikované v rámci daného tématu směrovány přímo do tohoto správce front bez nutnosti dalšího přechodu.
- Odběry ve stejném správci front jako vydavatel jsou uspokojeny bez prvního směrování publikací na hostitele objektu tématu.

Pokud jde o klastrované fronty, může více správců front konfigurovat stejný objekt tématu administrace. To poskytuje vyšší dostupnost směrování zpráv a vodorovné škálování prostřednictvím vyrovnávání pracovní zátěže. U objektů tématu routed routed v tématu platí, že když více správců front konfiguruje stejné pojmenované téma pro stejnou větev stromu témat, je každý hostitel tématu informován o odebíraných tématech každým správcem front, který je hostitelem odběru.

- Je-li publikována zpráva, odešle se do jednoho z správců front hostitele tématu, aby se přesměroval na odběr hostujících správců front. Volba správce front hostitele tématu postupuje podle stejných výchozích pravidel pro vyrovnávání pracovní zátěže, jako u klastrovaných front-point-to-point.
- Pokud jednoho nebo více správců front hostitele tématu nelze kontaktovat správcem front publikování, jsou zprávy směrovány do zbývajících dostupných témat hostujících správců front.

Každá publikace ve směrované větvi stromu témat se předá jednomu z hostitelů témat, a to i v případě, že v daném klastru neexistují žádné odběry daného tématu. Při výchozím nastavení jsou zprávy odesílány pouze z jiných správců front v klastru, které jsou hostiteli jednoho nebo více odběrů.

- To závisí na tom, že každý správce front hostitele je informován o všech odebíraných řetězcích témat v každém správci front v daném klastru.
- Znalost jednotlivých odebíraných témat lze odebrat změnou na model odesílání všech publikací směrovaných na hostitele tématu pro všechny správce front v daném klastru bez ohledu na to, zda mají odběry. Tím se sníží znalost odběru informací o odběru, ale pravděpodobně se zvýší provoz publikací a potenciálně i počet kanálů zavedených s každým tématem hostujícím správcem front. Viz téma [Výkon odběru v sítích typu publikování/odběr](#).

Toky zpráv publikování/odběru používající klastrovaná témata směrovaná z hostitele témat **ne mohou** zasahovat do více klastrů publikování/odběru prostřednictvím hierarchie publikování/odběru.

Podrobnější prozkoumání týkající se směrování hostitele témat naleznete v tématu [Směrování hostitele tématu v klastrech publikování/odběru](#).

Hierarchie publikování/odběru

Hierarchii publikování/odběru sestavíte tak, že propojíte správce front společně s použitím kanálů, a poté definujete vztah mezi dvojicemi správců front a jejich definicí mezi dvojicemi správců front. Toky zpráv od vydavatele k odběrům prostřednictvím přímých vztahů v hierarchii. Všimněte si, že to může znamenat, že se tam dostane více "přechodů" .

Pouze jedna kopie zprávy je odeslána mezi libovolnými dvojicemi správců front, bez ohledu na počet odběratelů pro zprávu v cílovém správci front. Při příchodu do správce front s jedním nebo více odběry je zpráva duplikována přes všechny odběry.

Ve výchozím nastavení jsou zprávy odesílány pouze jiným správcům front v hierarchii, které jsou na trase k odběru v jiném správci front:

- To závisí na každém správci front, který informuje každý přímý vztah o všech tématech, která aktuálně mají k dispozici jeden nebo více odběrů, a to buď na tomto správci front, nebo na jednom z jeho dalších vztahů. Výsledkem je, že všichni správci front v hierarchii jsou si vědomi všech témat, k jejichž odběru jsou přihlášení.
- Toto chování lze změnit tak, aby vždy odesílal publikace na všechny správce front v hierarchii bez ohledu na existující odběry. Tím dojde k odstranění potřeby šíření informací o odběru v rámci hierarchie, ale může zvýšit provoz publikování.

Při vytváření klastru je třeba dbát na to, abyste nevytvářeli cyklus způsobující, že zprávy budou v rámci sítě cyklického cyklu pokračovat. Žádné takové smyčky nelze vytvořit v hierarchii.

Každý správce front musí mít jedinečný název správce front.

Toky zpráv publikování/odběru mohou přesahovat více klastrů publikování/odběru. Chcete-li to provést, 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 typu publikování-odběr

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 je možné propojit do klastru publikování/odběru nebo do hierarchie publikování/odběru. Průběh odběru proxy mezi připojenými správci front. Odběry proxy způsobí, že publikace k tématu vytvořenému vydavatelem připojeným k jednomu správci front obdrží odběratelé na toto téma připojené k jiným správcům front. Viz [“Distribované síť typu publikování/odběr”](#) na stránce 84.

V topologiích publikování/odběru s mnoha tisíci odběrů na jednotlivé řetězce témat nebo v případech, kdy by mohla být existence těchto odběrů rychle změněna, je třeba zvážit režii šíření odběru proxy. Kromě automatické agregace popsané ve zbytku tohoto tématu můžete provést ruční změny konfigurace, které dále omezují tok proxy odběrů a publikování mezi propojenými správci front a které omezují latenci čekání na to, aby byl proxy odběr šířen do všech připojených správců front. Viz téma [Výkon odběru v sítích typu publikování/odběr](#).

Odběry proxy neobsahují žádné selektory používané lokálními odběry a řetězce témat odběru, které obsahují zástupné znaky, mohou být zjednodušeny. To může vést k publikování proxy odběrů v případě, že skutečné odběry nemají, což povede k dalšímu toku publikování mezi správci front. Správce front, který je hostitelem odběrů, filtruje tyto rozdíly, aby se nevrátily žádné další publikace k odběrům.

Agregace odběru proxy

Odběry proxy jsou agregovány pomocí duplicitního eliminačního systému. Pro konkrétní vyřešený řetězec tématu je odeslán proxy odběr na prvním lokálním odběru nebo na přijatém proxy odběru. Další odběry stejného řetězce tématu budou používat tento existující odběr serveru proxy.

Odběr proxy je zrušen po zrušení posledního odběru lokálního odběru nebo přijetí proxy odběru.

Agregace publikací

Je-li ve správci front více než jeden odběr ze stejného řetězce tématu, bude z jiných správců front v topologii publikování/odběru odeslán z jiných správců front pouze jedna kopie každé publikace odpovídající tomuto řetězci tématu. Při přijetí zprávy lokální správce front doručí kopii zprávy každému odpovídajícímu odběru.

Je možné, aby více než jeden odběr proxy odpovídal řetězci tématu v rámci jediné publikace, když proxy odběry obsahují zástupné znaky. Je-li zpráva publikována ve správci front, který odpovídá dvěma nebo více odběrům proxy vytvořeným pomocí jednoho připojeného správce front, předá se vzdálenému správci front pouze jedna kopie publikování, aby bylo možné splnit více odběrů proxy.

Související úlohy

[Detekce smyčky v distribuované síti publikování/odběr](#)

Zástupné znaky v odběrech proxy

Odběry mohou používat zástupné znaky v řetězcích témat, aby odpovídaly více řetězcům témat v publikacích.

Existují dvě schémata zástupných znaků, které může odběr použít: *na základě témat* a *na základě znaků*. Viz [“Režimy zástupných znaků”](#) na stránce 65.

V produktu IBM WebSphere MQ 7.0 a v novějších verzích jsou všechny odběry proxy pro odběry zástupných znaků převedeny na používání zástupných znaků založených na tématu. Je-li nalezen zástupný znak pro více znaků, bude nahrazen znakem # , a to zpět na nejbližší / . Například /aaa/bbb/c*d se převede na /aaa/bbb/#. Výsledky převodu ve vzdálených správcích front posílají o něco více publikací, než bylo explicitně upsána k odběru. Další publikace jsou odfiltrovány lokálním správcem front, když doručují publikace svým lokálním odběratelům.

Řízení použití zástupného znaku s vlastností WILDCARD

Pomocí vlastnosti MQSC **Topic WILDCARD** nebo ekvivalentních vlastností PCF Topic **WildcardOperation** můžete řídit doručování příruček aplikacím, které používají řetězce názvů témat zástupných znaků. Vlastnost WILDCARD může mít jednu ze dvou možných hodnot:

WILDCARD

Chování odběrů používajících zástupné znaky s ohledem na toto téma.

PASSTHRU

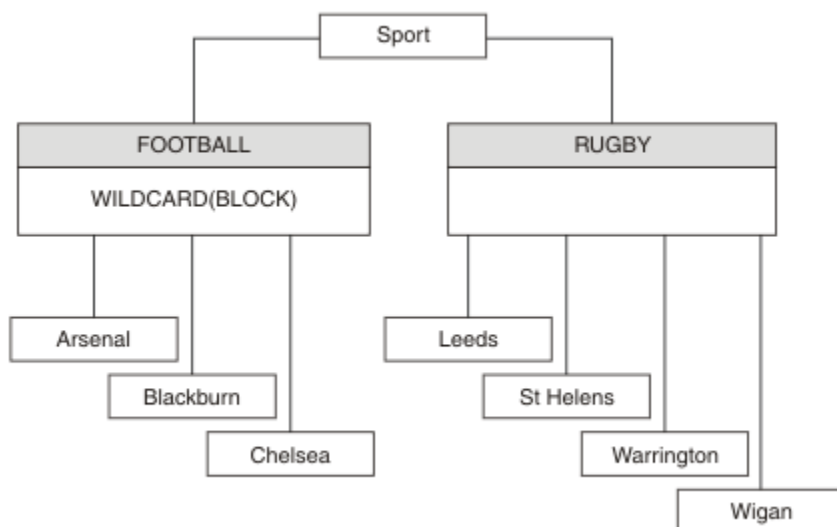
U odběrů registrovaných pro téma, které používá zástupné znaky a které je méně specifické než řetězec tématu v tomto objektu tématu, jsou poskytnuty publikace k tomuto tématu a k řetězcům tématu, které jsou specifitější než toto téma.

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 specifitější 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 pro publikování/odběr produktu Sport”](#) na stránce 78, můžete postupovat podle kroků k vytvoření stromové struktury témat zobrazené v [Obrázek 23](#) na stránce 74.



Obrázek 37. Strom témat, který používá vlastnost WILDCARD , BLOCK .

Odběratel používající řetězec tématu s zástupnými znaky # přijímá všechny publikace k tématu Sport a podstromu Sport /Rugby . Odběratel neobdrží žádné publikace do podstromu Sport/Football , protože hodnota vlastnosti WILDCARD u tématu Sport/Football je BLOCK.

Výchozí nastavení je PASSTHRU . Hodnotu vlastnosti ZÁSTUPNÝ ZNAK PASSTHRU můžete nastavit na uzly ve stromu Sport . Pokud uzly nemají hodnotu vlastnosti WILDCARD BLOCK, nastavení PASSTHRU nezmění chování pozorované odběratelem na uzlech ve stromu Sports .

V tomto příkladu vytvořte odběry a zjistěte, jak nastavení zástupných znaků ovlivňuje doručené publikace, viz Obrázek 27 na stránce 79. Spusťte publikační příkaz v produktu Obrázek 30 na stránce 80 a vytvořte některé publikace.

pub QMA

Obrázek 38. Publikovat na QMA

Výsledky jsou zobrazeny v Tabulka 3 na stránce 75. Všimněte si, jak nastavení hodnoty vlastnosti WILDCARD BLOCK brání odběrům se zástupnými znaky z příjmu publikací na témata v rámci rozsahu zástupného znaku.

Tabulka 6. Publikace přijaté v systému QMA			
Předplatně	Řetězec tématu	Přijaté publikace	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace ve fotbalovém podstromu blokováné uživatelem WILDCARD (BLOCK) v systému Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) v Sports/Football zabráňuje odběru zástupného znaku na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Předvolba WILDCARD na Sports/Rugby nezabrání odběru zástupného znaku na Leeds.

Poznámka:

Předpokládejme, že odběr má zástupný znak, který odpovídá objektu tématu s hodnotou vlastnosti WILDCARD BLOCK. Má-li odběr také řetězec tématu napravo od odpovídajícího zástupného znaku, odběr nikdy neobdrží publikování. Sada publikací, které nejsou blokovány, jsou publikace k tématům, která jsou nadřazenými prvky blokováného zástupného znaku. Publikování do témat, která jsou podřízenými prvky tématu s hodnotou vlastnosti produktu BLOCK, jsou blokovány zástupnými znaky. Proto řetězce témat odběru, které obsahují téma napravo od zástupného znaku, nikdy nepřijímají žádné publikace, které by se shodovaly.

Nastavení hodnoty 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 danému tématu s objektem tématu, který má hodnotu vlastnosti WILDCARD BLOCK. Používá zástupné znaky pro témata, která jsou nadřazenými nebo podřízenými prvky tématu s hodnotou vlastnosti WILDCARD BLOCK. V příkladu v produktu Obrázek 23 na stránce 74 může publikování, jako je 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 ve správci front. Proxy odběr je vytvořen ve všech ostatních správci front v klastru. Odběry pomocí řetězců témat obsahujících zástupné znaky, kombinované s tématy klastru, mohou být obtížné předpovědět chování. Chování je vysvětleno v následujícím příkladu.

V klastru nastavovaný pro tento příklad, "Příklad: Vytvoření klastru pro publikování/odběr produktu Sport" na stránce 78, má QMB stejnou sadu odběrů jako QMA, ale QMB nepřijala žádné publikace poté, co vydavatel publikován na QMA, viz Obrázek 24 na stránce 75. Ačkoli témata Sports/Football a Sports/Rugby jsou témata klastru, odběry definované v souboru fullsubs.tst se neodkazují na téma klastru. Z produktu QMB do produktu QMA nejsou šířeny žádné odběry proxy. Bez odběrů proxy jsou do produktu QMB předávány žádné publikace QMA.

Některé odběry, jako např. Sports/#/Leeds, se mohou v tomto případě odkazovat na téma klastru Sports/Rugby. Odběr produktu Sports/#/Leeds je ve skutečnosti interpretováno jako objekt tématu SYSTEM.BASE.TOPIC.

Pravidlo pro vyřešení objektu tématu, na který se odkazuje odběr, jako je například Sports/#/Leeds, je následující. Ořízněte řetězec tématu na první zástupný znak. Skenování bylo ponecháno přes řetězec tématu, který hledá první téma, které má přidružený objekt administrativního tématu. Objekt tématu může určovat název klastru, nebo definovat lokální objekt tématu. V příkladu, Sports/#/Leeds, řetězec tématu po oseknutí je Sports, který nemá žádný objekt tématu, a tak se Sports/#/Leeds dědí od SYSTEM.BASE.TOPIC, což je lokální objekt tématu.

Chcete-li zjistit, jak může přihlášení k odběru klastrovaných témat změnit způsob, jakým šíření zástupných znaků funguje, spusťte dávkový skript upsubs.bat. Skript vymaže fronty odběru a přidá odběry témat klastru do adresáře fullsubs.tst. Znovu spusťte soubor puba.bat a vytvořte dávku publikací; viz Obrázek 24 na stránce 75.

Produkt Tabulka 4 na stránce 76 zobrazuje výsledek přidání dvou nových odběrů do stejného správce front, ve kterém byly publikovány publikace. Výsledkem je podle očekávání, že nové odběry obdrží jedno publikování a počty publikování přijatých ostatními odběry budou nezměněny. Dojde k neočekávaným výsledkům na druhém správci front klastru, viz Tabulka 5 na stránce 77.

Tabulka 7. Publikace přijaté v systému QMA			
Předplatné	Řetězec tématu	Přijaté publikace	Notes
SPORTS	Sports/#	Sports Sports/Rugby Sports/Rugby/Leeds	Všechny publikace ve fotbalovém podstromu blokováné uživatelem WILDCARD (BLOCK) v systému Sports/Football

Předplatně	Řetězec tématu	Přijaté publikace	Notes
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) v Sports/ Football zabraňuje odběru zástupného znaku na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/Leeds	Předvolba WILDCARD na Sports/ Rugby nezabrání odběru zástupného znaku na Leeds.
FARSENAL	Sports/Football/ Arsenal	Sports/Football/ Arsenal	Produkt Arsenal obdrží publikování, protože odběr nemá zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikování v každém případě.

Produkt Tabulka 5 na stránce 77 zobrazuje výsledky přidání dvou nových odběrů v produktu QMB a publikování v produktu QMA. Připomeňme si, že produkt QMB nepřijal žádná publikování bez těchto dvou nových odběrů. Jak bylo očekáváno, tyto dva nové odběry přijímají publikace, protože Sports/FootBall a Sports/Rugby jsou obě témata klastru. QMB postoupil proxy odběry pro Sports/Football/Arsenal a Sports/Rugby/Leeds do QMA, které následně odeslaly publikace do QMB.

Neočekávaný výsledek je takový, že se dva odběry Sports/# a Sports/#/Leeds, které dříve neobdržely žádné publikace, nyní přijímají publikace. Důvodem je to, 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. V důsledku toho se odběry lokálních témat Sports/# a Sports/#/Leeds přijímají v publikaci Sports/Rugby/Leeds. Produkt Sports/#/Arsenal nadále neobdrží publikování, protože sport/fotbal má nastavenou hodnotu vlastnosti WILDCARD nastavenou na hodnotu BLOCK.

Předplatně	Řetězec tématu	Přijaté publikace	Notes
SPORTS	Sports/#	Sports/Rugby/ Leeds	Všechny publikace ve fotbalovém podstromu blokovány produktem WILDCARD (BLOCK) v systému Sports/Football
SARSENAL	Sports/#/Arsenal	-	WILDCARD (BLOCK) v Sports/ Football zabraňuje odběru zástupného znaku na Arsenal
SLEEDS	Sports/#/Leeds	Sports/Rugby/ Leeds	Výchozí hodnota WILDCARD na Sports/Rugby nezabrání odběru zástupných znaků v systému Leeds.
FARSENAL	Sports/Football/ Arsenal	Sports/Football/ Arsenal	Produkt Arsenal obdrží publikování, protože odběr nemá zástupný znak.
FLEEDS	Sports/Rugby/Leeds	Sports/Rugby/Leeds	Leeds obdrží publikování v každém případě.

Ve většině aplikací je nežádoucí pro jeden odběr ovlivnit chování jiného odběru. Jedno důležité použití vlastnosti WILDCARD s hodnotou BLOCK znamená, že se odběry ke stejnému řetězci tématu, které obsahují zástupné znaky, budou chovat jednotně. Údaj o tom, zda je odběr ve stejném správcí front jako vydavatel nebo jiný, výsledky odběru jsou stejné.

Zástupné znaky a proudy

Pro novou aplikaci napsanou na rozhraní API pro publikování/odběr je výsledkem, že odběr z produktu * neobdrží žádné publikace. Chcete-li přijmout všechny sportovní publikace, musíte se přihlásit k odběru produktu Sports/*nebo produktu Sports/#a podobně jako publikace Business .

Chování existující aplikace publikování/odběru ve frontě se nezmění, pokud je zprostředkovatel publikování/odběru migrován do produktu IBM WebSphere MQ 7 a novějších verzí. Vlastnost **StreamName** v příkazech **Publish**, **Register Publisher** nebo **Subscriber** je mapována na název tématu, na který byl proud migrován.

Zástupné znaky a body odběru

U nové aplikace napsané pro rozhraní API pro publikování/odběr je výsledkem migrace skutečnost, že odběr produktu * přijímá žádné publikace. Chcete-li přijmout všechny sportovní publikace, musíte se přihlásit k odběru produktu Sports/*nebo produktu Sports/#a podobně jako publikace Business .

Chování existující aplikace publikování/odběru ve frontě se nezmění, pokud je zprostředkovatel publikování/odběru migrován do produktu IBM WebSphere MQ 7 a novějších verzí. Vlastnost **SubPoint** v příkazech **Publish**, **Register Publisher** nebo **Subscriber** je mapována na název tématu, na které byl odběr migrován.

Příklad: Vytvoření klastru pro publikování/odběr produktu Sport

Kroky následující po vytvoření klastru CL1 se čtyřmi správci front: dvě úplná úložiště, CL1A a CL1Ba dvě dílčí úložiště, QMA a QMB. Úplná úložiště se používají k zadržení pouze definic klastru. QMA je označen jako hostitel tématu klastru. Trvalé odběry jsou definovány na serveru QMA i v produktu QMB.

Poznámka: Příklad je kódován pro Windows. Chcete-li nakonfigurovat a otestovat tento příklad na jiných platformách, musíte soubor [Create qmgrs.bat](#) a [create pub.bat](#) nakonfigurovat a otestovat.

1. Vytvořte skriptové soubory.
 - a. [Vytvoření topics.tst](#)
 - b. [Vytvořit wildsubs.tst](#)
 - c. [Vytvořit fullsubs.tst](#)
 - d. [Vytvořit qmgrs.bat](#)
 - e. [create pub.bat](#)
2. Chcete-li vytvořit konfiguraci, spusťte příkaz [Create qmgrs.bat](#) .

```
qmgrs
```

Vytvořte témata v produktu [Obrázek 23 na stránce 74](#). Skript na obrázku 5 vytvoří témata klastru Sports/Football a Sports/Rugby.

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

```

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 39. Odstraňte a vytvořte témata: topics.tst

Poznámka: Odstraňte témata, protože produkt REPLACE nebude nahrazovat řetězce témat.

Vytvoření odběrů se zástupnými znaky. Zástupné znaky odpovídají tématům s objekty témat v produktu Obrázek 23 na stránce 74. Vytvořte frontu pro každý odběr. Fronty jsou vymazány a odběry odstraněny, když je skript spuštěn nebo znovu spuštěn.

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

```

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 40. Vytvoření zástupných znaků pro zástupný znak: wildsubs.tst

Vytvořte odběry, které odkazují na objekty tématu klastru.

Poznámka:

Oddělovač, /, se automaticky vkládá mezi řetězec tématu, na který odkazuje TOPICOBJ, a řetězec tématu definovaný parametrem TOPICSTR.

Definice DEFINE SUB(FARSENAL) TOPICSTR('Sports/Football/Arsenal') DEST(QFARSENAL) vytváří stejný odběr. TOPICOBJ se používá jako rychlý způsob, jak odkazovat na řetězec tématu, který jste již definovali. Když se odběr vytvoří, již se nebude odkazovat 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 41. Odstraňte a vytvořte odběry: *fullsubs.tst*

Vytvořte klastr se dvěma úložišti. Vytvořte dvě dílčí úložiště pro publikování a odběr. Znovu spusťte skript k odstranění všeho a začněte znovu. Skript také vytvoří hierarchii témat a počáteční zástupné znaky odběrů.

Poznámka:

Na ostatních platformách napište podobný skript nebo zadejte všechny příkazy. Pomocí skriptu lze rychle odstranit vše a začít znovu se stejnou 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 42. Vytvoření správců 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 43. Aktualizovat odběry: *upsubs.bat*

Spuštěním příkazu `pub.bat` se správcem front jako parametrem publikujete zprávy obsahující řetězec tématu publikování. Produkt `Pub.bat` používá vzorový 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 44. Publikovat: pub.bat

Související pojmy

Použití zástupných znaků a zachovaných publikování

Obor publikování

Při konfiguraci klastru/klasteru publikování/odběru dále platí, že oblast rozsahu publikování dále řídí, zda správci front předají publikování vzdáleným správcům front. K administraci oboru publikování použijte atribut tématu **PUBSCOPE**.

Pokud není publikace předána ke vzdáleným správcům front, obdrží publikování pouze lokální odběratelé.

Při použití klastru publikování/odběru je obor publikování primárně řízen definicí objektů tématu klastru v určitých bodech ve stromu témat. Obor publikování musí být nastaven tak, aby povoloval tok publikací ostatním správcům front v klastru. Obor publikování pro klastrované téma byste měli omezit pouze v případě, že potřebujete řízení určitých témat u určitých správců front s možností dokončování na úrovni vlákn.

Když použijete hierarchii publikování/odběru, rozsah publikací je primárně řízen tímto atributem v kombinaci s atributem Rozsah odběru.

Atribut **PUBSCOPE** se používá k určení rozsahu publikací vytvořených pro určité téma. Atribut můžete nastavit na jednu z následujících hodnot:

QMGR

Publikování je doručeno pouze lokálním odběratelům. Tyto příručky se nazývají *lokální publikace*. Lokální publikace nejsou předávány vzdáleným správcům front, a proto nejsou odběratelé připojeni ke vzdáleným správcům front připojeni.

ALL

Publikování je doručeno lokálním odběratelům a odběratelům připojeným ke vzdáleným správcům front v klastru nebo v rámci hierarchie publikování/odběru. Tyto příručky 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 vložení zprávy produktu MQPMO_SCOPE_QMGR. Je-li tato volba použita, přepíše jakékoli chování nastavené pomocí atributu tématu **PUBSCOPE**.

Související pojmy

“Objekty administrativního tématu” na stránce 72

Pomocí objektu administrativního tématu můžete jednotlivým tématům přiřazovat specifické, nevýchozí atributy.

Související úlohy

Konfigurace sítí distribuovaných publikování/odběru

Obor odběru

Rozsah odběru řídí, zda odběr v jednom správci front přijímá publikování publikovaná v jiném správci front v klastru nebo v rámci hierarchie publikování/odběru nebo pouze publikování z lokálních vydavatelů.

Omezení rozsahu odběru na správce front zastaví předávání odběrů serverů proxy jiným správcům front v topologii publikování/odběru. Tím se sníží provoz přenosu zpráv mezi správci front v rámci správce front nebo odběru.

Při použití klastru publikování/odběru je rozsah odběrů primárně řízen definicí objektů tématu s klastru v určitých bodech ve stromu témat. Rozsah odběru musí být nastaven tak, aby byl povolen tok proxy odběrů pro ostatní správce front v klastru. Rozsah odběru pro klastrované téma byste měli omezit pouze v případě, že potřebujete řízení změn specifických témat u některých správců front.

Když použijete hierarchii publikování/odběru, rozsah odběrů je primárně řízen tímto atributem v kombinaci s atributem [rozsah publikace](#).

Atribut tématu **SUBSCOPE** se používá k určení rozsahu odběrů provedených u specifického tématu. Atribut můžete nastavit na jednu z následujících hodnot:

QMGR

Odběr přijímá pouze lokální publikování a proxy odběry se nešíří do vzdálených správců front.

ALL

Odběr proxy je šířen do vzdálených správců front v klastru nebo hierarchii publikování/odběru, a odběratel přijímá lokální a vzdálené publikování.

AsParent

Použijte nastavení **SUBSCOPE** nadřazeného tématu ve stromu témat.

Je-li obor odběru pro téma nastaven na hodnotu ALL, buď přímo, nebo pomocí volby ASPARENT, mohou jednotlivé odběry daného tématu omezit svůj rozsah na QMGR zadáním MQSO_SCOPE_QMGR při vytváření odběru. Odběr tématu, který má rozsah QMGR, nemůže rozšířit rozsah na ALL.

Související pojmy

[“Objekty administrativního tématu” na stránce 72](#)

Pomocí objektu administrativního tématu můžete jednotlivým tématům přiřazovat specifické, nevýchozí atributy.

Související úlohy

[Konfigurace sítí distribuovaných publikování/odběru](#)

Prostory tématu

Prostor tématu je sada témat, na kterých můžete odebírat a publikovat. Správce front v distribuované topologii publikování/odběru má prostor tématu, který potenciálně obsahuje témata, která byla přihlášena k odběru a která byla publikována v připojených správcích front v dané topologii.

Poznámka: Přehled témat v rámci správce front, jako jsou například administrativní objekty témat, řetězce témat a stromy témat, viz [“Témata” na stránce 63](#). Další odkazy na *témata* v aktuálním článku odkazují na *řetězce témat*, není-li uvedeno jinak.

Témata se na počátku vytvoří jedním z následujících způsobů:

- administrativně, pokud definujete objekt tématu nebo trvalý odběr.
- dynamicky, pokud aplikace vytvoří publikování nebo odběr dynamicky na nové téma.

Témata jsou šířena do dalších správců front pomocí proxy odběrů a vytvářením objektů témat administrativního klastru. Proxy odběry mají za následek postoupení publikací od správce front, ke kterému je vydavatel připojen, ke správcům front odběratelů.

Odběry proxy se šíří mezi všemi správci front, kteří jsou připojeni k relacím nadřazený-podřazený v hierarchii správce front. Výsledkem je, že se můžete přihlásit k odběru jednoho správce front s použitím tématu definovaného v libovolném jiném správci front v hierarchii. Dokud existuje propojená cesta mezi správci front, nezáleží na tom, jakým způsobem jsou správci front připojeni.

Odběry proxy se také šíří pro odběry témat klastru v klastru publikování/odběr. Téma klastru je téma, které je připojeno k objektu tématu, který má atribut **CLUSTER**, nebo který dědí atribut z jeho nadřazené položky. Témata, která nejsou témata klastru, jsou známa jako lokální témata a nereplikují se do klastru. Do klastru z odběrů na lokální témata nejsou šířeny žádné odběry proxy.

Chcete-li shrnout, proxy odběry jsou vytvářeny pro odběratele za dvou okolností.

1. Správce front je členem hierarchie a proxy odběr je předáván nadřízenému a podřízenému objektu správce front.
2. Správce front je členem klastru a řetězec tématu odběru se přeloží na téma, které je přidruženo k objektu tématu klastru. Je-li téma tématem klastru *přímé směřování*, proxy odběry jsou předány všem členům klastru. Je-li téma tématem klastru *téma směřované na hostitele témat*, proxy odběry jsou předávány pouze správcům front v klastru, který definoval objekt tématu klastru. Další informace viz [“Klastry publikování/odběru”](#) na stránce 87.

Je-li správce front členem klastru a hierarchie, proxy odběry jsou šířeny oběma mechanismy, aniž by doručíte duplicitní publikace odběrateli.

Témata se třemi topologiemi publikování/odběru jsou popsány v následujícím seznamu:

- [“Případ 1. Klastry publikování/odběru”](#) na stránce 100.
- [“Případ 2. Hierarchie publikování/odběru ve verzi 7 nebo novější”](#) na stránce 101.
- [“Případ 3. Hierarchie publikování/odběru a proudy v produktu IBM WebSphere MQ 6”](#) na stránce 101.

V samostatných tématech popisují následující konfigurační úlohy, jak kombinovat prostory témat.

- [Vytvoření jediného prostoru tématu v klastru publikování/odběru.](#)
- [Přidání správce front produktu IBM WebSphere MQ 7 nebo pozdějšího do existujících prostorů témat IBM WebSphere MQ 6.](#)
- [Sloučení prostorů tématu s více klastry.](#)
- [Kombinování a izolování prostorů témat ve více klastrech.](#)
- [Publikování a přihlášení k odběru prostorů témat ve více klastrech.](#)

Případ 1. Klastry publikování/odběru

V tomto příkladu předpokládáme, že správce front není připojen k hierarchii publikování/odběru.

Je-li správce front členem klastru publikování/odběru, je jeho prostor tématu vytvořen z lokálních témat a témat klastru. Lokální témata jsou přidružena k objektům tématu bez atributu **CLUSTER**. Pokud má správce front definice objektů lokálního tématu, 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 správcí front, pokud se nezahlásíte, že se přihlásíte k odběru objektu tématu klastru.

Jsou-li vyžadovány stejné pojmenované definice objektu tématu klastru ve více správcích front, například při použití *směřování hostitele témat*, je důležité, aby všechny definice odpovídaly v případě potřeby. Další informace naleznete v tématu [Vytvoření jediného prostoru tématu v klastru publikování/odběru](#).

Lokální definice objektu tématu, ať už definice je pro téma klastru nebo lokální téma, má přednost před stejným objektem tématu definovaným jinde v klastru. Používá se lokálně definované téma, a to i v případě, že je objekt definovaný na jiném místě novější.

Je důležité, aby objekt tématu klastru byl přidružen ke stejnému tématu kdekoli v klastru. Řetězec tématu, ke kterému je přidružen objekt tématu, nelze změnit. Chcete-li přidružit stejný objekt tématu k jinému řetězci tématu, je třeba objekt tématu odstranit a znovu jej vytvořit s použitím nového řetězce tématu. Je-li téma klastrováno, je výsledkem odstranění kopií objektu tématu uloženého na ostatních členech klastru a následné vytvoření kopií nového objektu tématu kdekoli v klastru. Kopie objektu tématu se všechny odkazují na stejný řetězec tématu.

Je možné náhodně vytvořit dvě definice stejného objektu tématu s názvem na různých správcích front v klastru s různými řetězci tématu. Výsledkem může být matoucí chování, protože více definic stejného objektu tématu s různými řetězci tématu může vytvářet různé výsledky v závislosti na tom, jak a kde se na dané téma odkazuje. Další informace o tomto důležitém bodu naleznete v tématu [Více definic tématu klastru se stejným názvem](#).

Případ 2. Hierarchie publikování/odběru ve verzi 7 nebo novější

V tomto příkladu předpokládáme, že správce front není členem klastru publikování/odběru.

Pokud je správce front v produktu IBM WebSphere MQ 7 nebo novějším členem hierarchie publikování/odběru, skládá se jeho prostor tématu o všech tématech definovaných lokálně a na připojených správcích front. Prostor tématu všech správců front v hierarchii je stejný. Do lokálních témat a na globální témata se nerozděluje žádná témata.

Nastavte některou z voleb **PUBSCOPE** a **SUBSCOPE** na hodnotu QMGR, abyste zabránili publikování v tématu toku z vydavatele na odběratele připojeného k různým správcům front v hierarchii.

Předpokládáme, že jste definovali objekt tématu Alabama s řetězcem tématu USA/Alabama ve správci front QMA. Výsledek je následující:

1. Prostor tématu v QMA nyní obsahuje objekt tématu Alabama a řetězec tématu USA/Alabama.
2. Aplikace nebo administrátor může vytvořit odběr v QMA pomocí objektu tématu s názvem Alabama.
3. Aplikace může vytvořit odběr pro libovolné téma, včetně produktu USA/Alabama, do libovolného správce front v hierarchii. Pokud nebyla položka QMA lokálně definována, bude téma USA/Alabama interpretováno jako objekt tématu SYSTEM.BASE.TOPIC.

Případ 3. Hierarchie publikování/odběru a proudy v produktu IBM WebSphere MQ 6

Před IBM WebSphere MQ 7 byl prostor tématu rozdělen do samostatných proudů, které obsahovaly výchozí proud, který byl přítomen ve všech správcích front. Publikování nemohou přecházet mezi různými proudy. Pokud jsou použity pojmenované toky dat, mohou být různé prostory témat v různých správcích front odlišné. Témata jsou rozdělena na témata ve výchozím proudu a témata v různých pojmenovaných proudech.

Poznámka: Každý pojmenovaný proud tvoří samostatný prostor tématu. Chcete-li vytvořit připojenou topologii, každý pojmenovaný proud musí existovat v připojených správcích front. Předpokládáme, že proud X je definován na QMA a QMC, ale ne na QMB. Pokud je QMA nadřazeným prvkem QMBa QMB je nadřazený QMC, žádná témata v proudu X nemohou přecházet mezi QMA a QMC.

Nastavení obou voleb **PUBSCOPE** a **SUBSCOPE** buď na hodnotu QMGR, nebo na hodnotu ALL vyžaduje, aby vydavatel a odběratel na téma vyměnili pouze publikace pro lokální spotřebu, nebo aby si vyměnili pouze publikace pro globální spotřebu.

Z produktu IBM WebSphere MQ 7 nejsou proudy k dispozici prostřednictvím rozhraní API pro publikování a odběr. Pokud používáte publikování/odběr ve frontě ve správci front produktu IBM WebSphere MQ 7, proudy jsou mapovány na různé objekty témat, které mohou simulovat účinek proudů. Proud je simulován vytvořením objektu tématu, který je kořenovým tématem pro všechna témata v proudu. Správce front mapuje publikování a odběry mezi proudem a příslušným kořenovým tématem každého stromu.

Související pojmy

[“Obor publikování” na stránce 98](#)

Při konfiguraci klastru/klastru publikování/odběru dále platí, že oblast rozsahu publikování dále řídí, zda správci front předají publikování vzdáleným správcům front. K administraci oboru publikování použijte atribut tématu **PUBSCOPE**.

[“Obor odběru” na stránce 98](#)

Rozsah odběru řídí, zda odběr v jednom správci front přijímá publikování publikovaná v jiném správci front v klastru nebo v rámci hierarchie publikování/odběru nebo pouze publikování z lokálních vydavatelů.

Související úlohy

[Konfigurace sítí distribuovaných 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 formou systému zpráv publikování/odběru, protože může být škálován na vysoký počet odběratelů, aniž by to mělo škodlivé účinky 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. Poté, co je téma namapováno na adresu skupiny, není třeba správce front, protože vydavatelé a odběratelé mohou pracovat v režimu na stejné úrovni. Tento přístup umožňuje snížit zatížení serverů správců front, takže servery správců front přestanou tvořit potenciální bod selhání.

Výchozí koncepce výběrového vysílání

Výběrové vysílání produktu IBM MQ lze snadno integrovat do stávajících systémů a aplikací pomocí objektu Informace o komunikaci (COMMINFO). Dvě pole objektu TOPIC umožňují rychlou konfiguraci existujících objektů TOPIC pro podporu nebo ignorování přenosů výběrového vysílání.

Objekty potřebné pro výběrové vysílání

Následující informace představují krátký přehled o dvou objektech potřebných pro výběrové vysílání produktu IBM MQ :

Objekt COMMINFO

Objekt COMMINFO obsahuje atributy přidružený k přenosu výběrového vysílání. Další informace o parametrech objektu COMMINFO najdete v tématu [DEFINE COMMINFO](#).

Jediným polem COMMINFO, které MUSÍ být nastaveno, je název objektu COMMINFO. Tento název se pak použije k identifikaci objektu COMMINFO na 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 najdete v tématu [DEFINE TOPIC](#).

Existující témata lze použít spolu s multicast změnou hodnot následujících parametrů objektu TOPIC: **COMMINFO** a **MCAST**.

- **COMMINFO** Tento parametr uvádí název objektu informací o komunikaci výběrového vysílání.
- **MCAST** Tento parametr uvádí, zda je výběrové vysílání povoleno na této pozici ve stromu témat. Při výchozím nastavení je volba **MCAST** nastavena na hodnotu **ASPARENT**, což znamená, že atribut výběrového vysílání daného 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 přehled o tom, co se děje s odběry s různými typy odběrů a definic témat. Všechny tyto příklady předpokládají, že parametr objektu TOPIC **COMMINFO** je nastaven na název platného objektu COMMINFO:

Nastavení tématu na multicast povoleno

Je-li parametr řetězce tématu **MCAST** nastaven na hodnotu **ENABLED**, jsou povoleny odběry pro klienty podporující výběrové vysílání a je proveden odběr výběrového vysílání, pokud:

- Jedná se o trvalý odběr z klienta podporujícího výběrové vysílání.
- Jedná se o nespravovaný odběr z klienta podporujícího výběrové vysílání.
- Jedná se o odběr klienta, který není multicast.

V těchto případech je proveden odběr bez výběrového vysílání a odběry jsou degradovány na normální publish/odběr.

Téma bylo vypnuto pro výběrové vysílání

Je-li parametr řetězce tématu **MCAST** nastaven na hodnotu **DISABLED**, je vždy proveden odběr bez výběrového vysílání a odběry jsou degradovány na normální publish/odběr.

Téma je nastaveno pouze na výběrové vysílání

Je-li parametr řetězce tématu **MCAST** nastaven na hodnotu **ONLY**, jsou povoleny odběry pro klienty podporující výběrové vysílání a je proveden odběr výběrového vysílání, pokud:

- Jedná se o trvalý odběr: Trvalé odběry jsou zamítnuty s kódem příčiny [2436 \(0984\) \(RC2436\): MQRC_DURABILITY_NOT_ALLOWED](#) .
- Jedná se o nespravovaný odběr: Nespravované odběry jsou zamítnuty s kódem příčiny [2046 \(07FE\) \(RC2046\): MQRC_OPTIONS_ERROR](#)
- Jedná se o odběr klienta, který není schopen výběrového 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, klientů telemetrie, které si můžete sami zapsat nebo stáhnout zdarma, a administrativní rozhraní příkazového řádku a průzkumníku. Telemetrie odkazuje na shromažďování dat a správa široké škály vzdálených zařízení. S produktem MQ Telemetry můžete integrovat shromažďování dat a ovládání zařízení s webovými aplikacemi.

V produktu IBM WebSphere MQ 7.1 je MQ Telemetry komponenta produktu IBM MQ. Upgrade pro tyto verze je v podstatě instalací novější verze produktu IBM MQ.

Produkt Client Software Development Kit však od IBM MQ 8.0 již není dodáván jako součást produktu. Podobné ukázkové aplikace jsou i nadále volně dostupné z prostředí Eclipse Paho a MQTT.org. Viz [IBM MQ Telemetry Transport ukázkových programů](#).

Protože MQ Telemetry je komponenta produktu IBM WebSphere MQ 7.1 a novější, MQ Telemetry může být buď nainstalována s hlavním produktem, nebo nainstalovaná po instalaci hlavního produktu. Informace o migraci naleznete v části [Migrace produktu MQ Telemetry v systému Windows](#) a [Migrace produktu MQ Telemetry v systému Linux](#).

Zahrnuto v produktu MQ Telemetry jsou následující komponenty:

Kanály telemetrie

Kanály telemetrie slouží ke správě připojení klientů MQTT k produktu IBM MQ . Kanály telemetrie používají nové objekty produktu IBM MQ , jako např. produkt `SYSTEM.MQTT.TRANSMIT.QUEUE`, pro interakci s produktem IBM MQ.

Služba (MQXR) telemetrie

Klienti produktu MQTT používají telemetrickou službu `SYSTEM.MQXR.SERVICE` k připojení k telemetrickým kanálům.

IBM MQ Explorer podpora MQ Telemetry

MQ Telemetry lze spravovat pomocí IBM MQ Explorer.

Documentation

Dokumentace produktu MQ Telemetry je obsažena ve standardní dokumentaci produktu IBM MQ produktu IBM WebSphere MQ 7.1 . Dokumentace SDK pro klienty Java a C je poskytována v dokumentaci k produktu a jako Javadoc a HTML.

Koncepce telemetrie

Shromažďujete informace z prostředí kolem vás, abyste se rozhodli, co máte dělat. Jako spotřebitel si můžete zkontrolovat, co máte v obchodě předtím, než se rozhodnete o tom, jaké potraviny koupit. Chcete vědět, jak dlouho se cesta bude trvat, pokud odejdete hned, než rezervaci připojení. Před tím, než se rozhodnete navštívit lékaře, zkontrolujte své příznaky. Zkontrolujete, kdy autobus dorazí, než se rozhodne,

zda čekat. Informace pro tato rozhodnutí pocházejí přímo z měřidel a zařízení, z písemného slova na papíře nebo z obrazovky a od vás. Kde vůbec jste, a kdy budete potřebovat, sbíráte informace, přivedete to dohromady, analyzujete ho, a budete jednat na něm.

Jsou-li zdroje informací široce rozptýlené nebo nedostupné, stává se obtížné a nákladné shromažďovat ty nejpřesnější informace. Existuje-li mnoho změn, které chcete provést, nebo je obtížné provést změny, změny se neprovedou, nebo se provedou, když jsou méně účinné.

Co když se náklady na shromažďování informací z a kontroly, široce rozptýlených zařízení výrazně sníží připojením zařízení s digitální technologií k internetu? Informace lze analyzovat pomocí prostředků na internetu a v podniku. Máte více příležitostí činit informovaná rozhodnutí a jednat na nich.

Technologické trendy a environmentální a ekonomické tlaky tyto změny nahánějí:

1. Náklady na připojení a řízení senzorů a regulátory se snižují kvůli 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 osobní počítače v řadě připojení k internetovým aplikacím. Další zařízení jsou jistě následující.
3. Internet a internetové technologie usnadňují aplikaci pro získávání dat. Snadný přístup k datům je řízení využití analýzy dat k tomu, aby se data ze senzorů přeměnou na informace, které jsou užitečné v mnoha dalších řešeních.
4. Inteligentní využívání zdrojů je často rychlejším a levnějším způsobem snižování uhlíkových emisí a nákladů. Alternativy: nalezení nových prostředků nebo vývoj nových technologií pro využití stávajících zdrojů by mohlo být dlouhodobým řešením. Krátkodobé rozvíjení nových technologií nebo hledání nových zdrojů je často rizikovější, pomalejší a nákladnější než zlepšení stávajících řešení.

Příklad

Příklad ukazuje, jak tyto trendy vytvářejí nové příležitosti pro interakci s prostředím inteligentním způsobem.

Mezinárodní úmluva o bezpečnosti lidského života na moři (SOLAS) vyžaduje, aby byl systém automatické identifikace (AIS) nasazen na mnoha lodích. Je vyžadována na obchodních lodích přes 300 tun a osobní lodě. AIS je primárně systém vyhýbání se srážkám na pobřežní plavbu. Je využívána námořními úřady k monitorování a kontrole pobřežních vod.

Enthusiasty po celém světě nasazují nízkonákladové sledovací stanice AIS a umísťují pobřežní lodní informace na internet. Ostatní nadšenci píší aplikace, které spojují informace z AIS s ostatními informacemi z internetu. Výsledky jsou uvedeny na webových stránkách a publikovány pomocí Twitter a SMS.

V jedné aplikaci jsou informace z AIS stanic poblíž Southamptonu kombinovány s vlastnictvím lodí a geografickými informacemi. Aplikace dodává živé informace o příjezdech a odjezdech trajektů na Twitter. Pravidelné dojíždějící osoby používající trajekty mezi Southampton a Isle of Wight se přihlašují ke zpravodajském kanálu pomocí Twitter nebo SMS. Pokud kanál ukazuje, že jejich trajekt běží pozdě, dojížděči mohou oddálit jejich odjezd a chytit trajekt, když to doky později než jeho plánovaný čas příjezdu.

Další příklady viz [“Příklady použití Telemetry”](#) na stránce 106.

Související úlohy

[Instalace produktu MQ Telemetry](#)

[Správa serveru MQ Telemetry](#)

[Migrace produktu MQ Telemetry v systému Windows](#)

[Migrace produktu MQ Telemetry v systému Linux](#)

[Vyvíjení aplikací pro MQ Telemetry](#)

[MQ Telemetry odstraňování problémů](#)

Související odkazy

[Odkaz na MQ Telemetry](#)

Lidé, firmy a vlády stále více chtějí využívat produkt MQ Telemetry k tomu, aby lépe interagovali s prostředím, v němž žijeme a pracujeme. MQ Telemetry připojuje všechny druhy zařízení k internetu a k podniku a snižuje náklady na sestavování aplikací pro inteligentní zařízení.

Co je MQ Telemetry?

- Jedná se o funkci produktu IBM MQ , která rozšiřuje páteřní páteřní síť systému zpráv poskytovanou produktem IBM MQ na širokou škálu vzdálených senzorů, ovládacích prvků a telemetrických zařízení. MQ Telemetry rozšiřuje IBM MQ tak, aby mohl vzájemně propojit inteligentní podnikové aplikace, služby a osoby s rozhodovací pravomocí se sítěmi instrumentovaných zařízení.
- Jádrovými částmi produktu MQ Telemetry jsou:

Služba produktu MQ Telemetry (MQXR).

Tato služba se spouští uvnitř serveru IBM MQ a používá protokol IBM MQ Telemetry Transport (MQTT) ke komunikaci s telemetrickým zařízením.

Aplikace MQTT , které napíšete.

Tyto aplikace řídí informace přenášené mezi telemetrickým zařízením a správcem front produktu IBM MQ a všechny akce, které jsou prováděny v reakci na tyto informace. Při vytváření těchto aplikací se používají knihovny klienta MQTT .

Co pro mě může udělat?

- Produkt MQTT je otevřený přenos systému zpráv, který umožňuje vytvoření implementací produktu MQTT pro širokou škálu zařízení.
- Klienti produktu MQTT se mohou spouštět na zařízeních s malými nároky, které 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.
- Doručování zpráv je zajištěno a odděleno od aplikace.
- Programátoři aplikací nepotřebují mít znalosti programování komunikací.
- Zprávy mohou být vyměňovány s jinými aplikacemi systému zpráv. Může se jednat o jinou aplikaci telemetrie nebo pro aplikaci MQI, JMS nebo podnikovou aplikaci systému zpráv.

Jak ji lze používat?

- K dispozici jsou ukázkové skripty, které pracují s ukázkovou aplikací klienta IBM MQ Telemetry Transport v3 (mqttv3app.jar). Viz [IBM MQ Telemetry Transport ukázkových programů](#).
- Použijte produkt IBM MQ Explorer a jeho přidružené nástroje pro administraci funkce telemetrie IBM MQ.
- Použijte knihovny klienta, které vám pomohou vytvořit aplikace produktu MQTT , které se připojí ke správci front, a které používají systém zpráv publikování/odběru.
- Distribuujte aplikaci a knihovnu klienta do zařízení, kde se má aplikace spustit.

Jak to funguje?

- MQTT je protokol publikování odběru. Klientská aplikace MQTT může publikovat zprávy na server MQTT nebo se přihlásit k odběru zpráv zasílaných aplikacemi, které se připojují k serveru MQTT .
- Klientské aplikace produktu MQTT používají knihovny klienta, které implementují přenos zpráv produktu MQTT .
- Základní klientská aplikace produktu MQTT pracuje podobně jako standardní klient MQ , ale může být spouštěna na mnoha různých platformách a sítích.
- Služba MQ Telemetry (MQXR) změní správce front produktu IBM MQ na server MQTT .

- Pokud správce front produktu IBM MQ pracuje jako server produktu MQTT , mohou ostatní aplikace, které se připojují ke správci front, odebírat zprávy z klienta produktu MQTT a přijímat je.
- Správce front funguje jako směrovač distribuce zpráv z publikování aplikací do odebírajících aplikací.
- Zprávy mohou být distribuovány mezi různými typy klientských aplikací. Například mezi klienty telemetrie a klienty JMS .

Poznámka: MQ Telemetry nahrazuje uzly SCADA, které byly staženy ve verzi 7 produktu WebSphere Message Broker (nyní známé jako IBM Integration Bus) a běží na Windows, Linuxu a AIX.

Windows

Linux

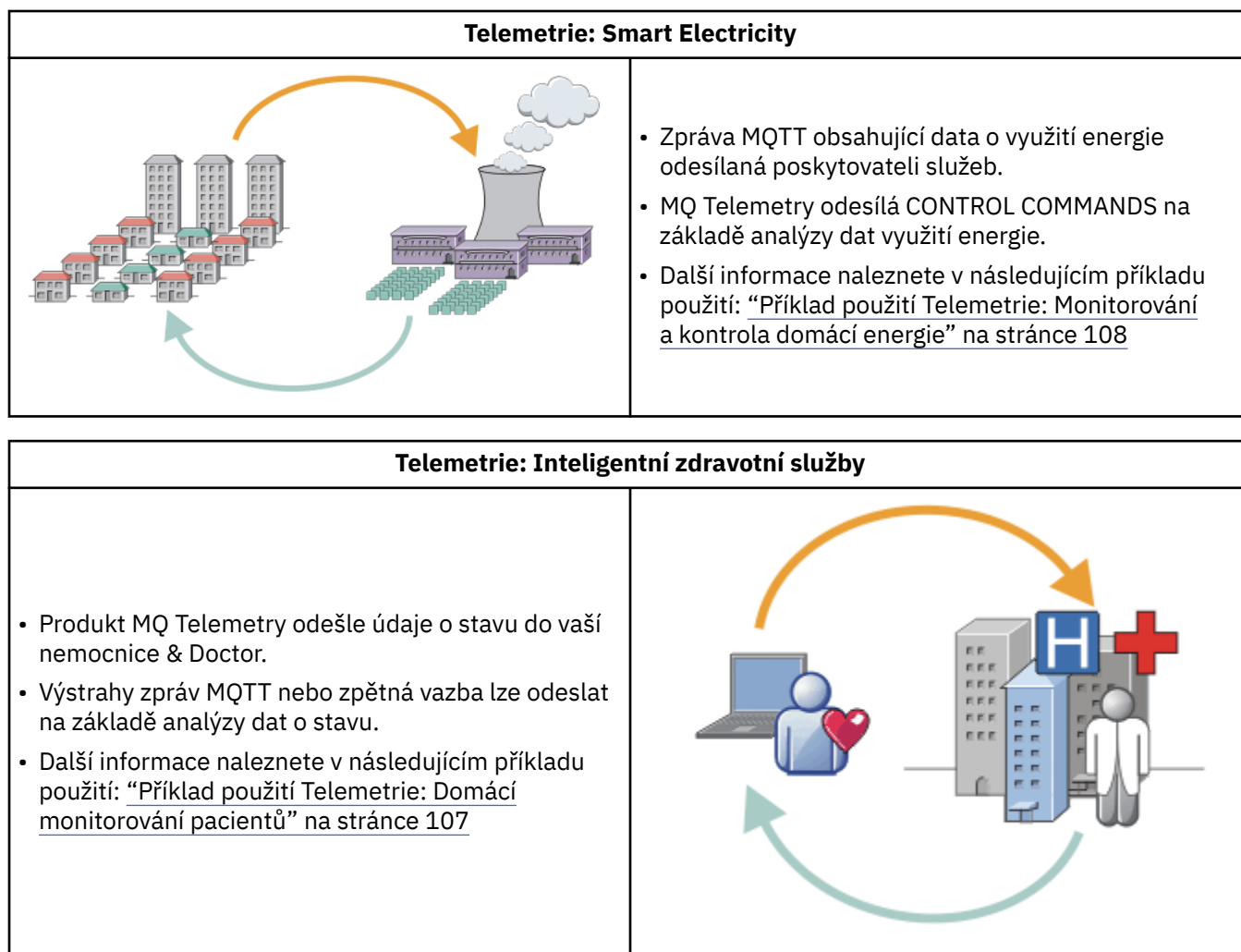
AIX

Příklady použití Telemetrie

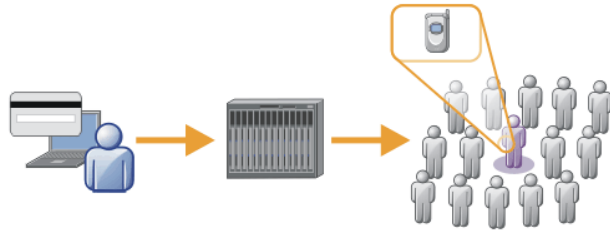
Produkt Telemetry je automatizovaný průzkum, měření dat a řízení vzdálených zařízení. Důraz je kladen na přenos dat ze zařízení na centrální řídicí bod. Telemetrie také zahrnuje odesílání konfiguračních a řídicích informací na zařízení.

Produkt MQ Telemetry propojuje malá zařízení s použitím produktu MQTT protokola připojuje zařízení k jiným aplikacím pomocí produktu IBM MQ. Produkt MQ Telemetry překlene propast mezi zařízeními a internetem a usnadňuje vytváření "chytrých řešení". Chytrá řešení odemykají množství informací dostupných na internetu a v podnikových aplikacích pro aplikace, které monitorují a řídí zařízení.

Následující diagramy demonstrují některé typické použití produktu MQ Telemetry:



Telemetrie: jeden v Crowd



- Na server banky se odešle jednoduchá transakce.
- Produkt MQ Telemetry identifikuje jednu osobu z tisíců a upozorní zákazníka na použití jejich karty.
- Produkt MQ Telemetry může použít nejjednodušší vstup informací a vyhledat daného jednotlivce.

Případy použití popsané v dílčích tématech jsou vykresleny ze skutečných příkladů. Ukazují některé způsoby použití telemetrie a některé z běžných problémů, které musí technologie telemetrie vyřešit.

Windows

Linux

AIX

Příklad použití Telemetrie: Domácí

monitorování pacientů

Ve spolupráci mezi IBM a zdravotnickým poskytovatelem na kardiálním systému péče o srdce, implantovaný kardioverter defibrilátor komunikuje s nemocnicí. Údaje o pacientovi a implantovaném zařízení se přenášejí pomocí RF telemetrie do zařízení MQTT v domácnosti pacienta.

Obvykle se přenos odehrává v noci na vysílač umístěné na nočním stolku. Vysílač přenáší data bezpečně přes telefonní systém do nemocnice, kde jsou data analyzována.

Systém snižuje počet návštěv, které musí pacient předložit lékařovi. Detekuje to, když pacient nebo zařízení potřebuje pozornost, a v případě nouze, to upozorní on-call lékaře.

Spolupráce mezi produktem IBM a poskytovatelem zdravotní péče má charakteristiky, které jsou společné pro několik případů použití telemetrie:

Neviditelnost

Zařízení nevyžaduje žádnou jinou uživatelskou intervenci, než je napájení, telefonní linku a blízkost k zařízení po část dne. Jeho provoz je spolehlivý a jednoduchý na použití.

Aby bylo možné zařízení nastavit tak, aby bylo zařízení nastaveno, předkonfiguruje zařízení. Pacient ji musí zapojit pouze. Eliminace konfigurace pacienta zjednodušuje činnost zařízení a snižuje riziko, že je zařízení nesprávně nakonfigurováno.

Klient MQTT je vestavěn jako součást zařízení. The device developer embeds the MQTT client implementation in the device and the developer, or supplier, configures the MQTT client as part of the preconfiguration.

Klient MQTT se dodává jako soubor JAR Java SE , který vývojář zahrne do své aplikace Java . Pro jiná prostředí než Java , jako je tato, může vývojář zařízení implementovat klienta v jiném jazyce s použitím publikovaných formátů a protokolu produktu MQTT . Alternativně může vývojář použít jednoho z klientů C dodávaných jako sdílené knihovny pro platformy Windows, Linux a ARM platformy.

Nepropojenost

Komunikace mezi defibrilátorem a nemocnicí má nerovné síťové charakteristiky. K řešení různých problémů se shromažďováním údajů od pacienta a k zaslání údajů do nemocnice se používají dvě různé sítě. Mezi pacientem a zařízením MQTT se používá nízkofrekvenční RF síť s krátkým rozsahem. Vysílač se připojuje do nemocnice pomocí VPN TCP/IP přes telefonní linku se sníženým šířkou pásma.

Je často nepraktické najít způsob, jak připojit každé zařízení přímo k síti Internet Protocol . Použití dvou sítí, propojených rozbočovačem, je společné řešení. Zařízení MQTT je jednoduchý rozbočovač, který ukládá informace od pacienta a přeposílá je do nemocnice.

Zabezpečení

Lékař musí být schopen důvěřovat pravosti údajů o pacientovi a pacient si přeje, aby byla zachována soukromí jejich údajů.

V některých situacích je dostačující k zašifrování připojení, pomocí VPN nebo TLS. V jiných situacích je žádoucí udržovat data zabezpečená i po jejich uložení.

Někdy není zařízení telemetrie zabezpečeno. Může se jednat například o sdílené obydlí. Uživatel zařízení musí být autentizován, aby se ujistil, že data jsou ze správného pacienta. Samotné zařízení lze ověřit na serveru pomocí TLS a server je ověřován na zařízení.

Kanál telemetrie mezi zařízením a správcem front podporuje službu JAAS pro ověřování uživatelů a TLS pro šifrování komunikací a ověřování zařízení. Přístup k publikaci je řízen správcem oprávnění k objektu v produktu IBM MQ.

Identifikátor použitý k ověření totožnosti uživatele může být mapován na jiný identifikátor, jako je například běžná identita pacienta. Obecný identifikátor zjednodušuje konfiguraci autorizace k publikování témat v produktu IBM MQ.

Připojitelnost

Spojení mezi zařízením MQTT a nemocnicí používá vytáčené připojení a pracuje se šířkou pásma až 300 baudů.

Pro efektivní provoz při 300 baudů přidá MQTT protocol do zprávy navíc záhlaví TCP/IP pouze několik dalších bajtů.

Produkt MQTT protocol poskytuje jednoduchý přenos *fire and forget* systému zpráv, který udržuje nízké latence. Může také použít více přenosů k zaručení *alespoň jednou a přesně jednou* doručení, pokud je zaručené doručení důležitější než doba odezvy. Aby bylo zaručeno doručení, jsou zprávy uloženy na zařízení, dokud nebudou úspěšně doručeny. Je-li zařízení připojeno bezdrátově, garantované doručení je zvláště užitečné.

Rozšiřitelnost

Telemetrické jednotky se obvykle nasazují ve velkém počtu, z desítek tisíc na miliony.

Připojení mnoha zařízení k systému klade vysoké nároky na řešení. Jsou zde obchodní požadavky, jako jsou náklady na zařízení a jejich software a administrativní požadavky na správu licencí, zařízení a uživatelů. Technické požadavky zahrnují zatížení na síti a na serverech.

Otevření připojení používá více prostředků serveru než udržování otevřených připojení. Ale v takovém případě, jako je tento případ, který používá telefonní linky, znamená, že připojení znamená, že spojení zůstanou otevřená déle, než se požaduje. Převody dat jsou z velké části popálené povahy. Spojení může být naplánováno přes noc, aby se zabránilo náhlý vrchol připojení před spaním.

Rozšiřitelnost klientů pomáhá klientovi v případě, že je na klientovi poskytnuta minimální požadovaná konfigurace. Klient MQTT je vestavěný do zařízení. Neexistuje žádný požadavek na konfiguraci nebo krok přijetí licence klienta MQTT, který má být zabudován do implementace zařízení pro pacienty.

Na serveru má MQ Telemetry počáteční cíl 50 000 otevřených připojení na správce front.

Připojení se spravují pomocí produktu IBM MQ Explorer. Produkt IBM MQ Explorer filtruje připojení, která se mají zobrazit, na spravovatelné číslo. Při vhodně vybraném schématu přidělení identifikátorů klientům můžete filtrovat připojení na základě zeměpisné oblasti nebo abecedně podle jména pacienta.

Příklad použití Telemetrie: Monitorování a kontrola domácí energie

Chytré měřiče shromažďují více podrobností o spotřebě energie než tradiční měřiče.

Inteligentní měřidla jsou často spojena s lokální telemetrickou sítí k monitorování a řízení jednotlivých spotřebičů v domácnosti. Některé jsou také připojeny vzdáleně pro monitorování a řízení ve vzdálenosti.

Vzdálené připojení může být nastaveno jednotlivcem, obslužným programem pro napájení nebo centrálním řídicím bodem. Vzdálený řídicí bod může číst využití napájení a poskytovat data o využití. Může poskytovat data, která ovlivňují využití, jako jsou průběžné oceňování cen a informací o počasí. Může omezit zátěž na zlepšení celkové efektivity výroby elektrické energie.

Chytré měřidlo se začíná široce rozmístit. Vláda Spojeného království, například, je v konzultaci o zavádění inteligentních měřičů do každého britského domova do roku 2020.

Příklady použití v domácím měřicím zařízení mají řadu společných charakteristik:

Neviditelnost

Pokud uživatel nechce být zapojen do úspory energie pomocí měřidla, nesmí měřidlo vyžadovat zásah uživatele. Nemusí snižovat spolehlivost dodávek energie jednotlivým spotřebičům.

Klient MQTT může být vložen do softwaru implementovaného měřidlem a nevyžaduje samostatnou instalaci nebo konfiguraci.

Nepropojenost

Komunikace mezi zařízeními a inteligentním měřidlem vyžaduje různé standardy konektivity než mezi měřidlem a vzdáleným přípojným bodem.

Připojení z inteligentního měřidla na zařízení musí být vysoce dostupné a musí splňovat síťové standardy pro domácí plochu sítě.

Je pravděpodobné, že vzdálená síť bude používat různá fyzická připojení. Některé z nich, jako jsou mobilní, mají vysoké přenosové náklady a mohou být intermitentní. Specifikace MQTT v3 je zaměřena na vzdálená připojení a připojení mezi lokálními adaptéry a inteligentním měřidlem.

Propojení mezi zásuvkami a použitostmi a měřičem, použijte síť domácí sítě, jako je Zigbee. MQTT pro síť senzoru (MQTT-S) je navržen pro práci se Zigbeem a dalšími síťovými protokoly s nízkou šířkou pásma. Produkt MQ Telemetry nepodporuje přímo MQTT-S. Vyžaduje bránu pro připojení MQTT-S k MQTT v3.

Podobně jako domácí monitorování pacientů, řešení pro domácí monitorování energie a řízení vyžadují více sítí, připojené pomocí inteligentní měřiče jako rozbočovač.

Zabezpečení

K inteligentním měřidlům je přidružena celá řada problémů zabezpečení. Tyto otázky zahrnují neodmítání transakcí, autorizaci všech akcí kontroly, které jsou zahájeny, a soukromí údajů o spotřebě energie.

Chcete-li zajistit soukromí, data přenášená mezi měřidlem a vzdáleným řídicím bodem pomocí produktu MQTT lze šifrovat pomocí TLS. Chcete-li zajistit autorizaci akcí řízení, spojení MQTT mezi měřidlem a vzdáleným řídicím bodem se může vzájemně ověřovat pomocí TLS.

Připojitelnost

Fyzický charakter vzdálené sítě se může výrazně lišit. Může používat existující širokopásmové připojení nebo může používat mobilní síť s vysokými náklady na volání a občasnou dostupností. Pro vysoké náklady, občasná, připojení MQTT je efektivní a spolehlivý protokol, viz [“Příklad použití Telemetrie: Domácí monitorování pacientů” na stránce 107](#).

Rozšiřitelnost

Nakonec energetické společnosti nebo centrální kontrolní body plánujte rozmístit desítky milionů inteligentních měřičů. Zpočátku se počty měřidel na implementaci nacházejí v desítkách na stovky tisíc. Toto číslo je srovnatelné s počátečním cílem MQTT 50 000 otevřených připojení klienta na správce front.

Kritickým aspektem architektury pro monitorování a řízení domácností je použití inteligentního měřiče jako koncentrátoru sítě. Každý adaptér zařízení je samostatný senzor. Připojením k místnímu rozbočovači pomocí produktu MQTT může rozbočovač soustředit toky dat na jednu relaci TCP/IP s centrálním řídicím bodem a také ukládat zprávy po krátkou dobu k překonání výpadků relace.

Vzdálené připojení musí být ponecháno otevřené v případech využití energie z domácností ze dvou důvodů. Zaprvé, vzhledem k tomu, že otevření připojení trvá dlouho relativně k odesílání požadavků. Čas na otevření mnoha připojení k odeslání požadavků "load-limitation" v krátkém intervalu je příliš dlouhý. Za druhé platí, že pro příjem požadavků na omezení zátěže od společnosti Power musí klient

nejprve otevřít připojení. S produktem MQTTjsou připojení vždy iniciována klientem a pro příjem žádostí o omezení zátěže od společnosti Power musí být připojení ponecháno otevřené.

Je-li rychlost otevírání připojení kritická, nebo server iniciuje časově kritické požadavky, řešení je obvykle udržován mnoha otevřenými připojeními.

Windows

Linux

AIX

Příklady použití v telemetrii: radiofrekvenční identifikace (RFID)

RFID je použití vestavěné značky RFID k identifikaci a stopování objektu bezdrátově. Štítky RFID lze číst až po rozsah několika měřících přístrojů a mimo zorné pole čtecího zařízení RFID. Pasivní značky jsou aktivovány čtečkou RFID. Aktivní značky jsou přeneseny bez externí aktivace. Aktivní značky musí mít zdroj napájení. Pasivní značky mohou zahrnovat zdroj energie ke zvýšení jejich rozsahu.

RFID se používá v mnoha aplikacích a typy příkladů použití se výrazně liší. Případy použití RFID a monitorování domovských pacientů a monitorování domácí spotřeby energie a kontroly případů použití mají určité podobnosti a rozdíly.

Neviditelnost

V mnoha případech použití je čtečka RFID implementována ve velkém počtu a musí pracovat bez zásahu uživatele. Čtecí program obsahuje vestavěného klienta MQTT pro komunikaci s centrálním řídicím bodem.

Například v distribučním skladu čtečka používá čidlo pohybu k detekci palety. Aktivuje značky RFID položek na paletě a odesílá data a požadavky do centrálních aplikací. Data se používají k aktualizaci umístění akcií. Požadavky řídí, co se stane na paletě vedle sebe, jako je například přesun do konkrétní pozice. Letecké společnosti a letištní zavazadlové systémy používají tímto způsobem RFID.

V některých případech použití RFID má čtecí zařízení standardní výpočetní prostředí, jako například Java Platform, Micro Edition (Java ME). V těchto případech může být klient produktu MQTT implementován v odlišném konfiguračním kroku po jeho výrobě.

Nepropojenost

Čtecí zařízení RFID mohou být odděleny od lokálního řídicího zařízení, které obsahuje klienta MQTT, nebo každý čtenář může vložit klienta MQTT. Typicky, geografické nebo komunikační faktory označují volbu topologie.

Zabezpečení

Ochrana soukromí a autentičnost se týká zabezpečení v příloze značek RFID. Štítky RFID jsou nenápadné a lze je tajně sledovat, obtěžovat nebo s nimi manipulováno.

Řešení problémů se zabezpečením RFID zvyšuje příležitost pro nasazení nových řešení RFID. Přestože bezpečnostní riziko je ve značce RFID a místní čtenář používá při zpracování informací o centrálním informacích přístupy k boji proti různým hrozbám. Například falšování značek může být detekováno dynamickým korinováním úrovní zásob proti doručení a odbavení.

Připojitelnost

Aplikace RFID obvykle zahrnují jak dávkové ukládání, tak i předávání informací získaných od čtecích zařízení RFID a okamžitých dotazů. V případě použití datového skladu je čtečka RFID připojena po celou dobu. Je-li značka čtena, publikuje se spolu s informacemi o čtečce. Aplikace datových skladů publikuje odezvu zpět do čtečky.

V aplikaci datových skladů je síť obvykle spolehlivá a okamžité požadavky mohou využívat zprávy *fire and forget* pro nízký výkon latence. Dávková a dopředná data mohou používat systém zpráv *přesně jednou* k minimalizaci administrativních nákladů přidružených k ztrátám dat.

Rozšiřitelnost

Pokud aplikace RFID vyžaduje okamžité odezvy, v pořadí druhé nebo dvě, pak musí být čtecí zařízení RFID připojen.

Shluk v životním prostředí používá telemetrii ke shromažďování informací o hladinách a kvalitě řek, znečišťujících látkách v ovzduší a o dalších environmentálních údajích.

Senzory se často nacházejí na vzdálených místech bez přístupu ke kabelové komunikaci. Bezdrátová šířka pásma je nákladná a spolehlivost může být nízká. Obvykle je řada senzorů prostředí v malé zeměpisné oblasti připojena k místnímu monitorovacímu zařízení v bezpečném místě. Lokální připojení mohou být připojená nebo bezdrátová.

Neviditelnost

Zařízení senzorů jsou pravděpodobně méně přístupná, nižší a jsou rozmístěna ve větším počtu, než je centrální monitorovací zařízení. Senzory jsou někdy "blbě" a lokální monitorovací zařízení obsahuje adaptéry pro transformaci a ukládání dat senzoru. The monitoring device is likely to incorporate a general-purpose computer that supports Java Platform, Standard Edition (Java SE) or Java Platform, Micro Edition (Java ME). Invisibility is unlikely to be a major requirement when configuring the MQTT client.

Nepropojenost

Možnosti senzorů a náklady a šířka pásma vzdáleného připojení obvykle vede k tomu, že lokální centrální server monitorování je připojen k centrálnímu serveru.

Zabezpečení

Pokud se řešení nepoužívá v vojenském nebo obranném případě, bezpečnost není významným požadavkem.

Připojitelnost

Mnoho použití nevyžaduje nepřetržité monitorování nebo okamžitou dostupnost dat. Výjimečná data, jako je například výstraha úrovně zahlcení, je třeba okamžitě postoupit. Data senzoru jsou agregována na lokálním monitoru, aby se snížily náklady na připojení a komunikace, a pak byly přeneseny pomocí naplánovaných připojení. Data výjimky jsou postoupena ihned, jakmile je zjištěna na monitoru.

Rozšiřitelnost

Snímače jsou soustředěny kolem místních nábojů a data senzorů jsou agregována do paketů, které jsou přenášeny podle plánu. Oba tyto faktory snižují zatížení na centrálním serveru, který by byl uložen pomocí přímo připojených senzorů.

Mobilní aplikace jsou aplikace, které běží na bezdrátových zařízeních. Zařízení jsou buď generické aplikační platformy, nebo vlastní zařízení.

Mezi obecné platformy patří handheld zařízení, jako jsou telefony a osobní data asistenti, a přenosná zařízení, jako jsou notebooky. Vlastní zařízení používají speciální účelový hardware upravený pro specifické aplikace. Příkladem vlastního mobilního zařízení je zařízení pro záznam doručení balíku "signed-for". Aplikace na vlastních mobilních zařízeních jsou často stavěny na generické softwarové platformě.

Neviditelnost

Implementace vlastních mobilních aplikací je spravována a může zahrnovat konfiguraci klientské aplikace MQTT. Invisibility is unlikely to be a major requirement when configuring the MQTT client.

Nepropojenost

Na rozdíl od lokální topologie centrálního serveru pro předchozí případy použití se mobilní klienti připojují vzdáleně. Vrstva aplikační vrstvy se připojuje přímo k aplikaci v centrálním rozbočovači.

Zabezpečení

S malou fyzickou bezpečností, mobilním zařízením a mobilními uživateli musí být autentizován. TLS se používá k potvrzení identity zařízení a JAAS pro ověření uživatele.

Připojitelnost

Pokud mobilní aplikace závisí na bezdrátovém pokrytí, musí být schopna pracovat v režimu offline a efektivně pracovat s přerušným připojením. V tomto prostředí je cílem zůstat připojení, ale aplikace musí být schopna ukládat a předávat zprávy. Často se jedná o objednávky zpráv nebo potvrzení o doručení a mají důležitou obchodní hodnotu. Musí být skladovány a předávány spolehlivě.

Rozšiřitelnost

Rozšiřitelnost není významným tématem. Počet aplikačních klientů, které pravděpodobně nepřekročí tisíce nebo desítky tisíc, v případech použití vlastních mobilních aplikací.

Windows

Linux

AIX

Připojení telemetrických zařízení ke správci

front

Zařízení telemetrie se připojují ke správci front pomocí klienta MQTT v3. Klient MQTT v3 používá protokol TCP/IP k připojení k modulu listener protokolu TCP/IP s názvem služby telemetrie (MQXR).

Připojíte-li ke správci front telemetrické zařízení, inicializuje klient produktu MQTT připojení TCP/IP pomocí metody `MqttClient.connect`. Podobně 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 instalovaného s produktem MQ Telemetry, který je označován jako služba telemetrie (MQXR). Každý správce front pracuje s maximální délkou jedné služby telemetrie (MQXR).

Služba telemetrie (MQXR) používá adresu vzdáleného soketu nastavenou každým klientem v metodě `MqttClient.connect` k přidělení připojení k telemetrickým kanálům. Adresa soketu je kombinací názvu hostitele TCP/IP a čísla portu. Více klientů používajících stejnou adresu vzdáleného soketu je připojeno ke stejnému kanálu telemetrie pomocí služby telemetrie (MQXR).

Je-li na serveru více správců front, rozdělte kanály telemetrie mezi správce front. Přidělte adresy vzdáleného soketu mezi správce front. Definujte každý kanál telemetrie pomocí jedinečné adresy vzdáleného soketu. Dva kanály telemetrie nesmí používat stejnou adresu soketu.

Pokud je stejná adresa vzdáleného soketu konfigurována pro kanály telemetrie ve více správcích front, vyhrává se první kanál telemetrie k připojení. Následné kanály, které se připojují na stejné adrese, selžou.

Je-li na serveru více síťových adaptérů, rozdělte adresy vzdálených soketů mezi kanály telemetrie. Přidělení adres soketů je zcela libovolné, pokud je jakákoli konkrétní adresa soketu konfigurována pouze na jednom kanálu telemetrie.

Nakonfigurujte produkt IBM MQ pro připojení klientů MQTT pomocí průvodců uvedených v dodatku MQ Telemetry pro IBM MQ Explorer. Případně postupujte podle pokynů v části [Konfigurace správce front pro telemetrie v systémech Linux a AIX](#) a [Konfigurace správce front pro telemetrii v produktu 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 modulem listener protokolu TCP/IP, který je spravován jako služba IBM MQ. Vytvořte službu pomocí průvodce IBM MQ Explorer nebo pomocí příkazu `runmqsc`.




Služba produktu MQ Telemetry (MQXR) se nazývá `SYSTEM.MQXR.SERVICE`.


Průvodce konfigurací ukázky Telemetrie poskytovaný ve funkci MQ Telemetry pro produkt IBM MQ Explorer vytvoří službu telemetrie a ukázkový kanál telemetrie; viz téma [Ověření instalace produktu MQ Telemetry pomocí produktu IBM MQ Explorer](#).

Vytvořte ukázkovou konfiguraci z příkazového řádku; viz téma [Ověření instalace produktu MQ Telemetry pomocí příkazového řádku](#).

Služba telemetrie (MQXR) se spouští a zastavuje automaticky se správcem front. Ovládejte službu pomocí složky služeb v produktu IBM MQ Explorer. Chcete-li službu zobrazit, musíte klepnout na ikonu a zastavit IBM MQ Explorer filtrování objektů SYSTEM z obrazovky.

Příklad, jak vytvořit službu ručně, viz

-   [Vytvoření SYSTEM.MQXR.SERVICE v systému Linux.](#)
-  [Vytvoření SYSTEM.MQXR.SERVICE v systému Windows.](#)

 Počínaje produktem IBM MQ 9.2.4 jsou položky [Vytvoření SYSTEM.MQXR.SERVICE v systému Linux](#) a [Vytvoření SYSTEM.MQXR.SERVICE v systému Windows](#) aktualizovány tak, aby určovaly výchozí klíč pro ukládání hesel MQTT TLS, které mají být šifrovány. Další informace naleznete v tématu [Šifrování hesel TLS pro kanály MQTT TLS](#).

Kanály telemetrie

Vytvořte kanály telemetrie k vytvoření připojení s různými vlastnostmi, jako je služba ověřování a autorizace produktu Java (JAAS) nebo ověření TLS nebo pro správu skupin klientů.

Vytvořte kanály telemetrie pomocí průvodce **New Telemetry Channel** dodávaného ve funkci MQ Telemetry pro produkt IBM MQ Explorer. Konfigurujte kanál pomocí průvodce, abyste akceptovali připojení od klientů MQTT na konkrétním portu TCP/IP. Od IBM WebSphere MQ 7.1 můžete nakonfigurovat MQ Telemetry pomocí programu příkazového řádku, **runmqsc**.

Vytvořte více kanálů telemetrie na různých portech, abyste usnadnili správu velkých počtů připojení klientů, rozdělením klientů do skupin. Každý kanál telemetrie má jiný název.

Můžete nakonfigurovat kanály telemetrie s různými atributy zabezpečení, chcete-li vytvořit různé typy připojení. Vytvořte více kanálů pro přijímání připojení klienta na různých adresách TCP/IP. Použijte TLS k šifrování zpráv a ověření kanálu telemetrie a klienta; viz [Konfigurace TLS pro klienty MQTT a kanály telemetrie](#). Určete ID uživatele pro zjednodušení autorizace přístupu k objektům produktu IBM MQ. Určete konfiguraci JAAS pro ověření uživatele produktu MQTT pomocí služby JAAS; viz téma [identifikace klienta, autorizace a ověření klienta produktu MQTT](#).

IBM MQ Telemetry Transport protokol

Protokol IBM MQ Telemetry Transport (MQTT) v3 je určen pro výměnu zpráv mezi malými zařízeními na nízké přenosové šířce pásma nebo nákladnými spojeními a pro spolehlivé odesílání zpráv. Používá TCP/IP.

MQTT protocol se publikuje; viz [IBM MQ Telemetry Transport formát a protokol](#). Verze 3 protokolu používá publikování/odběr a podporuje tři úrovně kvality služby: *fire and forget*, *alespoň jednou přesně jednou*.

Malá velikost záhlaví protokolu a informační obsah zprávy bajtového pole uchovává zprávy malé. Záhlaví mají 2 bajtové pevné záhlaví a až 12 bajtů dalších záhlaví proměnných. Protokol používá 12bajtová proměnná záhlaví k odběru a připojení a pouze 2 bajtová proměnná záhlaví pro většinu publikací.

Se třemi kvalitami služby můžete odobchodovat mezi nízkou latencí a spolehlivostí; viz téma [Kvality služby poskytované klientem MQTT](#). Volba *Fire and forget* nepoužívá žádné trvalé úložiště zařízení a pouze jeden přenos k odeslání nebo přijetí publikace. *Alespoň jednou přesně jednou* vyžadují trvalé úložiště na zařízení pro údržbu stavu protokolu a uložení zprávy, dokud nebude potvrzena.

MQTT klienti

Klientská aplikace MQTT je zodpovědná za shromažďování informací z telemetrického zařízení, připojení k serveru a publikování informací na server. Může se také přihlásit k odběru témat, přijímat publikace a řídit zařízení telemetrie.

Na rozdíl od aplikací klienta IBM MQ , aplikace klienta MQTT nejsou aplikacemi produktu IBM MQ . Neurčují, ke kterému se má správce front připojit. Nejsou omezeny na použití specifických programovacích rozhraní IBM MQ . Namísto toho klienti MQTT implementují protokol MQTT 3 . Můžete napsat vlastní knihovnu klienta, která má být rozhraním produktu MQTT protocol v programovacím jazyce a na platformě, dle vašeho výběru. Viz [IBM MQ Telemetry Transport format and protocol](#).

Chcete-li zjednodušit zápis aplikací klienta MQTT , použijte knihovny klienta C, Java a JavaScript , které zapouzdřují MQTT protocol pro počet platform. Pokud tyto knihovny začleníte do svých aplikací produktu MQTT , může být plně funkční klient MQTT tak krátký jako 15 řádků kódu. Knihovny klienta produktu MQTT jsou volně dostupné z prostředí Eclipse Paho a MQTT.org. Viz [IBM MQ Telemetry Transport ukázkových programů](#).

Aplikace klienta MQTT je vždy zodpovědná za zahájení připojení k telemetrickým kanálům. Po připojení buď aplikace klienta MQTT , nebo aplikace IBM MQ mohou zahájit výměnu zpráv.

Klientské aplikace MQTT a aplikace IBM MQ publikují a odebírají se do stejné sady témat. Aplikace IBM MQ může také odeslat zprávu přímo do aplikace klienta MQTT bez klientské aplikace, která nejprve vytvoří odběr. Další informace naleznete v tématu [Konfigurace distribuovaných front pro odesílání zpráv klientům MQTT](#).

Aplikace klienta MQTT jsou připojeny k produktu IBM MQ pomocí kanálu telemetrie. Kanál telemetrie funguje jako most mezi různými typy zpráv používanými produkty MQTT a IBM MQ. Vytvoří publikace a odběry ve správci front v zastoupení s klientskou aplikací MQTT . Kanál telemetrie odešle publikování, která se shodují s odběry aplikace klienta MQTT ze správce front do aplikace klienta MQTT .

Windows

Linux

AIX

Odeslání zprávy klientovi MQTT

Aplikace produktu IBM MQ mohou odesílat zprávy klientů produktu MQTT v3 publikováním do odběrů vytvořených klienty nebo přímým odesláním zpráv. Klienti produktu MQTT mohou odesílat zprávy do jiné prostřednictvím publikování v tématech přihlášených k odběru jinými klienty.

Klient produktu MQTT se přihlásí k odběru publikace, kterou přijímá z produktu IBM MQ .

Provedte úlohu [“Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer”](#) na stránce 116 , chcete-li odeslat publikaci z produktu IBM MQ na klienta produktu MQTT .

Standardní způsob příjmu zpráv klientem produktu MQTT v3 je za účelem vytvoření odběru tématu nebo sady témat. V ukázkovém úseku kódu Obrázek 45 na stránce 115 se klient produktu MQTT přihlašuje s použitím řetězce tématu "MQTT Examples". Aplikace IBM MQ C, Obrázek 46 na stránce 115, publikuje do tématu pomocí řetězce tématu "MQTT Examples". V úseku kódu Obrázek 47 na stránce 116 přijímá klient produktu MQTT publikování v metodě zpětného volání, `messageArrived`.

Další informace o konfiguraci produktu IBM MQ k odesílání publikování v rámci odpovědí na odběry klientů produktu MQTT naleznete v tématu [Publikování zprávy v reakci na odběr klienta MQTT](#).

Aplikace IBM MQ odešle zprávu přímo klientovi produktu MQTT .

Provedte úlohu [“Odeslání zprávy klientovi produktu MQTT pomocí produktu IBM MQ Explorer”](#) na stránce 120 , chcete-li odeslat zprávu přímo z produktu IBM MQ na klienta MQTT .

Zpráva odeslaná tímto způsobem do klienta produktu MQTT se nazývá nevyžádaná zpráva. Klienti MQTT v3 přijímají nevyžádané zprávy jako publikace s nastaveným názvem tématu. Služba telemetrie (MQXR) nastavuje název tématu na název vzdálené fronty.

Další informace o tom, jak nakonfigurovat produkt IBM MQ k odesílání zpráv přímo klientům MQTT , naleznete v tématu [Přímé odeslání zprávy klientovi](#).

Klient produktu MQTT publikuje zprávu.

Klient MQTT v3 může publikovat zprávu, která je přijata jiným klientem produktu MQTT v3, ale nemůže odeslat nevyžádanou zprávu. Úsek kódu [Obrázek 48](#) na stránce [116](#) zobrazuje způsob, jakým klient MQTT v3, zapsaný v produktu Java, publikuje zprávu.

Typický vzorek pro odeslání zprávy jednomu specifickému klientovi MQTT v3 je pro každého klienta, aby vytvořil odběr pro vlastní `ClientIdentifier`. Chcete-li publikovat zprávu z jednoho klienta produktu MQTT do jiného klienta produktu MQTT pomocí identifikátoru `ClientIdentifier` jako řetězec tématu, proveďte úlohu [“Publikování zprávy na konkrétní klienta MQTT v3”](#) na stránce [122](#).

Ukázka snippetů kódu

Úsek kódu v produktu [Obrázek 45](#) na stránce [115](#) zobrazuje, jak klient MQTT napsaný v produktu Java vytváří odběr. Také potřebuje metodu zpětného volání, `messageArrived` která přijímá 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 45. Odběratel klienta MQTT v3

Úsek kódu v souboru [Obrázek 46](#) na stránce [115](#) ukazuje, jak aplikace IBM MQ zapsaná v C odesílá publikaci. Úsek kódu je extrahován z úlohy, [Vytvořit vydavatele na téma 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 46. IBM MQ vydavatel

Když je publikace doručena, volá klient MQTT metodu `messageArrived` třídy `MqttCallback` aplikačního klienta MQTT.

```

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 47. Metoda `messageArrived`

Obrázek 48 na stránce 116 zobrazí MQTT v3 publikující zprávu na odběr vytvořený v produktu Obrázek 45 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 48. Vydavatel klienta MQTT v3

Windows Linux AIX Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer

Postupujte podle kroků uvedených v této úloze a publikujte zprávu pomocí produktu IBM MQ Explorera přihlaste se k ní pomocí obslužného programu klienta MQTT . Další úloha ukazuje, jak nakonfigurovat alias správce front místo nastavení výchozí přenosové fronty na `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Než začnete

Úloha předpokládá, že jste obeznámeni s IBM MQ a IBM MQ Explorer, a že jsou nainstalovány funkce IBM MQ a MQ Telemetry .

Uživatel, který vytváří prostředky správce front pro tuto úlohu, musí mít dostatečné oprávnění k provedení tohoto úkolu. Pro účely demonstrace se předpokládá, že ID uživatele produktu IBM MQ Explorer je členem skupiny `mqm` .

Informace o této úloze

V rámci úlohy vytvoříte téma v produktu IBM MQ a přihlásíte se k odběru tématu pomocí obslužného programu klienta MQTT . Když publikujete do tématu pomocí produktu IBM MQ Explorer, klient MQTT obdrží publikování.

Postup

Proveďte jednu z následujících úloh:

- Nainstalovali jste produkt MQ Telemetry, ale dosud jste jej nespustili. Proveďte úlohu: [“Spustit úlohu bez telemetrické služby \(MQXR\), která je ještě definována”](#) na stránce 117.

- Nyní jste spustili telemetrii IBM MQ , ale chcete použít nového správce front k provedení demonstrace. Proveďte úlohu: “Spustit úlohu bez telemetrické služby (MQXR), která je ještě definována” na stránce 117.
- Chcete tuto úlohu provést s použitím existujícího správce front, který nemá definovány žádné prostředky telemetrie. Nepřejete si spustit průvodce **Definovat ukázkovou konfiguraci** .
 - a. Chcete-li nastavit telemetrii, proveďte jednu z následujících úloh:
 - Konfigurace správce front pro telemetrie v systémech Linux a AIX
 - Konfigurace správce front pro telemetrie v produktu 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, proveďte tuto úlohu: “Spustit úlohu se spuštěnou službou telemetrie (MQXR)” na stránce 118.

Jak pokračovat dále

Chcete-li odeslat zprávu přímo do obslužného programu klienta, proveďte příkaz “Odeslání zprávy klientovi produktu MQTT pomocí produktu IBM MQ Explorer” na stránce 120 .

Spustit úlohu bez telemetrické služby (MQXR), která je ještě definována

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 ní pomocí obslužného programu klienta MQTT .

Informace o této úloze

Když nastavíte ukázkové prostředky telemetrie pomocí volby **Definovat ukázkovou konfiguraci**, průvodce nastaví oprávnění ID uživatele "guest". Pečlivě zvažte, zda chcete, aby bylo ID uživatele hosta autorizováno tímto způsobem. `guest` v systémech Windowsa nobody na systému Linuxmají oprávnění k publikování a odběru kořene stromu témat a k vkládání zpráv do `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Průvodce také nastaví výchozí přenosovou frontu na `SYSTEM.MQTT.TRANSMIT.QUEUE`, což může kolidovat s aplikacemi spuštěnými v existujícím správcí front. Je to možné, ale pracné, pro konfiguraci telemetrie a nepoužití výchozí přenosové fronty; proveďte následující úlohu: “Použití aliasu správce front” na stránce 119. V této úloze vytvoříte správce front, abyste se vyhnuli možnosti kolidovat s žádnou existující výchozí přenosovou frontou.

Postup

1. Pomocí produktu IBM MQ Explorer vytvořte a spusťte nového správce front.
 - a) Klepněte pravým tlačítkem myši na složku `Queue Managers` > **Nový** > **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** .
Zaškrťovací políčko **Spustit obslužný program klienta MQTT** ponechte zaškrtnuté.
3. Vytvořte odběr produktu MQTT Example pomocí obslužného programu klienta MQTT .
 - a) Klepněte na tlačítko **Připojit**.
Záznam **Historie klienta** zaznamená událost `Connected` .
 - b) Zadejte příkaz `MQTT Example` do pole **Odběr \ Téma** > **Přihlásit se k odběru**.
Záznam **Historie klienta** zaznamená událost `Subscribed` .
4. Vytvořte `MQTTExampleTopic` v IBM MQ.

- a) Klepněte pravým tlačítkem myši na složku Queue Managers*QmgrName*\Topics v produktu **MQ Explorer** > **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 IBM MQ Explorer.
 - b) Klepněte pravým tlačítkem myši na volbu MQTTExampleTopic > **Testovat publikování ...**
 - c) Zadejte Hello World! do pole **Data zprávy** > **Publikovat zprávu** > Přepněte se do okna obslužného programu klienta MQTT .

Záznam **Historie klienta** zaznamená událost Received .

Spustit úlohu se spuštěnou službou telemetrie (MQXR)

Vytvořit kanál telemetrie a téma. Autorizujte uživatele k použití tématu a přenosové fronty telemetrie. Publikujte zprávu pomocí produktu IBM MQ Explorera přihlaste se k ní pomocí obslužného programu klienta MQTT .

Než začnete

V této verzi této 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 odesílal publikování do obslužného programu klienta MQTT .

Krok “1” na stránce 118 úlohy nastaví výchozí přenosovou frontu na SYSTEM.MQTT.TRANSMIT.QUEUE, což může kolidovat s aplikacemi spuštěnými v existujícím správci front. Je to možné, ale pracné, pro konfiguraci telemetrie a nepoužití výchozí přenosové fronty; proveďte následující úlohu: “Použití aliasu správce front” na stránce 119.

Postup

1. Nastavte SYSTEM.MQTT.TRANSMIT.QUEUE jako výchozí přenosovou frontu.
 - a) Klepněte pravým tlačítkem myši na nabídku Queue Managers*QmgrName* folder > **Vlastnosti ...**
 - b) V navigátoru klepněte na volbu **Komunikace** .
 - c) Klepněte na tlačítko **Vybrat ...** > Vybrat SYSTEM.MQTT.TRANSMIT.QUEUE > **OK** > **OK**.
2. Vytvořte kanál telemetrie MQTTExampleChannel pro připojení obslužného programu klienta MQTT k produktu IBM MQa spusťte obslužný program klienta MQTT .
 - a) Klepněte pravým tlačítkem myši na složku Queue Managers*QmgrName* \Telemetry\Channels v produktu **MQ Explorer** > **Nový** > **Kanál telemetrie ...**
 - b) Zadejte MQTTExampleChannel do pole **Název kanálu** > **Další** > **Další**.
 - c) Změňte **Pevné ID uživatele** na panelu autorizace klienta na ID uživatele, který se chystá publikovat, a přihlaste se k odběru MQTTExample > **Další**.
 - d) Ponechte volbu **Spustit obslužný program klienta** zaškrtnutou > **Dokončit**.
3. Vytvořte odběr produktu MQTT Example pomocí obslužného programu klienta MQTT .
 - a) Klepněte na tlačítko **Připojit**.

Záznam **Historie klienta** zaznamená událost Connected .

b) Zadejte příkaz `MQTT Example` do pole **Odběr \ Téma > Přihlásit se k odběru**.

Záznam **Historie klienta** zaznamená událost `Subscribed`.

4. Vytvořte `MQTTExampleTopic` v IBM MQ.

a) Klepněte pravým tlačítkem myši na složku `Queue Managers\QmgrName\Topics` v produktu **MQ Explorer > 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. Chcete-li uživatele, nikoli ve skupině `mqm` publikovat a přihlásit se k odběru tématu `MQTTExample`, postupujte takto:

a) Autorizovat uživatele pro publikování a přihlášení k odběru tématu `MQTTExampleTopic`:

```
setmqaut -m qMgrName -t topic -n MQTTExampleTopic -p User ID -all +pub +sub
```

b) Autorizujte uživatele, aby vložil zprávu do `SYSTEM.MQTT.TRANSMIT.QUEUE`:

```
setmqaut -m qMgrName -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p User ID -all +put
```

6. Publikujte `Hello World!` do tématu `MQTT Example` pomocí IBM MQ Explorer.

a) Klepněte na složku `Queue Managers\QmgrName\Topics` v IBM MQ Explorer.

b) Klepněte pravým tlačítkem myši na volbu `MQTTExampleTopic > Testovat publikování ...`

c) Zadejte `Hello World!` do pole **Data zprávy > Publikovat zprávu > Přepněte se do okna obslužného programu klienta MQTT**.

Záznam **Historie klienta** zaznamená událost `Received`.

Použití aliasu správce front

Publikujte zprávu do obslužného programu klienta MQTT pomocí produktu IBM MQ Explorer bez nastavení výchozí přenosové fronty na `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Úloha je pokračováním předchozí úlohy a používá alias správce front, aby se zabránilo nastavení výchozí přenosové fronty na `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Než začnete

Dokončete buď úlohu, "[Spustit úlohu bez telemetrické služby \(MQXR\), která je ještě definována](#)" na stránce 117 nebo úlohu, "[Spustit úlohu se spuštěnou službou telemetrie \(MQXR\)](#)" na stránce 118.

Informace o této úloze

Pokud klient produktu MQTT vytvoří odběr, produkt IBM MQ odešle svou odezvu pomocí volby `ClientIdentifier` jako název vzdáleného správce front. V této úloze používá hodnotu `ClientIdentifier`, `MyClient`.

Pokud zde není žádná přenosová fronta nebo alias správce front s názvem `MyClient`, odezva se umístí do výchozí přenosové fronty. Nastavením výchozí přenosové fronty na `SYSTEM.MQTT.TRANSMIT.QUEUE` se klientovi MQTT dostane odpovědi.

Pomocí aliasů správce front se můžete vyhnout nastavení výchozí přenosové fronty na `SYSTEM.MQTT.TRANSMIT.QUEUE`. Je třeba nastavit alias správce front pro všechny `ClientIdentifier`. Obvykle je příliš mnoho klientů, aby bylo praktické používat aliasy správce front. Často `ClientIdentifier` je nepředvídatelný, takže je nemožné nakonfigurovat telemetrii tímto způsobem.

Nicméně za určitých okolností může být nutné nakonfigurovat výchozí přenosovou frontu na jinou hodnotu než `SYSTEM.MQTT.TRANSMIT.QUEUE`. Kroky v [Procedure](#) konfigurují alias správce front místo nastavení výchozí přenosové fronty na `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Postup

- Odeberte `SYSTEM.MQTT.TRANSMIT.QUEUE` jako výchozí přenosovou frontu.
 - Klepněte pravým tlačítkem myši na nabídku `Queue Managers\QmgrName folder > Vlastnosti ...`
 - V navigátoru klepněte na volbu **Komunikace**.
 - Odeberte `SYSTEM.MQTT.TRANSMIT.QUEUE` z pole **Výchozí přenosová fronta > OK**.
- Ověřte, že již nelze vytvořit odběr pomocí obslužného programu klienta MQTT :
 - Klepněte na tlačítko **Připojit**.

Záznam **Historie klienta** zaznamená událost `Connected`.
 - Zadejte příkaz `MQTT Example` do pole **Odběr \ Téma > Přihlásit se k odběru**.

Historie klienta zaznamená událost `Subscribe failed` a `Connection lost`.
- Vytvořte alias správce front pro `ClientIdentifier, MyClient`.
 - Klepněte pravým tlačítkem myši na složku `Queue Managers\QmgrName\Queues > Nový > Definice vzdálené fronty`.
 - Pojmenujte definici, `MyClient > Další`.
 - Do pole **Vzdálený správce front** zadejte hodnotu `MyClient`.
 - Zadejte `SYSTEM.MQTT.TRANSMIT.QUEUE` do pole **Přenosová fronta > Dokončit**.
- Připojte znovu obslužný program klienta MQTT .
 - Zkontrolujte, zda je **Identifikátor klienta** nastaven na hodnotu `MyClient`.
 - Připojit**

Záznam **Historie klienta** zaznamená událost `Connected`.
- Vytvořte odběr produktu `MQTT Example` pomocí obslužného programu klienta MQTT .
 - Klepněte na tlačítko **Připojit**.

Záznam **Historie klienta** zaznamená událost `Connected`.
 - Zadejte příkaz `MQTT Example` do pole **Odběr \ Téma > Přihlásit se k odběru**.

Záznam **Historie klienta** zaznamená událost `Subscribed`.
- Publikujte `Hello World!` do tématu `MQTT Example` pomocí `IBM MQ Explorer`.
 - Klepněte na složku `Queue Managers\QmgrName\Topics` v `IBM MQ Explorer`.
 - Klepněte pravým tlačítkem myši na volbu `MQTTExampleTopic > Testovat publikování ...`
 - Zadejte `Hello World!` do pole **Data zprávy > Publikovat zprávu > Přepněte se do okna obslužného programu klienta MQTT**.

Záznam **Historie klienta** zaznamená událost `Received`.

Odeslání zprávy klientovi produktu MQTT pomocí produktu IBM MQ Explorer

Odešlete zprávu do fronty produktu MQTT prostřednictvím vložení zprávy do fronty produktu IBM MQ pomocí příkazu `IBM MQ Explorer`. Tato úloha ukazuje, jak nakonfigurovat definici vzdálené fronty pro odeslání zprávy přímo klientovi produktu MQTT .

Než začnete

Provedte úlohu “Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer” na stránce 116. Ponechte obslužný program klienta produktu MQTT připojený.

Informace o této úloze

Tato úloha demonstruje odeslání zprávy klientovi produktu MQTT pomocí fronty namísto publikování do tématu. V klientovi se nevytvoří odběr. 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 provedení úlohy, zadejte vlastní ClientIdentifier místo použití generovaného ClientIdentifier vytvořeného pomocí obslužného programu klienta MQTT .

- a) Chcete-li odpojit obslužný program klienta MQTT z kanálu telemetrie, klepněte na volbu **Odpojit** .

Volba **Historie klienta** zaznamenává událost Disconnected .

- b) Změňte **Identif klienta** na MyClient.

- c) Klepněte na tlačítko **Připojit**.

Volba **Historie klienta** zaznamenává událost Connected .

2. Zkontrolujte, že obslužný program klienta MQTT již nepřijímá publikování pro produkt MQTTExampleTopic.

- a) Klepněte na složku Queue Managers\QmgrName\Topics v IBM MQ Explorer.

- b) Klepněte pravým tlačítkem myši na volbu MQTTExampleTopic > **Testovat publikování ...**

- c) Zadejte Hello World! do pole **Data zprávy** > **Publikovat zprávu** > Přepněte se do okna obslužného programu klienta MQTT .

Žádná událost se nezaznamenává v **historii klienta**.

3. Vytvořte definici vzdálené fronty pro klienta.

Nastavte parametr ClientIdentifier, MyClient, jako název vzdáleného správce front v definici vzdálené fronty. Použijte libovolný název, který se vám líbí jako název vzdálené fronty. Název vzdálené fronty je předán klientovi MQTT jako název tématu.

- a) Klepněte pravým tlačítkem myši na složku Queue Managers\QmgrName\Queues > **Nový** > **Definice vzdálené fronty**.

- b) Pojmenujte definici, MyClientRemoteQueue > **Další**.

- c) Zadejte MQTTExampleQueue do pole **Vzdálená fronta** .

- d) Do pole **Vzdálený správce front** zadejte hodnotu MyClient .

- e) Zadejte SYSTEM.MQTT.TRANSMIT.QUEUE do pole **Přenosová fronta** > **Dokončit**.

4. Vložte testovací zprávu do produktu MyClientRemoteQueue.

- a) Klepněte pravým tlačítkem myši na **MyClientRemoteQueue** > **Put test message ...**

- b) Zadejte Hello queue! do datového pole Zpráva > **Vložit zprávu** > **Zavřít** .

Záznam **Historie klienta** zaznamená událost Received .

5. Odeberte SYSTEM.MQTT.TRANSMIT.QUEUE jako výchozí přenosovou frontu.

- a) Klepněte pravým tlačítkem myši na nabídku Queue Managers\QmgrName folder > **Vlastnosti ...**

- b) V navigátoru klepněte na volbu **Komunikace** .

- c) Odeberte SYSTEM.MQTT.TRANSMIT.QUEUE z pole **Výchozí přenosová fronta** > **OK**.
6. Zopakujte krok "4" na stránce 121.
- MyClientRemoteQueue je definice vzdálené fronty, která explicitně pojmenovává přenosovou frontu. Chcete-li odeslat zprávu do produktu MyClient, nemusíte definovat výchozí přenosovou frontu.

Jak pokračovat dále

Vzhledem k tomu, že výchozí přenosová fronta již není nastavena na SYSTEM.MQTT.TRANSMIT.QUEUE, obslužný program klienta produktu MQTT nemůže vytvořit nový odběr, pokud není alias správce front definován pro ClientIdentifier, MyClient. Obnovte výchozí přenosovou frontu do SYSTEM.MQTT.TRANSMIT.QUEUE.

Windows Linux AIX **Publikování zprávy na konkrétní klienta MQTT v3**

Publikujte zprávu z jednoho klienta MQTT v3 do jiného pomocí příkazu ClientIdentifier jako názvu tématu a produktu IBM MQ jako zprostředkovatele publikování a odběru.

Než začnete

Provedte úlohu "[Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer](#)" na stránce 116. Ponechte obslužný program klienta produktu MQTT připojený.

Informace o této úloze

Úloha ukazuje dvě věci:

1. Přihlášení k odběru tématu v rámci jednoho klienta MQTT a příjem publikování z jiného klienta produktu MQTT .
2. Nastavení "dvoubodových" odběrů pomocí řetězce 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 provedení úlohy, zadejte vlastní ClientIdentifier místo použití generovaného ClientIdentifier vytvořeného pomocí obslužného programu klienta MQTT .

- a) Chcete-li odpojit obslužný program klienta MQTT z kanálu telemetrie, klepněte na volbu **Odpojit** .

Volba **Historie klienta** zaznamenává událost Disconnected .

- b) Změňte **Identif klienta** na MyClient.

- c) Klepněte na tlačítko **Připojit**.

Volba **Historie klienta** zaznamenává událost Connected .

2. Vytvoření odběru pro téma MyClient

MyClient je ClientIdentifier tohoto klienta.

- a) Zadejte příkaz MyClient do pole **Odběr \ Téma** > **Přihlásit se k odběru**.

Záznam **Historie klienta** zaznamená 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 obslužný program klienta MQTT ...**

- c) Klepněte na tlačítko **Připojit**.

Volba **Historie klienta** zaznamenává událost Connected .

4. Publikujte Hello MyClient! na téma MyClient.

- a) Zkopírujte téma odběru, MyClient, z obslužného programu klienta MQTT spuštěného s ClientIdentifier, MyClient.
- b) Vložte MyClient do pole **Publikování \ Téma** každé z instancí obslužného programu klienta MQTT .
- c) Do pole **Publikační \ zpráva** zadejte Hello MyClient! .
- d) V obou instancích klepněte na tlačítko **Publikovat** .

Výsledky

Historii klienta v obslužném programu klienta MQTT s hodnotou ClientIdentifier, MyClient, zaznamenává dvě události **Přijato** a jednu událost **Publikováno** . Další 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 SYSTEM.MQTT.TRANSMIT.QUEUE ?
2. Vytvořili jste alias správce front nebo definice vzdálených front, které odkazují na MyClient při provádění jiných cvičení? V případě, že máte problém s konfigurací, odstraňte všechny prostředky, které odkazují na MyClient, jako jsou alias správce front nebo přenosové fronty. Odpojte obslužné programy klienta, zastavte a znovu spusťte službu telemetrie (MQXR).

Windows

Linux

AIX

Odeslání zprávy aplikaci IBM MQ z klienta MQTT

Aplikace produktu IBM MQ může obdržet zprávu od klienta MQTT v3 přihlášením k odběru tématu. Klient produktu MQTT se připojí k produktu IBM MQ pomocí kanálu telemetrie a odešle zprávu do aplikace produktu IBM MQ prostřednictvím publikování na stejné téma.

Proveďte úlohu [“Publikování zprávy v produktu IBM MQ z klienta produktu MQTT”](#) na stránce 123, abyste se dozvěděli, jak odeslat publikaci z klienta produktu MQTT na odběr definovaný v produktu IBM MQ.

Je-li téma klastrováno nebo distribuováno pomocí hierarchie publikování/odběru, může být odběr umístěn v jiném správci front do správce front, ke kterému je klient MQTT připojen.

Windows

Linux

AIX

Publikování zprávy v produktu IBM MQ z klienta produktu MQTT

Vytvořte odběr tématu pomocí produktu IBM MQ Explorer a publikujte na téma pomocí obslužného programu klienta MQTT .

Než začnete

Proveďte úlohu [“Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer”](#) na stránce 116. Ponechte obslužný program klienta produktu MQTT připojený.

Informace o této úloze

Úloha demonstruje publikování zprávy pomocí klienta produktu MQTT a přijetí publikování s použitím nespravovaného trvalého odběru vytvořeného pomocí produktu IBM MQ Explorer.

Postup

1. Vytvořte trvalý odběr pro řetězec 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 MQTTEsampleQueue 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 tlačítko **Vybrat ...** > MQTTExampleTopic > **OK**.
 Již jste vytvořili téma MQTTExampleTopic v kroku “4” na stránce 117 z “Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer” na stránce 116.
 - f) Zadejte MQTTExampleQueue jako název místa určení > **Dokončit**.
2. Jako volitelný krok nastavte frontu pro použití jiným uživatelem, bez oprávnění mqm .
- Nastavujete-li konfiguraci pro uživatele s nižším oprávněním než mqm, musíte předat oprávnění put a get k MQTTExampleQueue. Přístup k tématu a k přenosové frontě byl nakonfigurován v produktu “Publikování zprávy do obslužného programu klienta MQTT z produktu IBM MQ Explorer” na stránce 116.
- a) Autorizujte uživatele k vložení a získání do fronty MQTTExampleQueue:
- ```
setmqaut -m qMgrName -t queue -n MQTTExampleQueue -p User ID -all +put +get
```
3. Publikujte Hello IBM MQ! na téma MQTT Example pomocí obslužného programu klienta MQTT .
- Pokud jste neopustili obslužný program klienta MQTT připojený, klepněte pravým tlačítkem myši na kanál **PlainText** > **Spustit obslužný program klienta MQTT ...** > **Připojit**.
- a) Zadejte MQTT Example do pole **Publikace \ Topic** .
  - b) Zadejte Hello IBM MQ! do pole **Publikace \ Message** > **Publish**.
4. Otevřete složku Queue Managers\QmgrName\Queues a vyhledejte MQTTExampleQueue.
- Pole **Aktuální hloubka fronty** je 1
5. Klepněte pravým tlačítkem myši na položku MQTTExampleQueue > **Procházet zprávy ...** a prozkoumejte publikaci.

Windows

Linux

AIX

## Aplikace pro publikování/odběr produktu MQTT

Prostřednictvím publikování/odběru založeného na tématu můžete psát aplikace produktu MQTT .

Když je klient MQTT připojen, tok publikací probíhá buď ve směru mezi klientem a serverem. Publikace se odesílají z klienta, když jsou informace publikovány na klientovi. Publikace jsou přijímány na klientovi, když je publikována zpráva na téma, které odpovídá odběru vytvořenému klientem.

Zprostředkovatel IBM MQ publish/subscribe spravuje témata a odběry vytvořené klienty MQTT . Témata vytvořená klienty produktu MQTT sdílejí stejný prostor témat jako témata vytvořená aplikacemi produktu IBM MQ .

Publikace, které se shodují s řetězcem tématu v odběru klienta MQTT , jsou umístěny v produktu SYSTEM.MQTT.TRANSMIT.QUEUE se vzdáleným názvem správce front nastaveným na hodnotu ClientIdentifier klienta. Služba telemetrie (MQXR) předává publikování klientovi, který vytvořil odběr. Používá ClientIdentifier, který byl nastaven jako název vzdáleného správce front k identifikaci klienta.

Obvykle musí být SYSTEM.MQTT.TRANSMIT.QUEUE definováno jako výchozí přenosová fronta. Je možné, ale znesnadňovat konfiguraci produktu MQTT , aby nepoužíval výchozí přenosovou frontu. Další informace naleznete v tématu [Konfigurace distribuovaných front pro odesílání zpráv klientům MQTT](#) .

Klient produktu MQTT může vytvořit trvalou relaci; viz “MQTT bezstavové a stavové relace” na stránce 127. Odběry vytvořené v trvalé relaci jsou trvalé. Publikování, která dorazí pro klienta s trvalou relací, jsou uloženy v produktu SYSTEM.MQTT.TRANSMIT.QUEUE a jsou předávány klientovi po opětovném připojení.

Klient produktu MQTT může také publikovat a odebírat zachovaná publikování, viz téma [Zachovaná publikování a klienti MQTT](#). Odběratel na téma zachovaného publikování přijímá nejnovější publikování

v rámci daného tématu. Odběratel obdrží zachované publikování, když vytvoří odběr, nebo když se znovu připojí ke své dřívější relaci.

Windows

Linux

AIX

## Aplikace Telemetrie

Zapisovat telemetrické aplikace pomocí toků zpráv IBM MQ nebo IBM Integration Bus .

K programování aplikací telemetrie v produktu IBM MQ použijte produkt JMS, MQI nebo jiná programovací rozhraní produktu IBM MQ .

Služba telemetrie (MQXR) se převádí mezi zprávami MQTT v3 a IBM MQ zprávami. Vytváří odběry a publikování v zastoupení klientů MQTT a předává publikace klientům MQTT . Publikování je informační obsah zprávy produktu MQTT v3 . Informační obsah obsahuje záhlaví zpráv a bajtové pole ve formátu `jms-bytes` . Server telemetrie mapuje záhlaví mezi zprávou MQTT v3 a zprávou IBM MQ ; viz [“Integrace produktu MQ Telemetry se správci front”](#) na stránce 125.

Vstupní uzly Publication, MQInput a JMSInput slouží k odesílání a přijímání publikování mezi klienty IBM Integration Bus a MQTT .

Pomocí toků zpráv můžete integrovat telemetrii s webovými servery pomocí HTTP a s dalšími aplikacemi pomocí adaptérů IBM MQ a WebSphere .

Windows

Linux

AIX

## Integrace produktu MQ Telemetry se správci front

### front

Klient produktu MQTT je integrován s produktem IBM MQ jako aplikace typu publikování/odběr. 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 jako výsledek klientů produktu MQTT , včetně sebe samotného, nebo jiných aplikací produktu IBM MQ publikujících na témata svých odběrů. Pravidla se používají pro rozhodnutí o atributech publikování.

Mnohé z atributů přidružených k tématům, publikacím, odběrům a zprávám, které poskytuje produkt IBM MQ, nejsou podporovány. [“Klient produktu MQTT pro zprostředkovatele publikování/odběru IBM MQ”](#) na stránce 125 a [“IBM MQ na klienta MQTT”](#) na stránce 127 popisují, jak jsou nastaveny atributy publikování. Nastavení závisí na tom, zda publikování probíhá nebo ze zprostředkovatele publikování a odběru IBM MQ .

V tématech IBM MQ publish/subscribe jsou témata přidružená k objektům tématu administrace. Témata vytvořená klienty MQTT se neliší. Když klient MQTT vytvoří řetězec tématu pro publikování zprostředkovatele publikování a odběru IBM MQ , přiřadí jej k objektu administrativního tématu. Zprostředkovatel mapuje řetězec tématu v publikování na nejbližší nadřazený objekt tématu administrativního tématu. Mapování je stejné jako u aplikací produktu IBM MQ . Není-li vytvořeno žádné uživatelem vytvořené téma, je téma publikování mapováno na SYSTEM.BASE.TOPIC. Atributy, které jsou použity na publikování, jsou odvozeny z objektu tématu.

Když aplikace IBM MQ nebo administrátor vytvoří odběr, je tento odběr pojmenován. Vypište odběry pomocí produktu IBM MQ Explorer nebo pomocí příkazů `runmqsc` či PCF. Všechny odběry klienta MQTT mají název. Dostanou název formuláře: `ClientIdentifier:Topic name`

### Klient produktu MQTT pro zprostředkovatele publikování/odběru IBM MQ

Klient MQTT odeslal publikaci IBM MQ. Služba telemetrie (MQXR) převede publikování na zprávu produktu IBM MQ . Zpráva IBM MQ obsahuje tři části:

1. [MQMD](#)
2. [RFH2](#)
3. Zpráva

Vlastnosti MQMD jsou nastaveny na výchozí hodnoty, kromě případů uvedených v části [Tabulka 9](#) na stránce 126.

| <i>Tabulka 9. Nastavení pro vlastnosti MQMD</i> |            |                                                                                                                                                                             |
|-------------------------------------------------|------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Pole MQMD</b>                                | <b>Typ</b> | <b>Hodnota</b>                                                                                                                                                              |
| <b>Format</b>                                   | MQCHAR8    | MQFMT_RF_HEADER_2                                                                                                                                                           |
| <b>UserIdentifier</b>                           | MQCHAR12   | Nastavit na jednu z možností:<br>MqttClient.ClientIdentifier<br>MqttConnectOptions.UserName<br>ID uživatele nastavené administrátorem produktu IBM MQ pro kanál telemetrie. |
| <b>Priority</b>                                 | MLONG      | MQPRI_PRIORITY_AS_Q_DEF (Výchozí hodnota pro IBM MQ, která se liší od JMS , která má výchozí hodnotu 4.)                                                                    |
| <b>Persistence</b>                              | MLONG      | QoS=0→MQPER_NOT_PERSISTENT<br>QoS=1→MQPER_PERSISTENT<br>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 . Standardní typ zprávy JMS je jms - bytes zpráva. Aplikace má k dispozici další informace o záhlaví jako vlastnosti zprávy, viz [Vlastnosti zprávy](#).

Hodnoty RFH2 jsou nastaveny tak, jak je zobrazeno v části [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> |                   |                                                                                                                                        |
|--------------------------------------------------|-------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| <b>Vlastnost RFH2</b>                            | <b>Typ/Složka</b> | <b>Header</b>                                                                                                                          |
| Formát                                           | MQCHAR8           | MQFMT_NONE                                                                                                                             |
| ClientIdentifier                                 | mqtt/<br>clientId | Zkopírujte MqttClient.ClientIdentifier s délkou 1...23 bajtů.                                                                          |
| QoS                                              | mqtt/qos          | Zkopírujte QoS z příchozí zprávy MQTT .                                                                                                |
| ID zprávy                                        | mqtt/msgid        | Zkopírujte ID zprávy z příchozí zprávy MQTT , pokud je QoS 1 nebo 2.                                                                   |
| MQIsRetained                                     | mmps/Ret          | Nastavte, zda byla původní publikace produktu MQTT odeslána se sadou vlastností RETAIN a zpráva je přijata jako zachované publikování. |
| MQTopicString                                    | mmps/Top          | Téma, na které byla publikována zpráva MQTT .                                                                                          |

Informační obsah v publikaci MQTT je mapován na obsah zprávy produktu IBM MQ :

| <i>Tabulka 11. Způsob, jakým informační obsah v publikaci MQTT je mapován na obsah zprávy produktu IBM MQ</i> |                 |                                                           |
|---------------------------------------------------------------------------------------------------------------|-----------------|-----------------------------------------------------------|
| <b>Obsah zprávy</b>                                                                                           | <b>Typ</b>      | <b>Obsah zprávy IBM MQ</b>                                |
| Vyrovňovací paměť                                                                                             | MQBYTE <i>n</i> | Kopie bajtů z příchozí zprávy MQTT . Délka může být nula. |

## IBM MQ na klienta MQTT

Klient se přihlásil k odběru tématu publikování. An IBM MQ application has published to the topic, resulting in a publication being sent to the MQTT subscriber by the IBM MQ publish/subscribe broker. Alternativně, aplikace IBM MQ odeslala nevyžádanou zprávu přímo klientovi MQTT . Část [Tabulka 12](#) na stránce [127](#) popisuje, jak jsou záhlaví pevných zpráv nastavena ve zprávě zasílané klientovi MQTT . Jakákoli jiná data v záhlaví zprávy produktu IBM MQ nebo jakákoli jiná záhlaví budou zahozena. Data zprávy ve zprávě IBM MQ se odešlou jako informační obsah zprávy ve zprávě MQTT , bez změny. Zpráva MQTT se odešle klientovi MQTT pomocí služby telemetrie (MQXR).

| <i>Tabulka 12. Jak jsou záhlaví pevných zpráv nastavena ve zprávě IBM MQ odeslané klientovi MQTT</i> |             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|------------------------------------------------------------------------------------------------------|-------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| MQTT pole                                                                                            | Typ         | Hodnota                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| <b>DUP</b>                                                                                           | typ boolean | Nastavte, pokud je QoS = 1 nebo 2, a zpráva byla odeslána tomuto klientovi v předchozím přenosu a zpráva nebyla potvrzena po čase.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| <b>QoS</b>                                                                                           | celé číslo  | Způsob, jakým je hodnota QoS v odchozí publikaci ze zprostředkovatele publikování/odběru v produktu IBM MQ , závisí na příchozí publikaci. Závisí na tom, zda byla příchozí publikace odeslána z klienta produktu MQTT nebo z aplikace IBM MQ .<br><b>MQTT</b><br>Nižší hodnota QoS v příchozí publikaci a v QoS vyžádaná odběratelem.<br><b>IBM MQ</b><br>Nižší hodnota z QoS odvozená z příchozího publikování:<br>MQPER_NOT_PERSISTENT→QoS=0<br>MQPER_PERSISTENT→QoS=2<br>a server QoS požadovaný odběratelem. Je-li zpráva odeslána klientovi bez odběru, je produkt QoS nastaven standardně na 2. Klient může tuto hodnotu změnit tak, že se přihlásíte k odběru DEFAULT . QoS s jiným QoS. |
| <b>RETAIN</b>                                                                                        | typ boolean | Nastavte, má-li příchozí publikování zachovanou sadu vlastností.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |

Část [Tabulka 13](#) na stránce [127](#) popisuje, jak jsou záhlaví proměnných zpráv nastavena ve zprávě MQTT odeslané klientovi MQTT .

| <i>Tabulka 13. Jak se nastavují vlastnosti záhlaví proměnné produktu MQTT ve zprávě MQTT odeslané klientovi MQTT .</i> |         |                                                                                                                   |
|------------------------------------------------------------------------------------------------------------------------|---------|-------------------------------------------------------------------------------------------------------------------|
| MQTT pole                                                                                                              | Typ     | Hodnota                                                                                                           |
| <b>Topic name</b>                                                                                                      | Řetězec | Řetězec tématu, se kterým byla zpráva publikována.                                                                |
| <b>Message ID</b>                                                                                                      | Řetězec | Poslední 2 bajty MQMD . MsgId vlastnost publikace, když je umístěna v produktu SYSTEM . MQTT . TRANSMIT . QUEUE . |
| <b>Payload</b>                                                                                                         | byte [] | Přímá kopie bajtů z příchozích publikování do zprostředkovatele publikování a odběru. Délka může být nula.        |

Windows

Linux

AIX

## MQTT bezstavové a stavové relace

Klienti produktu MQTT mohou vytvořit stavovou relaci se správcem front. Když se stavový klient MQTT odpojí, udržuje správce front odběry vytvořené klientem a zprávy v době letu. Když se klient znovu připojí,

vyřeší se to jako-letová zpráva. Odešle všechny zprávy, které jsou ve frontě pro doručení, a přijímá zprávy publikované pro odběry, zatímco byla odpojena.

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 žádné publikace čekající na doručení. Když se klient připojí, určuje, zda má být spuštěn s čistou relací, nebo zda má pokračovat v existující relaci; viz [Vyčistit relace](#).

Pokud klient obnoví existující relaci, bude pokračovat tak, jako by nedošlo k přerušení spojení. Publikace čekající na doručení jsou odeslány klientovi a všechny přenosy zpráv, které nebyly potvrzeny, jsou dokončeny. Když se klient v trvalé relaci odpojí od služby telemetrie (MQXR), všechny odběry vytvořené klientem zůstanou. Publikace pro odběry se posílají na klienta, když se znovu připojí. Pokud se znovu připojí, aniž by došlo k obnovení původní relace, budou publikace zrušeny prostřednictvím služby telemetrie (MQXR).

Informace o stavu relace jsou uloženy správcem front ve frontě `SYSTEM.MQTT.PERSISTENT.STATE`.

Administrátor produktu IBM MQ může relaci odpojit a vymazat.

Windows

Linux

AIX

## Není-li klient MQTT připojen

Není-li klient připojen, může správce front nadále přijímat publikace jejím jménem. Jsou předány klientovi, jakmile se znovu připojí. Klient může vytvořit "Poslední vůli a testament", kterou správce front publikuje v zastoupení klienta, pokud se klient neočekávaně odpojí.

Chcete-li být upozorněni, když se klient neočekávaně odpojí, můžete registrovat poslední publikaci a sestavu testament; viz [Poslední zpráva a zveřejnění testatu](#). Je odeslán službou telemetrie (MQXR), pokud zjistí, že připojení k klientovi bylo přerušeno bez požadavku klienta na něj.

Klient může kdykoli publikovat zachované publikování; viz téma [Zachovaná publikování a klienti MQTT](#). Nový odběr tématu může požádat o odeslání jakékoliv zachované publikace přidružené k tématu. Vytvoříte-li poslední vůli a testament jako zachované publikování, můžete ji použít k monitorování stavu klienta.

Klient například publikuje zachované publikování, jakmile se připojí, bude reklamovat jeho dostupnost. Současně vytvoří zachovanou poslední publikaci a závěť zveřejněná, která oznamuje její nedostupnost. Kromě toho, těsně před tím, než se naplánuje odpojení, publikuje svou nedostupnost jako zachované publikování. Chcete-li zjistit, zda je klient dostupný, přihlásíte se k odběru tématu zachovaného publikování. Vždy obdržíte jednu ze tří publikací.

Má-li klient přijímat zprávy publikovaná, když je odpojen, pak znovu připojte klienta k jeho předchozí relaci; viz ["MQTT bezstavové a stavové relace"](#) na stránce 127. 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 publikování mezi klienty MQTT a aplikacemi IBM MQ je volně vázaný. Publikování mohou pocházet buď z klienta produktu MQTT, nebo z aplikace IBM MQ, a bez nastaveného pořadí. Vydavatelé a odběratelé jsou volně spřaženy. Komunikují se s sebou nepřímo prostřednictvím publikací a odběrů. Zprávy můžete také odesílat přímo klientovi produktu MQTT z aplikace IBM MQ.

Klienti MQTT a IBM MQ jsou volně přiřazeny ve dvou smyslech:

1. Vydavatelé a odběratelé jsou volně sdruženi přidružením publikování a odběru k tématu. Vydavatelé a odběratelé si obvykle nejsou vědomi adresy nebo identity jiného zdroje publikování nebo odběru.
2. Klienti MQTT publikují, odebírají, přijímají publikace a zpracovávají potvrzení doručení na oddělených podprocesech.

Aplikace klienta MQTT nečeká, dokud nebude publikace doručena. Aplikace předá zprávu klientovi produktu MQTT a poté bude pokračovat ve vlastním podprocesu. Doručovací token se používá k synchronizaci aplikace s doručením publikace, viz téma [Tokeny doručení](#).



Po předání zprávy klientovi produktu MQTT má aplikace volbu čekání na token doručení. Místo čekání může klient poskytnout metodu zpětného volání, která je volána, když je publikace doručena do produktu IBM MQ. Může také ignorovat token doručení.

V závislosti na kvalitě služby přidružené ke zprávě se token doručení vrátí okamžitě do metody zpětného volání nebo pravděpodobně po určité velké době. Token doručení může být vrácen i poté, co byl klient odpojen a znovu připojen. Je-li kvalita služby *fire and forget*, je doručovací token okamžitě vrácen. 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 doručovány do metody zpětného volání `messageArrived`. Produkt `messageArrived` se spouští v jiném podprocesu než hlavní aplikace.

## Odesílání zpráv přímo klientovi produktu MQTT

Můžete odeslat zprávu konkrétnímu klientovi produktu MQTT jedním ze dvou způsobů.

1. Aplikace produktu IBM MQ může odeslat zprávu přímo klientovi produktu MQTT bez odběru; viz téma [Odesílání zprávy přímo klientovi](#).
2. Alternativním přístupem je použití konvence pojmenování `ClientIdentifier`. Učinit všechny MQTT odběratele vytvářet odběry pomocí jejich jedinečného identifikátoru `ClientIdentifier` jako tématu. Publikujte na `ClientIdentifier`. Publikování je odesláno na klienta, který je přihlášen 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 nelze pečlivě kontrolovat. Můžete použít VPN k zabezpečení připojení ze zařízení produktu MQTT ke službě telemetrie (MQXR). Produkt MQ Telemetry poskytuje dva další mechanismy zabezpečení, TLS a JAAS.

TLS se především používá k šifrování komunikace mezi zařízením a kanálem telemetrie a k ověření, že se zařízení připojuje ke správnému serveru; viz téma [Ověření kanálu telemetrie pomocí TLS](#). TLS můžete také použít ke kontrole toho, zda je klientské zařízení povoleno připojit se k serveru; viz téma [Ověření klienta MQTT pomocí TLS](#).

JAAS se v zásadě používá ke kontrole, zda má uživatel zařízení povoleno používat serverovou aplikaci, viz téma [Ověření klienta MQTT pomocí hesla](#). JAAS lze použít spolu s protokolem LDAP ke kontrole hesla pomocí adresáře jednotného přihlášení.

TLS a JAAS lze použít ve spojení k poskytnutí dvoufaktorové autentizace. Šifry použité TLS můžete omezit na šifry, které vyhovují standardům FIPS.

S alespoň desítkami tisíc uživatelů není vždy praktické poskytovat jednotlivé profily zabezpečení. Stejně tak není vždy praktické používat profily k autorizaci jednotlivých uživatelů pro přístup k objektům produktu IBM MQ. Místo toho seskupte uživatele do tříd za účelem autorizace publikování a odběru témat a odesílání publikací na klienty.

Nakonfigurujte každý kanál telemetrie, aby mapoval klienty na společná ID uživatele klienta. Použijte obecné ID uživatele pro každého klienta, který se připojuje k určitému kanálu; viz téma [Identita a autorizace klienta produktu MQTT](#).

Autorizace skupin uživatelů nekompromituje ověření jednotlivých jednotlivců. Každý jednotlivý uživatel může být ověřen na klientu nebo na serveru pomocí jména `Jméno uživatele` a `Heslo`a poté na serveru autorizován pomocí společného ID uživatele.

Windows

Linux

AIX

## MQ Telemetry globalizace

Informační obsah zprávy v protokolu MQTT v3 je kódován jako bajtové pole. Obecně řečeno, aplikace zpracovávající text vytváří informační obsah zprávy v produktu UTF-8. Kanál telemetrie popisuje

informační obsah zprávy jako UTF-8, ale neprovádí žádné konverze kódové stránky. Řetězec tématu publikování musí být UTF-8.

Aplikace je zodpovědná za převod abecedních dat na správnou kódovou stránku a numerických dat do správného kódování čísel.

Klient MQTT Java má vhodnou metodu `MqttMessage.toString`. Metoda považuje informační obsah zprávy za kódovaný ve výchozí znakové sadě lokální platformy, která je obecně UTF-8. Převádí informační obsah na řetězec Java. Java má řetězcovou metodu, `getBytes`, která převádí řetězec na bajtové pole kódovaný pomocí výchozí znakové sady lokální platformy. Dva programy MQTT Java, které si vyměňují text v informačním obsahu zpráv, mezi platformami se stejnou výchozí znakovou sadou tak snadno a účinně v produktu UTF-8.

Není-li výchozí znaková sada jedné z platform UTF-8, pak aplikace musí vytvořit konvenci pro výměnu zpráv. Příklad: Vydavatel určuje převod z řetězce na 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 produktu MQTT jako produkt UTF-8. Nastavuje `MQMD.CodedCharSetId` do UTF-8a `RFH2.CodedCharSetId` do `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, ani `MQGET`.

Windows

Linux

AIX

## Výkon a rozšiřitelnost produktu MQ Telemetry

Při správě velkého počtu klientů a zlepšení rozšiřitelnosti produktu MQ Telemetry vezměte v úvahu následující faktory.

### Plánování kapacity

Informace o sestavách výkonu pro produkt MQ Telemetry naleznete v tématu [Dokumenty výkonu produktu MQ](#).

### Připojení

Náklady spojené s připojeními zahrnují

- Náklady na vytvoření připojení samotného, pokud jde o využití procesoru a čas.
- Náklady na síť.
- Paměť použitá při udržování otevřeného připojení, ale nevyužívá jej.

Když klienti zůstanou připojeni, vzniknou další zatížení. Je-li připojení udržováno otevřené, toky TCP/IP a zprávy produktu MQTT používají síť k ověření, zda je připojení stále v provozu. Navíc se paměť používá na serveru pro každé připojení klienta, které je stále otevřené.

Pokud odesíláte zprávy více než jednu minutu, nechte otevřené připojení, abyste se vyhnuli nákladům na zahájení nového připojení. Pokud odesíláte zprávy méně než jednu každých 10 až 15 minut, zvažte zrušení připojení, abyste se vyhnuli nákladům na jejich ponechání. Možná budete chtít ponechat spojení TLS otevřené, ale nečinné, po delší období, protože je to pro nastavení dražší.

Kromě toho zvažte možnosti klienta. Existuje-li na straně klienta služba uložení a předání, můžete zanechat dávku zpráv a zrušit připojení mezi odesláním dávek. Pokud je však klient odpojen, pak není možné, aby klient přijal zprávu ze serveru. Proto má účel vaší žádosti vliv na rozhodnutí.

Pokud má váš systém jeden klient, který posílá mnoho zpráv, například přenosy souborů, nečekejte na odpověď serveru na zprávu. Místo toho odešlete všechny zprávy a zkontrolujte, zda byly všechny přijaty všechny zprávy. Případně použijte [QoS \(Quality of Service\) \(QoS\)](#).

Můžete měnit QoS pomocí zpráv, doručovat nedůležité zprávy pomocí QoS 0 a důležitých zpráv pomocí QoS z 2. Propustnost zpráv může být přibližně dvakrát vyšší než QoS s hodnotou 0, než jakou má QoS 2.

## Konvence pojmenování

Pokud navrhujete svou aplikaci pro mnoho klientů, implementujte efektivní konvenci pojmenování. Chcete-li namapovat každého klienta na správnou hodnotu `ClientIdentifier`, použijte `ClientIdentifier`, který má význam. Dobrá konvencí pojmenování usnadňuje administrátorovi práci na tom, kteří klienti jsou spuštěni. Konvence pojmenování pomáhá administrátorovi filtrovat dlouhý seznam klientů v produktu IBM MQ Explorer a pomáhá při určování problémů; viz [Identifikátor klienta](#).

## Propustnost

Délka názvů témat ovlivňuje počet bajtů, které tečí po síti. Při publikování nebo odběru zprávy může být důležité, aby byl počet bajtů ve zprávě důležitý. Proto omezte počet znaků v názvu tématu. Když se klient MQTT přihlásí k odběru tématu IBM MQ, zobrazí se 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 produktu MQTT se připojuje ke vzdálenému správci front produktu IBM MQ. Existují dvě základní metody pro aplikaci produktu IBM MQ k odesílání zpráv klientovi MQTT: nastavte výchozí přenosovou frontu na `SYSTEM.MQTT.TRANSMIT.QUEUE` nebo použijte aliasy správce front. Definujte výchozí přenosovou frontu správce front, pokud existují velké počty klientů MQTT. Použití výchozího nastavení přenosové fronty zjednodušuje pracovní výkon administrace. Další informace naleznete v tématu [Konfigurace distribuovaných front pro odesílání zpráv klientům MQTT](#).

## Zlepšování rozšiřitelnosti vyloučením odběrů.

Když se klient MQTT V3 přihlásí k odběru tématu, vytvoří se odběr ze služby telemetrie (MQXR) v produktu IBM MQ. Odběr směřuje publikace pro klienta do produktu `SYSTEM.MQTT.TRANSMIT.QUEUE`. Název vzdáleného správce front v záhlaví přenosu každé publikace je nastaven na hodnotu `ClientIdentifier` klienta produktu MQTT, který vytvořil odběr. Pokud existuje mnoho klientů, z nichž každý provádí své vlastní odběry, je výsledkem mnoha proxy odběrů udržovaných v rámci klastru nebo hierarchie publikování a odběru IBM MQ. Informace o tom, jak nepoužívat `publish/subscribe`, ale místo toho používat řešení založené na bodech, viz [Odeslání zprávy přímo klientovi](#).

## Správa velkého počtu klientů

Chcete-li podporovat mnoho souběžně připojených klientů, zvyšte velikost paměti dostupnou 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 [Telemetry \(MQXR\) v produktu Windows](#) nebo [Konfigurační adresář služby Telemetry v systému Linux](#).
2. Postupujte podle pokynů v souboru; halda 1 GB je dostatečná pro 50 000 současně připojených klientů.

```
Heap sizing options - uncomment the following lines to set the heap to 1G
#-Xmx1024m
#-Xms1024m
```

3. Přidejte další argumenty příkazového řádku pro předání do prostředí JVM se spuštěnou službou telemetrie (MQXR) v souboru `java.properties`; viz téma [Předávání parametrů prostředí JVM do služby telemetrie \(MQXR\)](#).

Chcete-li zvýšit počet deskriptorů otevřených souborů v systému Linux, přidejte následující řádky do `/etc/security/limits.conf`/a znovu se přihlaste.

```
@mqm soft nofile 65000
@mqm hard nofile 65000
```

Každý soket vyžaduje jeden popisovač souboru. Služba telemetrie vyžaduje některé další deskriptory souboru, takže tento počet musí být větší než požadovaný počet otevřených soketů.

Správce front používá pro každý netrvalý odběr popisovač objektu. Pro podporu mnoha aktivních netrvalých odběrů se zvyšuje maximální počet aktivních manipulátorů ve správci front, například:

```
echo ALTER QMGR MAXHANDS(99999999) | runmqsc qMgrName
```

Obrázek 49. Změnit maximální počet popisovačů v systému Windows

```
echo "ALTER QMGR MAXHANDS(99999999)" | runmqsc qMgrName
```

Obrázek 50. Změnit maximální počet popisovačů v systému Linux

## Další aspekty

Při plánování systémových požadavků vezměte v úvahu dobu, po kterou trvá restart systému. Plánovaná prostoj může mít důsledky pro počet zpráv, které čekají ve frontě, čekají na zpracování. Konfigurujte systém tak, aby zprávy mohly být úspěšně zpracovány v přijatelném čase. Přezkoumejte diskovou paměť, paměť a výpočetní výkon. U některých klientských aplikací může být možné zahodit zprávy, když se klient znovu připojí. Chcete-li zprávy zahodit, nastavte `CleanSession` v parametrech připojení klienta, viz [Vyčistit relace](#). Případně můžete publikovat a odebrat nejlepší kvalitu práce služby `0v` klientovi MQTT ; viz [QoS \(Quality of Service\)](#). Při odesílání zpráv z produktu IBM MQ použijte netrvalé zprávy. Zprávy s těmito vlastnostmi kvality služby se neobnoví, když se systém nebo připojení restartuje.

Windows

Linux

AIX

## Zařízení podporovaná produktem MQ Telemetry

Klienti produktu MQTT mohou pracovat na řadě zařízení, od senzorů a ovládacích prvků, k ručníkům a systémům vozidel.

Klienti produktu MQTT jsou malí a spouštějí se na zařízeních s omezeným množstvím paměti a nízkou výpočetní silou. Produkt MQTT protocol je spolehlivý a má malá záhlaví, která vyhovuje sítím s omezením šířky pásma, vysokými náklady a intermitentní dostupností.

Produkt MQ Telemetry komunikuje s telemetrickým zařízením 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 k sestavení nativních aplikací pro zařízení (například pro systémy Android, OS X, Linux nebo Windows ). Aplikace, které používají tuto knihovnu klienta, mohou být spuštěny ve všech variantách produktu Java z nejmenšího CLDC (Připojené konfigurace připojeného zařízení) /MIDP (Mobile Information Device Profile) prostřednictvím CDC (Connected Device Configuration) /Foundation, J2SE ( Java Platform, Standard Edition) a J2EE ( Java Platform, Enterprise Edition). Je také podporována přizpůsobená knihovna tříd produktu IBM jclRM . Platforma Java ME se obvykle používá na malých zařízeních, jako jsou ovládací prvky, senzory, mobilní telefony a jiná zařízení. Platforma Java SE je obecně instalována na vyšších koncových zařízeních, jako jsou stolní počítače a servery.
- Objekt *MQTT client for C*, který se používá k sestavování nativních aplikací pro zařízení (například iOS, OS X, Linux nebo Windows ). Tato klientská knihovna poskytuje referenční implementaci C společně s předem sestavovaným nativním klientem pro systémy Windows a Linux . Referenční implementace jazyka C umožňuje portovat produkt MQTT do široké škály zařízení a platform. Některé systémy Windows na systémech Intel, včetně systémů Windows 7, RedHat, Ubuntu a některých systémů

Linux na platformách ARM, jako je například společnost Eurotech Viper, implementují verze produktu Linux, které spouštějí klienta C, ale produkt IBM neposkytuje podporu služeb pro platformy. Pokud máte v úmyslu volat centrum podpory produktu IBM, je třeba při práci s klientem na podporované platformě reprodukovat problémy s klientem.

- Objekt *MQTT client for Java*, který se používá pro sestavení webových aplikací na bázi prohlížeče.

Knihovny klienta produktu MQTT jsou volně dostupné z prostředí Eclipse Paho a MQTT.org. Viz [IBM MQ Telemetry Transport ukázkových programů](#).

## Zabezpečení v produktu IBM MQ

---

V produktu IBM MQ existuje několik metod poskytování zabezpečení: rozhraní služby autorizace, uživatelem napsané nebo třetí strana, uživatelské procedury kanálu; zabezpečení kanálu s použitím zabezpečení TLS (Transport Layer Security), záznamy ověřování kanálu a zabezpečení zpráv.

### Rozhraní autorizační služby

Autorizace pro použití volání MQI, příkazů a přístupu k objektům je poskytována správcem **Object Authority Manager** (OAM), který je standardně povolen. Přístup k entitám IBM MQ je řízen pomocí skupin uživatelů produktu IBM MQ a OAM. Administrátoři mohou podle potřeby udělovat nebo odvolávat autorizace pomocí rozhraní příkazového řádku.

Další informace o vytváření komponent autorizační služby najdete v tématu [Nastavení zabezpečení v systémech AIX, Linux, and Windows](#).

### Uživatelské procedury zapsané uživatelem nebo kanály třetích stran

Kanály mohou používat uživatelské procedury zapsané uživatelem nebo kanály třetích stran. Další informace naleznete v tématu [Programy ukončení kanálů pro kanály systému zpráv](#).

### Zabezpečení kanálu pomocí protokolu TLS

Protokol Transport Layer Security (TLS) poskytuje standardní zabezpečení kanálu pro průmysl, s ochranou proti odposlouchávání, falšování a ztělesnění.

TLS používá veřejný klíč a symetrické techniky k zajištění utajení a integrity zpráv a vzájemného ověření.

Úplný přehled zabezpečení v produktu IBM MQ, včetně podrobných informací o TLS, najdete v tématu [Zabezpečení](#). Přehled TLS, včetně ukazatelů na příkazy popsané v tomto oddílu, najdete v tématu [Šifrovací protokoly zabezpečení: TLS](#).

### Záznamy ověření kanálu

Záznamy ověření kanálu umožňují vykonávat přesnou kontrolu nad přístupem udělenou k připojovacím systémům na úrovni kanálu. Další informace naleznete v tématu [Záznamy ověřování kanálu](#).

### Zabezpečení zpráv

Použijte produkt Advanced Message Security, který je samostatně nainstalovanou a licencovanou komponentou produktu IBM MQ, chcete-li poskytovat kryptografickou ochranu zprávám odeslaným a přijímat pomocí produktu IBM MQ. Viz [Advanced Message Security](#).

### Související úlohy

[Zabezpečení](#)

[Plánování bezpečnostních požadavků](#)

## Podpora TLS spravovaného klienta IBM MQ.NET

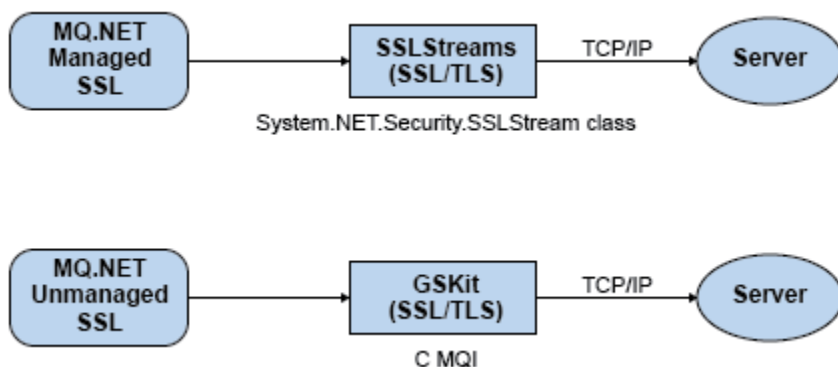
Plně spravovaný klient produktu IBM MQ.NET poskytuje podporu TLS (Transport Layer Security), která je založena na sadě SSLStreams produktu Microsoft.NET . To se liší od ostatních klientů IBM MQ , které jsou založeny na sadě GSKit.

Můžete vyvíjet aplikace produktu IBM MQ.NET pro spuštění ve spravovaném režimu nebo v nespravovaném režimu.

- Ve spravovaném režimu pracují aplikace produktu .NET v prostředí .NET CLR (Common Language Runtime) bez jakýchkoli vyvolání mezi platformami, jako je např. vyvolání rozhraní C MQI.
- V nespravovaném režimu je rozhraní C MQI vyvoláno pro základní operace MQI. Rozhraní v nespravovaném režimu se v podstatě skládá z tříd modulů wrapper produktu .NET v horní části rozhraní C MQI.

Spravovaný klient IBM MQ.NET používá knihovny Microsoft.NET Framework k implementaci protokolů zabezpečených soketů TLS. Systém.NET.Security.SSLStream třída z Microsoft se používá pro implementaci zabezpečení (TLS) v IBM MQ.NET.

Nespravovaný režim klienta IBM MQ.NET již podporuje funkci TLS, která je založena na rozhraní C MQI (a sadě GSKit). To znamená, že operace TLS jsou zpracovávány prostřednictvím rozhraní C MQI. V takovém případě sada GSKit implementuje protokoly zabezpečeného soketu TLS.



Obrázek 51. Spravované a nespravované porovnání TLS produktu IBM MQ.NET

Následující tabulka shrnuje rozdíly mezi spravovanými a nespravovanými implementacemi:

| Tabulka 14. Rozdíly mezi správou a nespravovanými implementacemi |           |                                                                                                      |                                                                   |
|------------------------------------------------------------------|-----------|------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------|
| Režim                                                            | Protokoly | Implementace                                                                                         | Komentáře                                                         |
| IBM MQ.NET spravované SSL                                        | TLS       | Systém.NET.Security.SSLStream , třída<br>SSLStream třída pracuje jako proud přes připojený soket TCP | TLS 1.0<br>TLS 1.2 (pouze s rámcem Microsoft.NET Framework v4.5 ) |
| Nespravované SSL produktu IBM MQ.NET                             | TLS       | GSKIT a C-MQI                                                                                        | TLS zabezpečené soketové protokoly                                |

### Související pojmy

Podpora zabezpečení SSL (Secure Sockets Layer) a TLS (Transport Layer Security) pro produkt .NET

## Klienti a servery

Úvod do způsobu, jakým produkt IBM MQ podporuje konfigurace klient-server pro své aplikace.

*Klient IBM MQ MQI* je komponenta, která umožňuje aplikaci spuštěné v systému vydávat volání MQI ke správci front spuštěnému v jiném systému. Výstup z volání se odešle zpět klientovi, který jej předá zpět aplikaci.

IBM MQ *server* je správce front, který poskytuje služby řazení do fronty jednomu nebo více klientům. Všechny objekty produktu IBM MQ, například fronty, existují pouze v počítači se správcem front (počítač serveru IBM MQ) a nikoli na straně klienta. 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 pro každého klienta vyhrazenou komunikační linku. Další informace o vytváření kanálů pro klienty a servery najdete v tématu [Konfigurace distribuovaných front](#).

Informace o klientech obecně viz [“Přehled produktu IBM MQ MQI clients”](#) na stránce 135.

## Aplikace produktu IBM MQ v prostředí klient-server

Při propojení se serverem mohou klientské aplikace klienta IBM MQ vydávat většinu volání MQI stejným způsobem jako lokální aplikace. Klientská aplikace vydá volání MQCONN pro připojení k určenému správci front. Všechny další volání MQI, které specifikují popisovač připojení vrácený z požadavku na připojení, jsou pak zpracovány tímto správcem front.

Musíte propojit své aplikace s příslušnými knihovnamy klienta. Viz téma [Sestavování aplikací pro produkt IBM MQ MQI clients](#).

### Související pojmy

“Správa a podpora transakcí” na stránce 141

Úvod do správy transakcí a to, jak produkt IBM MQ podporuje transakce.

“Rozšíření zařízení správce front” na stránce 143

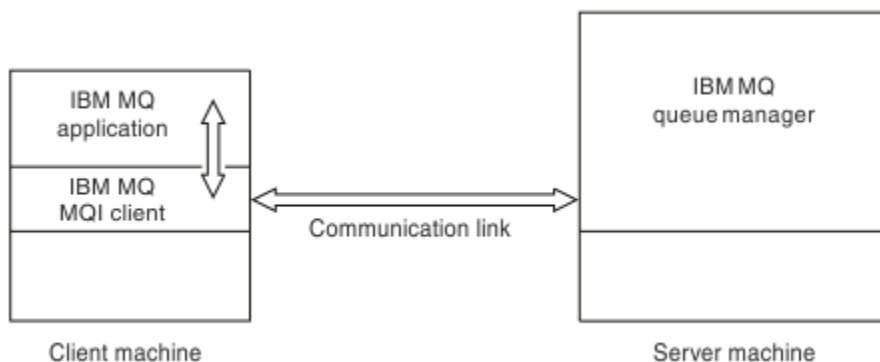
Zařízení správce front lze rozšířit pomocí uživatelských procedur, uživatelských procedur rozhraní API nebo instalovatelných služeb.

## Přehled produktu IBM MQ MQI clients

IBM MQ MQI client je komponenta produktu IBM MQ, kterou lze instalovat na systém, na kterém není spuštěn žádný správce front.

Pomocí produktu IBM MQ MQI client lze aplikaci spuštěnou na stejném systému jako klient připojit ke správci front, který je spuštěn v jiném systému. Aplikace může vydat volání MQI pro daného správce front. Taková aplikace se nazývá aplikace IBM MQ MQI client a správce front se nazývá *správce front serveru*.

Aplikace produktu IBM MQ MQI client a správce front serveru spolu navzájem komunikují prostřednictvím *kanálu MQI*. Kanál MQI se spustí, když klientská aplikace vydá volání **MQCONN** nebo **MQCONNX** k připojení ke správci front a ukončí se, když klientská aplikace vyše volání **MQDISC** k odpojení od správce front. Vstupní parametry toku volání MQI v jednom směru v kanálu MQI a výstupní parametry se ubíjí směrem opačným směrem.



Obrázek 52. Propojení mezi klientem a serverem

Mohou být použity následující platformy. Kombinace závisí na tom, jaký produkt IBM MQ používáte a jak jsou popsány v tématu [“Podpora platformy pro klienty IBM MQ”](#) na stránce 138.

### IBM MQ MQI client

AIX and Linux  
Windows  
IBM i

### Server IBM MQ

AIX and Linux  
Windows  
IBM i  
z/OS

Modul MQI je k dispozici aplikacím spuštěným na platformě klienta; fronty a další objekty produktu IBM MQ se nacházejí ve správci front, kterého jste nainstalovali na server.

Aplikace, kterou chcete spustit v prostředí produktu IBM MQ MQI client, musí být nejprve propojena s příslušnou knihovnou klienta. Když aplikace odešle volání MQI, přesměruje IBM MQ MQI client požadavek na správce front, kde je zpracován a odkud je odpověď odeslána zpět do IBM MQ MQI client.

Propojení mezi aplikací a produktem IBM MQ MQI client se vytváří dynamicky za běhu.

Klientské aplikace můžete také vyvíjet 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 použít na následujících platformách:

-  IBM i
-  AIX
-  Linux
-  Windows

Použití Java a JMS zde není popsáno. Úplné podrobnosti o instalaci, konfiguraci a použití produktů IBM MQ classes for Java a IBM MQ classes for JMS naleznete v části [Použití produktu IBM MQ classes for Java](#) a [Použití produktu IBM MQ classes for JMS](#).

### Související pojmy

[“Proč používat klienty IBM MQ ?”](#) na stránce 136

Použití klientů IBM MQ je efektivním způsobem implementace systému zpráv IBM MQ a front.

[“Jak nastavit IBM MQ MQI client”](#) na stránce 138

Postupujte podle těchto pokynů, chcete-li nastavit klienta.

[“Co je rozšířený transakční klient?”](#) na stránce 139

Rozšířený transakční klient produktu IBM MQ může aktualizovat prostředky spravované jiným správcem prostředků pod kontrolou externího správce transakcí.

[“Jak se klient připojuje k serveru”](#) na stránce 140

Klient se připojuje k serveru pomocí MQCONN nebo MQCONNX a komunikuje prostřednictvím kanálu.

## Proč používat klienty IBM MQ ?

Použití klientů IBM MQ je efektivním způsobem implementace systému zpráv IBM MQ a front.

Můžete mít aplikaci, která používá rozhraní MQI spuštěné na jednom počítači, a správce front spuštěného na jiném počítači (fyzického nebo virtuálního). Výhody tohoto provedení jsou:

- Na klientském počítači není potřeba úplné implementace produktu IBM MQ .
- Hardwarové požadavky na systém klienta se snižují.
- Požadavky na správu systému se snižují.



- Aplikace produktu IBM MQ spuštěná v klientu se může připojit k více správcům front v různých systémech.
- Mohou být použity alternativní kanály používající různé přenosové protokoly.

### Související odkazy

“Jaké aplikace jsou spuštěny na serveru IBM MQ MQI client?” na stránce 137

Úplné rozhraní MQI je podporováno v prostředí klienta.

“Podpora platformy pro klienty IBM MQ” na stránce 138

IBM MQ na všech podporovaných platformách serverů přijímá klientská připojení z IBM MQ MQI clients na řadě platform.

### Jaké aplikace jsou spuštěny na serveru IBM MQ MQI client?

Úplné rozhraní MQI je podporováno v prostředí klienta.

To umožňuje konfiguraci téměř všech aplikací produktu IBM MQ pro spuštění v systému IBM MQ MQI client propojením aplikace na serveru IBM MQ MQI client s knihovnou MQIC a nikoli s knihovnou MQI. Výjimky jsou:

- MQGET se signálem
- Aplikace, která potřebuje koordinaci bodu synchronizace s jinými správci prostředků, musí používat rozšířeného transakčního klienta.

Je-li povoleno čtení napřed, chcete-li zlepšit výkon přechodných zpráv, ne všechny volby MQGET jsou k dispozici. V tabulce jsou uvedeny volby, které jsou povoleny, a informace o tom, zda je lze změnit mezi voláními MQGET.

| Tabulka 15. Volby MQGET povolené, je-li povoleno čtení napřed |                                                                                                                                                                                             |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                                                                                                                                     |
|---------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------|
| Hodnoty                                                       | Povoleno, je-li dopředné čtení povoleno a lze je měnit mezi voláními MQGET                                                                                                                  | Povoleno, je-li dopředné čtení povoleno, ale nelze je měnit mezi voláními MQGET <sup>1</sup>                                                                                                                                                                                                                                                                                                                                                                                                                                                   | Volby MQGET, které nejsou povoleny, je-li povoleno čtení napřed, <sup>2</sup>                                                                       |
| Hodnoty MQGET<br>MD                                           | MsgId <sup>3</sup><br>CorrelId <sup>3</sup>                                                                                                                                                 | Kódování<br>CodedCharSetId                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                     |
| Volby MQGMO<br>MQGET                                          | MQGMO_WAIT<br>MQGMO_NO_WAIT<br>FUNKCE<br>MQGMO_FAIL_IF QUIESCING<br>MQGMOPRE_FIRST <sup>4</sup><br>MQGMOROWSE_NEXT <sup>4</sup><br>ZPRÁVA<br>MQGMO_BROWSE_MESSAGE_UNDER_CURSOR <sup>4</sup> | MQGMO_SYNCPOINT_IF_PERSISTENT<br>MQGMO_NO_SYNCPOINT<br>SOUBOR<br>MQGMO_ACCEPT_TRUNCATED_MSG<br>MQGMO_CONVERT<br>MQGMO_LOGICAL_ORDER<br>ZPRÁVA MQGMO_COMPLETE_MSG<br>MQGMO_ALL_MSGS_AVAILABLE<br>DOSTUPNÉ<br>MQGMO_ALL_SEGMENTS_AVAILABLE<br>POPISOVAČ MQGMO_MARK_BROWSE_HANDLE<br>MQGMO_MARKER_BROWSE_CO_OP<br>MQGMO_UNMARK_BROWSE_CO_OP<br>POPISOVAČ MQGMO_UNMARK_BROWSE_HANDLE<br>MQGMO_UNMARKED_BROWSE_MSG<br>,<br>MQGMO_PROPERTIES_FORCE_MQR<br>FH2<br>MQGMO_NO_PROPERTIES<br>MQGMO_PROPERTIES_IN_HANDLE<br>MQGMO_PROPERTIES_COMPATIBILITY | SIGNÁL MQGMO_SET_DATA<br>MQGMO_SYNCPOINT<br>PŘESKOČENO MQGMO_MARK_VYDANI<br>MQGMO_MSG_UNDER_CURSOR <sup>4</sup><br>MQGMOVÝ_ZÁMEK<br>MQGMO_ODEMKNOUT |
| 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 produktu IBM MQ MQI client se může připojit k více než jednomu správci front souběžně nebo může použít název správce front s hvězdičkou (\*) v rámci volání MQCONN nebo MQCONNX (viz příklady v tématu [Připojení aplikací IBM MQ MQI client ke správcům front](#)).

### **Podpora platformy pro klienty IBM MQ**

IBM MQ na všech podporovaných platformách serverů přijímá klientská připojení z IBM MQ MQI clients na řadě platform.

Produkt IBM MQ instalovaný jako *Základní produkt a server* na všech podporovaných platformách serverů může přijímat připojení z produktu IBM MQ MQI clients na následujících platformách:




-  IBM i
-  AIX
-  Linux
-  Windows

Připojení klienta závisí na rozdílech v identifikátoru kódované znakové sady (CCSID) a v komunikačním protokolu.

### **Jak nastavit IBM MQ MQI client**

Postupujte podle těchto pokynů, chcete-li nastavit klienta.

Chcete-li nastavit produkt IBM MQ MQI client, musíte mít již nainstalován a funkční server IBM MQ, ke kterému se bude klient připojovat. Postup nastavení klienta je:

1. Zkontrolujte, zda máte vhodnou platformu pro IBM MQ MQI client a že hardware a software splňuje požadavky. Podpora platformy je popsána v tématu [“Podpora platformy pro klienty IBM MQ”](#) na stránce 138.
2. Rozhodněte se, jak budete instalovat produkt IBM MQ na pracovní stanici klienta, a poté postupujte podle pokynů pro vaši konkrétní kombinaci platformy klienta a serveru. Instalace je popsána v následujících tématech:
  -  [Instalace klienta IBM MQ v systému AIX](#)
  -  [Instalace klienta IBM MQ v systému Linux](#)
  -  [Instalace klienta IBM MQ v systému Windows](#)
  -  [Instalace klienta IBM MQ v systému IBM i](#)
3. Ujistěte se, že jsou vaše komunikační odkazy konfigurovány a připojeny. Konfigurace spojů komunikace je popsána v tématu [Konfigurace připojení mezi serverem a klientem](#).
4. Zkontrolujte, zda vaše instalace funguje správně. Prohlédněte si sekci verifikace instalační procedury pro platformu nebo platformy, které váš podnik používá.

5. Máte-li ověřené instalace produktu IBM MQ MQI client , zvažte, zda je třeba zabezpečit klienta. Zabezpečení klienta je popsáno v tématu [Nastavení zabezpečení produktu IBM MQ MQI client](#).
6. Nastavte kanály mezi serverem IBM MQ MQI client a serverem, které jsou vyžadovány aplikacemi produktu IBM MQ , které chcete spustit na klientovi. Nastavení kanálů je popsáno v tématu [Definování kanálů MQI](#). Používáte-li protokol TLS, je třeba zvážit několik dalších aspektů. Tyto aspekty jsou popsány v části [Určení, že kanál MQI používá TLS](#). K nastavení kanálů může být zapotřebí použít konfigurační soubor IBM MQ MQI client nebo proměnné prostředí IBM MQ . Proměnné prostředí IBM MQ jsou popsány v tématu [Použití proměnných prostředí IBM MQ](#).
7. Aplikace produktu IBM MQ jsou plně popsány v tématu [Vývoj aplikací](#).
8. Existují určité rozdíly v prostředí správce front, které je třeba zvážit při návrhu, sestavení a spuštění aplikací v prostředí produktu IBM MQ MQI client . Informace o těchto rozdílech viz:
  - [Použití rozhraní MQI \(Message Queue Interface\) v klientské aplikaci](#)
  - [Sestavování aplikací pro produkt IBM MQ MQI clients](#)
  - [Připojení aplikací IBM MQ MQI client ke správcům front](#)
  - [Řešení problémů s produktem IBM MQ MQI clients](#)

## Co je rozšířený transakční klient?

Rozšířený transakční klient produktu IBM MQ může aktualizovat prostředky spravované jiným správcem prostředků pod kontrolou externího správce transakcí.

Pokud nejste obeznámeni s koncepcemi správy transakcí, přečtěte si téma [“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 pracovní jednotce, kterou spravuje správce front, ke kterému je připojena. V rámci pracovní jednotky může klientská aplikace vkládat zprávy do front, které tento správce front vlastní, a zprávy z nich získávat. Klientská aplikace pak může použít volání **MQCMIT** k potvrzení pracovní jednotky nebo volání obslužného programu **MQBACK** k vrácení pracovní jednotky. V rámci stejné pracovní jednotky však aplikace klienta nemůže aktualizovat prostředky jiného správce prostředků, například tabulky databáze Db2 . Tento omezení se odstraní pomocí rozšířeného transakčního klienta IBM MQ .


Rozšířený transakční klient IBM MQ je IBM MQ MQI client s některými dalšími funkcemi. Při použití této funkce může klientská aplikace v rámci stejné jednotky práce provádět následující úlohy:

- Umístit zprávy do front, které jsou ve vlastnictví správce front, k němuž je připojen správce front, do front, do kterých je připojen, a zprávy z nich.
- Aktualizovat prostředky jiného správce prostředků, než je správce front IBM MQ

Tato jednotka práce musí být spravována externím správcem transakcí, který je spuštěn na stejném systému jako klientská aplikace. Pracovní jednotku nelze spravovat pomocí správce front, ke kterému je klientská aplikace připojena. To znamená, že správce front může vystupovat pouze jako správce prostředků, nikoli jako správce transakcí. To také znamená, že klientská aplikace může jednotku práce potvrdit nebo vrátit pomocí rozhraní API (Application Programming Interface) poskytovaného externím správcem transakcí. Klientská aplikace proto nemůže používat volání MQI, **MQBEGIN**, **MQCMIT** a **MQBACK**.

Externí správce transakcí komunikuje se správcem front jako správce prostředků s použitím téhož kanálu MQI, který je používán klientskou aplikací, která je připojena ke správcovi front. Pokud však v situaci zotavení po selhání nejsou spuštěny žádné aplikace, může správce transakcí použít vyhrazený kanál MQI k obnovení všech nedokončených jednotek práce, v nichž se správce front účastnil v době selhání.

V této sekci je IBM MQ MQI client , který nemá rozšířenou transakční funkci, označovaný jako základní klient IBM MQ . Z tohoto důvodu můžete zvážit, zda klient IBM MQ rozšířeného transakčního klienta se skládá z základního klienta IBM MQ s přidáním rozšířené funkce transakcí.

**Poznámka:**  Produkt IBM MQ MQI client v systému IBM i nepodporuje rozšířenou transakční funkci IBM MQ .

## Související odkazy

[“Podpora platformem pro rozšířené transakční klienty” na stránce 140](#)

Rozšířené transakční klienty jsou k dispozici pro všechny platformy Multiplatforms, které podporují základního klienta. Klienti nejsou k dispozici pro produkt z/OS.

### **Multi** Podpora platformem pro rozšířené transakční klienty

Rozšířené transakční klienty jsou k dispozici pro všechny platformy Multiplatforms, které podporují základního klienta. Klienti nejsou k dispozici pro produkt z/OS.

Klientská aplikace, která používá rozšířený transakční klient, se může připojit ke správci front pouze z následujících produktů IBM MQ 9.0 :

- ▶ **AIX** IBM MQ for AIX
- ▶ **IBM i** IBM MQ for IBM i
- ▶ **Linux** položky IBM MQ pro Linux
- ▶ **Windows** IBM MQ for Windows

▶ **z/OS** Although there are no extended transactional clients that run on z/OS, a client application that is using an extended transactional client can connect to a queue manager that runs on z/OS.

Požadavky na hardware a software pro rozšířeného transakčního klienta jsou pro každou platformu stejné jako požadavky na základního klienta produktu IBM MQ . Programovací jazyk je podporován rozšířeným transakčním klientem, pokud je podporován základním klientem produktu IBM MQ a správcem transakcí, který 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 klientském prostředí IBM MQ musí udržovat aktivní spojení mezi klientem a serverovou počítačem.

Připojení se provádí aplikací, která vydala volání MQCONN nebo MQCONNX. Klienti a servery komunikují prostřednictvím *kanálů MQI* nebo při použití sdílených konverzací komunikují konverzace mezi každou sdílenou instancí kanálu MQI. Je-li volání úspěšné, zůstane instance kanálu MQI nebo konverzace připojena, dokud aplikace neodešle volání MQDISC. Jedná se o případ pro každého správce front, ke kterému se aplikace potřebuje připojit.

### Související pojmy

[“Klient a správce front na stejném počítači” na stránce 140](#)

Můžete také spustit aplikaci v prostředí produktu IBM MQ MQI client , pokud má váš počítač také nainstalovaný správce front.

[“Klienti na různých platformách” na stránce 141](#)

Zde je další příklad serveru IBM MQ MQI client a systému serveru. V tomto příkladu komunikuje serverový stroj se třemi IBM MQ MQI clients na různých platformách.

[“Použití různých verzí softwaru klienta a serveru” na stránce 141](#)

Pokud používáte předchozí verze produktů IBM MQ , ujistěte se, že server podporuje převod kódu z CCSID vašeho klienta.

### **Klient a správce front na stejném počítači**

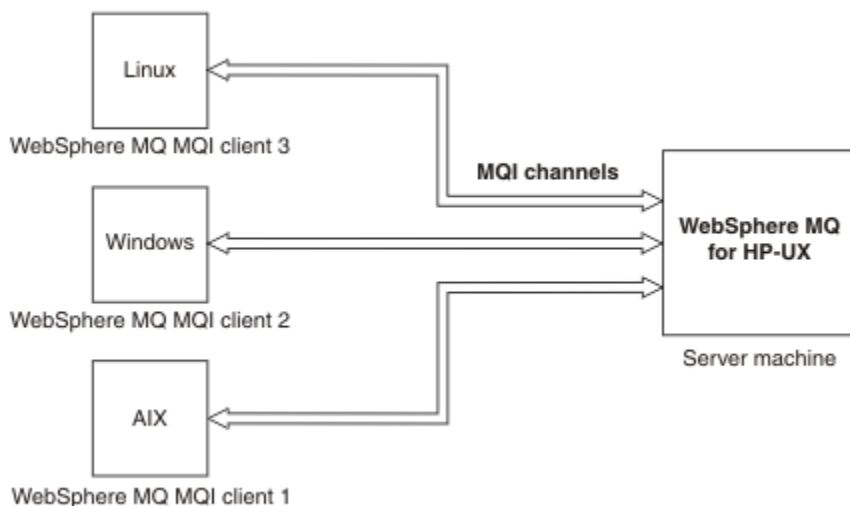
Můžete také spustit aplikaci v prostředí produktu IBM MQ MQI client , pokud má váš počítač také nainstalovaný správce front.

V této situaci máte možnost volby propojení s knihovnamy správce front nebo knihovnamy klienta, ale pamatujte, že pokud odkazujete na knihovny klienta, budete stále muset definovat připojení kanálu. To může být užitečné během vývojové fáze aplikace. Můžete testovat svůj program na svém vlastním

počítači, bez závislosti na ostatních a mít jistotu, že bude stále pracovat, když jej přesunete do nezávislého prostředí produktu IBM MQ MQI client .

### **Klienti na různých platformách**

Zde je další příklad serveru IBM MQ MQI client a systému serveru. V tomto příkladu komunikuje serverový stroj se třemi IBM MQ MQI clients na různých platformách.



Obrázek 53. Server IBM MQ připojený k klientům na různých platformách

Další složitější prostředí jsou možná. Klient produktu 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 podporovanou úroveň 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 to, jak produkt IBM MQ podporuje transakce.

*Správce prostředků* je počítačový subsystém, který vlastní a spravuje prostředky, ke kterým lze přistupovat a které jsou aktualizovány aplikacemi. Niže jsou uvedeny příklady správců prostředků:

- Správce front produktu 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 úspěšně dokončeny jako skupina, nebo se žádný z nich nedokončí. Důvodem tohoto druhu požadavku je to, že obchodní data by byla ponechána v nekonzistentním stavu, pokud by některé z těchto aktualizací byly úspěšně dokončeny, ale jiné nikoli.

Aktualizace prostředků, které jsou spravovány tímto způsobem, se mohou vyskytnout v rámci *pracovní jednotky* nebo *transakce*. Aplikační program může seskupovat sady aktualizací do pracovní jednotky.

Během pracovní jednotky aplikace vydá požadavek na správce prostředků, aby aktualizoval své prostředky. Jednotka práce se ukončí, když aplikace vydá požadavek na potvrzení všech aktualizací. Dokud nebudou aktualizace potvrzeny, nebude žádný z nich viditelný pro jiné aplikace, které přistupují ke stejným prostředkům. Jinak, pokud se aplikace rozhodne, že z nějakého důvodu nemůže dokončit pracovní jednotku, může vydat požadavek na vrácení všech aktualizací, které požadovala až k tomuto bodu. V takovém případě se žádná z aktualizací nikdy nestane viditelnou pro jiné aplikace. Tyto aktualizace jsou obvykle logicky související a všechny musí být úspěšné, aby byla zachována integrita dat. Pokud dojde k aktualizaci jedné aktualizace, zatímco jiná selže, integrita dat se ztratí.

Když se jednotka práce dokončí úspěšně, je třeba ji *potvrdit*. Všechny aktualizace provedené v rámci této jednotky práce jsou jednou potvrzeny a jsou trvalé a nevratné. Pokud však jednotka práce selže, všechny aktualizace jsou místo *zajištěné*. Tento proces, kde jsou jednotky práce buď potvrzeny, nebo zazálohovány s integritou, je označováno jako *koordinace bodu synchronizace*.

Časový okamžik, kdy jsou všechny aktualizace v rámci pracovní jednotky buď potvrzeny, nebo mají být vráceny zpět, se nazývá *synchronizační bod*. Aktualizace v rámci pracovní jednotky se říká, že se má vyskytnout *v rámci ovládacího prvku bodu synchronizace*. Pokud aplikace požádá o aktualizaci, která je *mimo kontrolu bodu synchronizace*, správce prostředků potvrdí aktualizaci okamžitě, a to i v případě, že probíhá nějaká pracovní jednotka, a aktualizaci nelze provést později.

Počítačový subsystém, který spravuje jednotky práce, se nazývá *správce transakcí* nebo *koordinátor bodů*.

*Lokální* jednotka práce je taková, v níž jsou jedinými aktualizovanými prostředky správce front produktu IBM MQ. Zde je koordinace bodu synchronizace poskytována samotným správcem front pomocí procesu jednofázového potvrzení.

*Globální* jednotka práce je taková, v níž jsou aktualizovány také prostředky patřící jiným správcům prostředků, jako jsou databáze kompatibilní s podporou XA. Zde je třeba použít proceduru s dvoufázovým potvrzováním a jednotku práce může být koordinována správcem front samotným nebo externě jiným správcem transakcí kompatibilním s XA, jako je například IBM TXSeries nebo BEA Tuxedo.

Správce transakcí je zodpovědný za zajištění toho, že všechny aktualizace prostředků v rámci jednotky práce byly úspěšně dokončeny nebo žádné z nich nebyly dokončeny. Jedná se o správce transakcí, který aplikace vydá požadavek na potvrzení nebo vrácení pracovní jednotky. Příklady správců transakcí jsou CICS a WebSphere Application Server, i když obě tyto mají také jinou funkci.

Někteří správci prostředků poskytují svou funkci správy transakcí. Správce front produktu IBM MQ může například spravovat jednotky práce zahrnující aktualizace svých vlastních prostředků a aktualizace tabulek produktu Db2. Správce front nepotřebuje k provedení této funkce samostatného správce transakcí, lze jej však použít, pokud se jedná o požadavek uživatele. Je-li použit oddělený správce transakcí, odkazuje se na něj jako na *správce externích transakcí*.

Aby mohl externí správce transakcí spravovat jednotku práce, musí existovat standardní rozhraní mezi správcem transakcí a každým správcem prostředků, který se podílí na pracovní jednotce. Toto rozhraní umožňuje správcem transakcí a správcem prostředků komunikovat mezi sebou navzájem. Jedno z těchto rozhraní je rozhraní *XA Interface*, které je standardním rozhraním podporovaným řadou správců transakcí a správců prostředků. Rozhraní XA je publikováno skupinou Open Group v tématu *Zpracování distribuovaných transakcí: Specifikace XA*.

Pokud se více než jeden správce prostředků podílí na pracovní jednotce, musí správce transakcí použít protokol *dvoufázového potvrzení*, aby bylo zajištěno, že všechny aktualizace v rámci jednotky práce byly úspěšně dokončeny nebo že žádné z nich nejsou dokončeny, i když dojde k selhání systému. Když aplikace vydá požadavek na správce transakcí za účelem potvrzení transakce, provede správce transakcí následující akce:

#### **Fáze 1 (Připavit na potvrzení)**

Správce transakcí požádá každého správce prostředků, který se podílí na pracovní jednotce, aby zajistil, že všechny informace o zamýšlených aktualizacích svých prostředků budou ve zotavitelném stavu. Správce prostředků to obvykle provádí zápisem informací do protokolu a zajištěním toho, aby informace byly zapsány na pevný disk. Fáze 1 se dokončí poté, co správce transakcí obdrží od každého

správce prostředků oznámení o tom, že informace o zamýšlených aktualizacích jejich prostředků jsou ve zotavitelném stavu.

### **Fáze 2 (Potvrdit)**

Když je fáze 1 úplná, správce transakcí učiní neodvolatelné rozhodnutí o potvrzení transakce. Žádá se každého správce prostředků, který se podílí na transakci, aby potvrdil aktualizace svých prostředků. Když správce prostředků přijme tento požadavek, musí potvrdit aktualizace. V této fázi není k dispozici možnost vrácení těchto dat zpět. Fáze 2 se dokončí poté, co správce transakcí obdrží od každého správce prostředků oznámení o tom, že potvrdil aktualizace svých prostředků.

Rozhraní XA používá protokol s dvoufázovým potvrzováním.

Další informace viz [Scénáře transakční podpory](#).

Produkt IBM MQ také poskytuje podporu pro produkt Microsoft Transaction Server (COM +). [Použití produktu Microsoft Transaction Server \(COM +\)](#) poskytuje informace o tom, jak nastavit produkt IBM MQ pro využití výhod podpory COM +.

## **Rozšíření zařízení správce front**

---

Zařízení správce front lze 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. Mezi podporované uživatelské procedury patří:

#### **Uživatelské procedury kanálu**

Tyto východy mění způsob, jakým kanály fungují. Uživatelské procedury kanálu jsou popsány v tématu [Programy výstupních bodů kanálů pro kanály systému zpráv](#).

#### **Ukončení převodu dat**

Tyto uživatelské procedury vytvoří fragmenty zdrojového kódu, které lze vložit do aplikačních programů pro převod dat z jednoho formátu do jiného. Uživatelské procedury pro převod dat jsou popsány v části [Zápis uživatelských procedur pro převod dat](#).

#### **Ukončení pracovní zátěže klastru**

Funkce, kterou tato procedura provádí, je definována poskytovatelem uživatelské procedury. Informace o definici volání jsou uvedeny v souboru [MQ\\_CLUSTER\\_WORKLOAD\\_EXIT-Call description](#).

### **Uživatelské procedury rozhraní API**

Uživatelské procedury rozhraní API vám umožňují psát kód, který změní chování volání rozhraní API produktu IBM MQ, jako je například MQPUT a MQGET, a pak tento kód vloží bezprostředně před nebo bezprostředně za těmito voláními. Vložení je automatické; správce front řídí návratový kód v registrovaných bodech. Další informace o uživatelských procedurách rozhraní API najdete 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*. Komponenty dodané s produktem IBM MQ můžete použít nebo můžete napsat svou vlastní komponentu za účelem provedení funkcí, které požadujete.

V současné době jsou k dispozici následující instalovatelné služby:

#### **Autorizační služba**

Autorizační služba vám umožňuje vytvořit vlastní mechanismus zabezpečení.

Výchozí komponenta služby, která implementuje službu, je správce oprávnění k objektu (OAM). Ve výchozím nastavení je OAM aktivní a vy nemusíte dělat nic pro jeho konfiguraci. Rozhraní autorizační

služby můžete použít k vytvoření dalších komponent pro nahrazení nebo rozšíření OAM. Další informace o OAM viz téma [Nastavení zabezpečení na systémech AIX, Linux, and Windows](#).

### **služba názvů**

Služba názvů umožňuje aplikacím sdílet fronty tak, že identifikují vzdálené fronty, jako by šlo o lokální fronty.

Můžete napsat svou vlastní komponentu služby názvů. Tuto akci můžete chtít provést, pokud chcete například použít službu názvů s produktem IBM MQ. Chcete-li použít službu názvů, musíte mít buď komponentu, která je buď uživatelem napsaná, nebo dodána jiným dodavatelem softwaru. Ve výchozím nastavení je služba názvů neaktivní.

### **Související pojmy**

[Uživatelské procedury, uživatelské procedury rozhraní API a instalovatelné služby produktu IBM MQ](#)

## **Rozhraní jazyka produktu IBM MQ Java**

---

Produkt IBM MQ poskytuje dvě alternativní rozhraní API (Application Programming Interface) pro použití v aplikacích produktu Java : IBM MQ classes for Java Message Service a IBM MQ classes for Java.

Produkt IBM podporuje a je aktivním účastníkem otevřených standardů a v rámci oblasti systému zpráv je standardem rozhraní API Java Message Service (JMS). V produktu IBM MQ 8.0 produkt implementuje standard JMS 2.0 , který zavádí nové zjednodušené rozhraní API spolu s funkcemi, jako jsou sdílené odběry. Kromě toho má produkt WebSphere Liberty podporu pro produkt JMS 2.0 s výchozím poskytovatelem systému zpráv i s produktem IBM MQ.

V produktu IBM MQ jsou k dispozici dvě alternativní rozhraní API pro použití v aplikacích produktu Java :

### **IBM MQ classes for JMS**

IBM MQ classes for Java Message Service (JMS) je poskytovatel JMS , který je dodáván s IBM MQ.

Produkt Java Platform, Enterprise Edition Connector Architecture (JCA) poskytuje standardní způsob připojení aplikací spuštěných v prostředí produktu Java EE k podnikovému informačnímu systému (EIS), jako je například produkt IBM MQ nebo Db2.

### **IBM MQ classes for Java**

IBM MQ classes for Java umožňuje používat IBM MQ v prostředí Java . IBM MQ classes for Java umožňuje aplikaci Java připojit se k IBM MQ jako klient IBM MQ nebo se připojit přímo ke správci front IBM MQ .

**Poznámka:** IBM MQ classes for Java jsou funkčně stabilizované na úrovni dodávané v produktu IBM MQ 8.0. Existující aplikace, které používají IBM MQ classes for Java budou nadále plně podporovány, ale toto rozhraní API je stabilizované, takže nové funkce nebudou přidány a požadavky na vylepšení jsou odmítnuté. Plně podporováno znamená, že defekty budou opraveny společně s případnými změnami vynuceny změnami systémových požadavků produktu IBM MQ .

V produktu IBM MQ 8.0 jsou produkty IBM MQ classes for Java a IBM MQ classes for JMS sestaveny pomocí produktu Java 7. Běžové prostředí produktu Java 7 podporuje spuštění starších verzí souborů tříd.

### **Související pojmy**

[Proč bych měl používat produkt IBM MQ classes for JMS?](#)

[Proč bych měl používat produkt IBM MQ classes for Java?](#)

[Model produktu JMS](#)

### **Související úlohy**

[Použití funkčnosti produktu JMS 2.0](#)

## **IBM MQ classes for JMS**

IBM MQ classes for JMS je poskytovatel JMS , který je dodáván s produktem IBM MQ. Produkt IBM MQ classes for JMS implementuje rozhraní definovaná v balíku javax.jms a také poskytuje dvě sady rozšíření rozhraní API produktu JMS . Aplikace Java Platform, Standard Edition ( Java SE) i Java Platform, Enterprise Edition ( Java EE) mohou používat IBM MQ classes for JMS.



Specifikace JMS definuje sadu rozhraní, které aplikace mohou 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 k použití. Další informace viz [“JMS model”](#) na stránce 145 a specifikace JMS 2.0 na adrese [Java.net](http://Java.net).

Balík `javax.jms` uvádí podrobnosti o rozhraních produktu JMS a poskytovatel platformy JMS implementuje tato rozhraní pro specifický produkt systému zpráv. IBM MQ classes for JMS je poskytovatel JMS , který implementuje rozhraní JMS pro produkt IBM MQ a také poskytuje následující dvě sady rozšíření rozhraní API produktu JMS :

- Rozšíření produktu IBM MQ JMS
- Rozšíření produktu IBM JMS

Objekt továrny připojení, fronty nebo tématu vytvořený pomocí rozhraní `javax.jms` nebo sada rozšíření produktu JMS lze adresovat pomocí některého z těchto rozhraní API, tj. lze je přetypovat na kterékoli z těchto rozhraní. Chcete-li udržet přenositelnost aplikací na nejvyšší úrovni, použijte nejobecnější rozhraní API, které je vhodné pro vaše požadavky.

## Rozšíření produktu IBM MQ JMS

Produkt IBM MQ classes for JMS také poskytuje rozšíření rozhraní API JMS . Předchozí vydání produktu IBM MQ classes for JMS obsahují rozšíření, která jsou implementována v objektech `MQConnectionFactory`, `MQQueue` a `MQTopic`. Tyto objekty mají vlastnosti a metody, které jsou specifické pro produkt IBM MQ. Objekty mohou být spravovanými objekty nebo aplikace může dynamicky vytvářet objekty za běhu. Produkt IBM MQ classes for JMS udržuje tato rozšíření a vy můžete nadále používat, beze změny, všechny aplikace, které tato rozšíření používají. Tato rozšíření jsou známá jako rozšíření produktu IBM MQ JMS . Všimněte si, že v této sadě dokumentace se objekty, které jsou dynamicky vytvářeny aplikací za běhu, nepovažují za spravované objekty.

## Rozšíření produktu IBM JMS

Kromě rozšíření produktu IBM MQ JMS poskytuje produkt IBM MQ classes for JMS obecnější sadu rozšíření rozhraní API produktu JMS . Tato rozšíření jsou známá jako rozšíření produktu IBM JMS a mají následující obecné cíle:

- Poskytovat vyšší úroveň konzistence mezi poskytovateli produktu IBM JMS
- Chcete-li usnadnit zápis do aplikace mostu mezi dvěma systémy zasílání zpráv IBM
- Usnadní portům aplikace z jednoho poskytovatele IBM JMS do jiného.

Hlavní zaměření těchto rozšíření se týká vytváření a konfigurace továren připojení a míst určení dynamicky za běhu, ale rozšíření také poskytují funkce, které nejsou přímo spojené se systémem zpráv, jako je funkce určování problémů.

## JMS model

Model produktu JMS definuje sadu rozhraní, která mohou aplikace produktu Java používat k provádění operací systému zpráv. IBM MQ classes for JMS, jako poskytovatel JMS definuje, jak se objekty JMS vztahují k konceptům IBM MQ . Specifikace JMS očekává určité objekty JMS , které mají být spravovány objekty.

Specifikace JMS a sada `javax.jms` definují sadu rozhraní, která mohou aplikace produktu Java používat k provádění operací systému zpráv.

V produktu IBM MQ 8.0 produkt podporuje JMS 2.0 verzi standardu JMS, která zavádí zjednodušené rozhraní API a zároveň zachovává klasické rozhraní API z produktu JMS 1.1.

## Zjednodušené API

Produkt JMS 2.0 zavádí zjednodušené rozhraní API a zároveň zachovává doménová specifická rozhraní a nezávislá rozhraní domén z produktu JMS 1.1. Zjednodušené rozhraní API snižuje počet objektů potřebných k odesílání a přijímání zpráv a skládá se z následujících rozhraní:

### ConnectionFactory

ConnectionFactory je spravovaný objekt, který používá klient produktu JMS k vytvoření připojení. Toto rozhraní se také používá v klasickém rozhraní API.

### JMSKontext

Tento objekt kombinuje objekty Connection a Session z klasického rozhraní API. JMSKontextové objekty lze vytvořit z jiných kontextových objektů JMS, přičemž základní připojení bude duplikováno.

### JMSProducent

Producent produktu JMS je vytvořen kontextem produktu JMS a používá se k odesílání zpráv do fronty nebo tématu. Objekt Producent produktu JMS způsobuje vytvoření objektů, které jsou nezbytné k odeslání zprávy.

### JMSSpotřebitel

Odběratel produktu JMS je vytvořen kontextem produktu JMS a slouží k příjmu zpráv z tématu nebo z fronty.

Zjednodušené rozhraní API má několik efektů:

- Kontextové objekty produktu JMS vždy automaticky spustí základní připojení.
- JMSProducenti a JMSSpotřebitelé mohou nyní pracovat přímo se zprávami, aniž by museli získat celý objekt zprávy, a to pomocí metody Message's `getBody`.
- Vlastnosti zprávy lze nastavit na objektu Producent produktu JMS pomocí řetězení metod, před odesláním 'těla', obsahu zpráv. Producent produktu JMS bude zpracovávat vytváření všech objektů, které jsou potřebné k odeslání zprávy. Pomocí produktu 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 zavádí sdílené odběry, ve kterých lze sdílet zprávy mezi více spotřebiteli. Všechny odběry JMS 1.1 jsou považovány za nesdílené odběry.

## Klasické API

Následující seznam shrnuje hlavní rozhraní API 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 vytvoření relací.

### Relace

Relace je jednovláknový kontext pro odesílání a příjem zpráv. Aplikace používá relaci k vytváření zpráv, producentů zpráv a spotřebitelů zpráv. Relace je buď zpracovávána, nebo se nejedná o transakci.

### Zpráva

Objekt Message zapouzdřuje zprávu, kterou aplikace odesílá nebo přijímá.

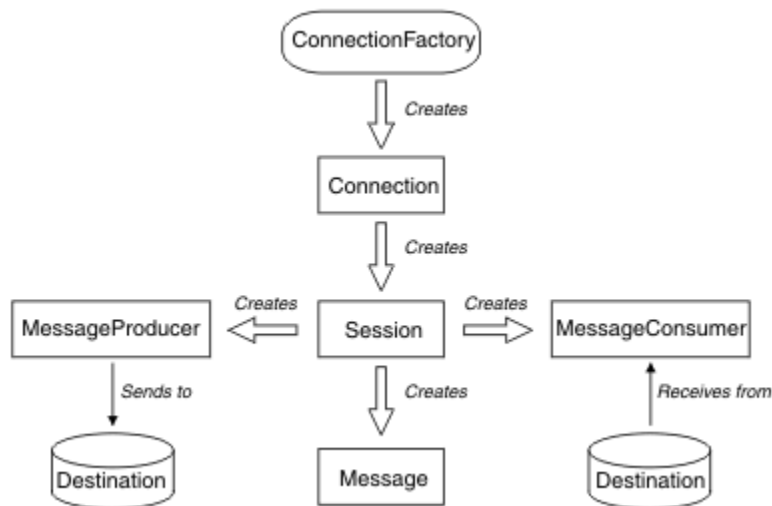
## MessageProducer

Aplikace používá producenta zpráv k odesílání zpráv do místa určení.

## MessageConsumer

Aplikace používá spotřebitele zpráv k přijetí zpráv odeslaných do místa určení.

Produkt [Obrázek 54 na stránce 147](#) zobrazuje tyto objekty a jejich vztahy.



Obrázek 54. Objekty produktu JMS a jejich vztahy

Diagram zobrazuje hlavní rozhraní: ConnectionFactory, Connection, Session, MessageProducer, MessageConsumer, Message, and Destination. Aplikace používá továrnu připojení k vytvoření připojení a používá připojení k vytváření relací. Aplikace pak může použít relaci k vytváření zpráv, producentů zpráv a spotřebitelů zpráv. Aplikace používá producenta zpráv k odesílání zpráv do místa určení a k příjmu zpráv odeslaných na místo určení používá spotřebitele zpráv.

Místo určení, ConnectionFactorynebo Objekt připojení mohou být použity souběžně různými podprocesy aplikace s více podprocesy, ale objekt Session, MessageProducernebo MessageConsumer nemůže být použit souběžně různými podprocesy. Nejjednodušším způsobem, jak zajistit, aby relace MessageProducernebo MessageConsumer nebyla použita souběžně, je vytvoření samostatného objektu relace pro každý podproces.

Produkt JMS podporuje dva styly systému zpráv:

- Dvoubodový systém zpráv
- Publikování/odběr zpráv

Tyto styly systému zpráv jsou označovány také jako *domény systému zpráva* v aplikaci můžete kombinovat oba styly systému zpráv. V doméně dvoubodového spojení je cílem fronta a v doméně publikování/odběru je cílem téma.

S verzí produktu JMS před JMS 1.1, programování pro dvoubodovou doménu 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 1.1můžete používat společnou sadu rozhraní a metod, které podporují obě domény systému zpráv. Společná rozhraní poskytují nezávislý pohled domény pro každou doménu systému zpráv. Příkaz [Tabulka 16 na stránce 147](#) vypisuje nezávislá rozhraní domény JMS a jejich odpovídající rozhraní specifická pro doménu.

| Tabulka 16. Doména JMS nezávislá a jejich odpovídající rozhraní specifická pro doménu. |                                                     |                                                               |
|----------------------------------------------------------------------------------------|-----------------------------------------------------|---------------------------------------------------------------|
| Nezávislá rozhraní domény                                                              | Doména specifická pro doménu pro dvoubodovou doménu | Rozhraní specifická pro doménu pro doménu publikování/ odběru |
| ConnectionFactory                                                                      | Továrna QueueConnection                             | Továrna TopicConnection                                       |

Tabulka 16. Doména JMS nezávislá a jejich odpovídající rozhraní specifická pro doménu. (pokračování)

| Nezávislá rozhraní domény | Doména specifická pro doménu pro dvoubodovou doménu | Rozhraní specifická pro doménu pro doménu publikování/ odběru |
|---------------------------|-----------------------------------------------------|---------------------------------------------------------------|
| Připojení                 | QueueConnection                                     | TopicConnection                                               |
| Místo určení              | Fronta                                              | Téma                                                          |
| Relace                    | QueueSession                                        | TopicSession                                                  |
| MessageProducer           | QueueSender                                         | TopicPublisher                                                |
| MessageConsumer           | QueueReceiver<br>QueueBrowser                       | TopicSubscriber                                               |

Produkt JMS 2.0 uchovává všechna rozhraní specifická pro doménu a existující aplikace tak mohou tato rozhraní používat i nadále. Pro nové aplikace však zvažte použití nezávislých rozhraní domény produktu JMS 1.1 nebo zjednodušeného rozhraní API produktu JMS 2.0.

V produktu IBM MQ classes for JMS se objekty JMS vztahují k koncepcím produktu IBM MQ následujícími způsoby:

- Objekt připojení má vlastnosti, které jsou odvozeny od vlastností továrny připojení, která byla použita k vytvoření připojení. Tyto vlastnosti řídí způsob, jakým se aplikace připojuje ke správci front. Příklady těchto vlastností jsou název správce front a pro aplikaci, která se připojuje ke správci front v režimu klienta, název hostitele nebo adresa IP systému, v němž je spuštěn správce front.
- Objekt relace zapouzdřuje popisovač připojení IBM MQ, který proto definuje transakční rozsah relace.
- Objekt MessageProducer a objekt MessageConsumer obsahují každý zapouzdřující popisovač objektu IBM MQ.

Při použití IBM MQ classes for JMS se použijí všechna běžná pravidla produktu IBM MQ. Všimněte si zejména, že aplikace může odeslat zprávu do vzdálené fronty, ale může přijmout zprávu pouze z fronty, kterou vlastní správce front, ke kterému je aplikace připojena.

Specifikace JMS očekává objekty ConnectionFactory a Destination Object, aby byly spravovány objekty. Administrátor vytváří a udržuje spravované objekty v centrálním úložišti a aplikace JMS načítá tyto objekty pomocí rozhraní JNDI (Java Naming and Directory Interface).

V produktu IBM MQ classes for JMS je implementace rozhraní Destination abstraktní nadtržída fronty a tématu, takže instance cíle je buď objekt fronty, nebo objekt Topic. Nezávislá rozhraní domény zpracovávají frontu nebo téma jako cíl. Doména systému zpráv pro objekt MessageProducer nebo MessageConsumer je určena podle toho, zda je cílem fronta nebo téma.

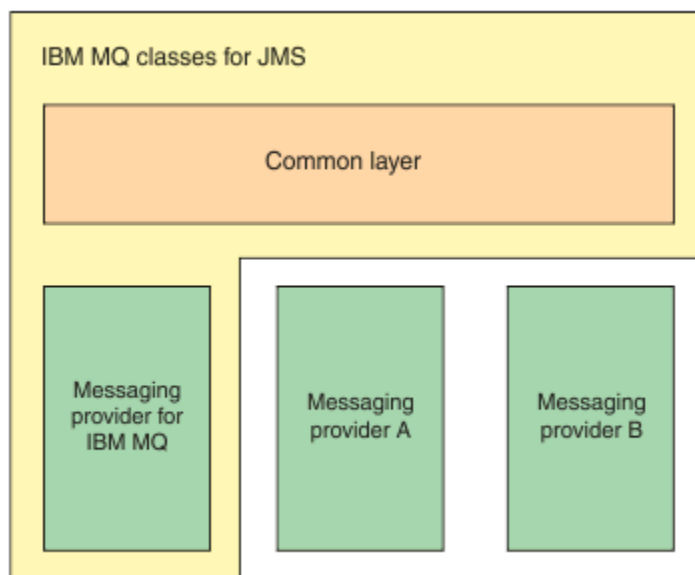
V produktu IBM MQ classes for JMS mohou být objekty následujících typů administrované objekty:

- ConnectionFactory
- Továrna QueueConnection
- Továrna TopicConnection
- Fronta
- Téma
- XAConnectionFactory
- Továrna XAQueueConnection
- Továrna XATopicConnection-továrna

## Architektura produktu IBM MQ classes for JMS

Produkt IBM MQ classes for JMS má vrstvenou architekturu. Nejvyšší vrstva kódu je obecná vrstva, kterou může použít libovolný poskytovatel produktu IBM JMS.

Produkt IBM MQ classes for JMS má vrstvenou architekturu, jak je zobrazeno v diagramu Obrázek 55 na stránce 149. Nejvyšší vrstva kódu je běžná vrstva, kterou může použít libovolný poskytovatel produktu IBM JMS . Když aplikace volá metodu JMS , provede se veškerá zpracování volání, která není specifická pro systém zasílání zpráv, prováděna běžnou vrstvou, která také poskytuje konzistentní odpověď na volání. Veškeré zpracování volání, které je specifické pro systém zpráv, je delegováno na nižší vrstvu. V následujícím diagramu je poskytovatel systému zpráv produktu IBM MQ zobrazen v dolní vrstvě spolu se dvěma dalšími poskytovateli systému zpráv (poskytovatel systému zpráv A a poskytovatel systému zpráv B.).



Obrázek 55. Vrstvená architektura pro poskytovatele IBM JMS

Vrstvená architektura splňuje následující cíle:

- Zlepšit konzistenci chování různých poskytovatelů produktu IBM JMS
- Chcete-li usnadnit zápis do aplikace mostu mezi dvěma systémy zasílání zpráv IBM
- Uspadní portům aplikace z jednoho poskytovatele IBM JMS do jiného.

## Podpora pro spravované objekty

Produkt IBM MQ classes for JMS podporuje použití spravovaných objektů.

Tok logiky v aplikaci JMS začíná s objekty ConnectionFactory a Destination objects. Aplikace používá objekt ConnectionFactory k vytvoření objektu připojení, který představuje aktivní připojení z aplikace k serveru systému zpráv. Aplikace používá objekt Connection k vytvoření objektu relace, který je jediným kontextem podprocesu pro vytváření a příjem zpráv. Aplikace pak může použít objekt Relace a Cílový objekt k vytvoření objektu MessageProducer , který aplikace používá k odeslání zpráv do zadaného místa určení. Místo určení je buď fronta, nebo téma v systému zasílání zpráv a je zapouzdřeno objektem místa určení. Aplikace může také použít objekt Relace a Cílový objekt k vytvoření objektu MessageConsumer , který aplikace používá k příjmu zpráv, které byly odeslány do zadaného cíle.

Specifikace JMS očekává objekty ConnectionFactory a Destination Object, aby byly spravovány objekty. Administrátor vytváří a udržuje spravované objekty v centrálním úložišti a aplikace JMS načte tyto objekty pomocí produktu Java Naming Directory Interface (JNDI). Úložiště spravovaných objektů může být v rozsahu od jednoduchého souboru do adresáře LDAP (Lightweight Directory Access Protocol).

Produkt IBM MQ classes for JMS podporuje použití spravovaných objektů. Aplikace může používat všechny funkce produktu IBM MQ classes for JMS , které jsou vystaveny prostřednictvím produktu IBM MQ , aniž by byly všechny informace specifické pro produkt IBM MQ kódovány do samotné aplikace. Toto uspořádání poskytuje aplikaci se stupněm nezávislosti na základní konfiguraci produktu IBM MQ . Aby bylo možné tuto nezávislost dosáhnout, může aplikace pomocí rozhraní JNDI načítat továrny připojení

a cíle, které jsou uloženy jako spravované objekty, a používat pouze rozhraní definovaná v balíku javax.jms k provádění operací systému zpráv. Administrátor může pomocí nástroje pro administraci produktu IBM MQ JMS nebo Průzkumníka IBM MQ vytvářet a spravovat spravované objekty v centrálním úložišti. Aplikační server však obvykle poskytuje vlastní úložiště pro spravované objekty a své vlastní nástroje pro vytváření a údržbu objektů. Aplikace Java EE proto může pomocí produktu JNDI načítat spravované objekty buď z úložiště aplikačního serveru, nebo z centrálního úložiště.

### **Související informace**

[Konfigurace prostředků JMS](#)

## **Podporované typy komunikace na platformách Java EE**

Produkt IBM MQ classes for JMS na platformě Java EE podporuje dva typy komunikace mezi komponentou aplikace a správcem front IBM MQ .

Jsou podporovány následující dva typy komunikace mezi komponentou aplikace a správcem front produktu IBM MQ :

- Odchozí komunikace
- Příchozí komunikace

### **Odchozí komunikace**

Přímé použití rozhraní API produktu JMS , komponenta aplikace vytvoří připojení ke správci front a poté odesílá a přijímá zprávy.

Aplikační komponenta může být například aplikační klient, servlet, stránka JSP ( Java Server Page), objekt EJB (Enterprise Java Bean) nebo objekt typu message-driven bean (MDB). V tomto typu komunikace kontejner aplikačního serveru poskytuje pouze funkce nízké úrovně pro podporu operací systému zpráv, jako je sdružování připojení a správa podprocesů.

### **Příchozí komunikace**

V případě příchozí komunikace je doručena zpráva přicházející do místa určení k objektu MDB, který poté zpracuje zprávu.

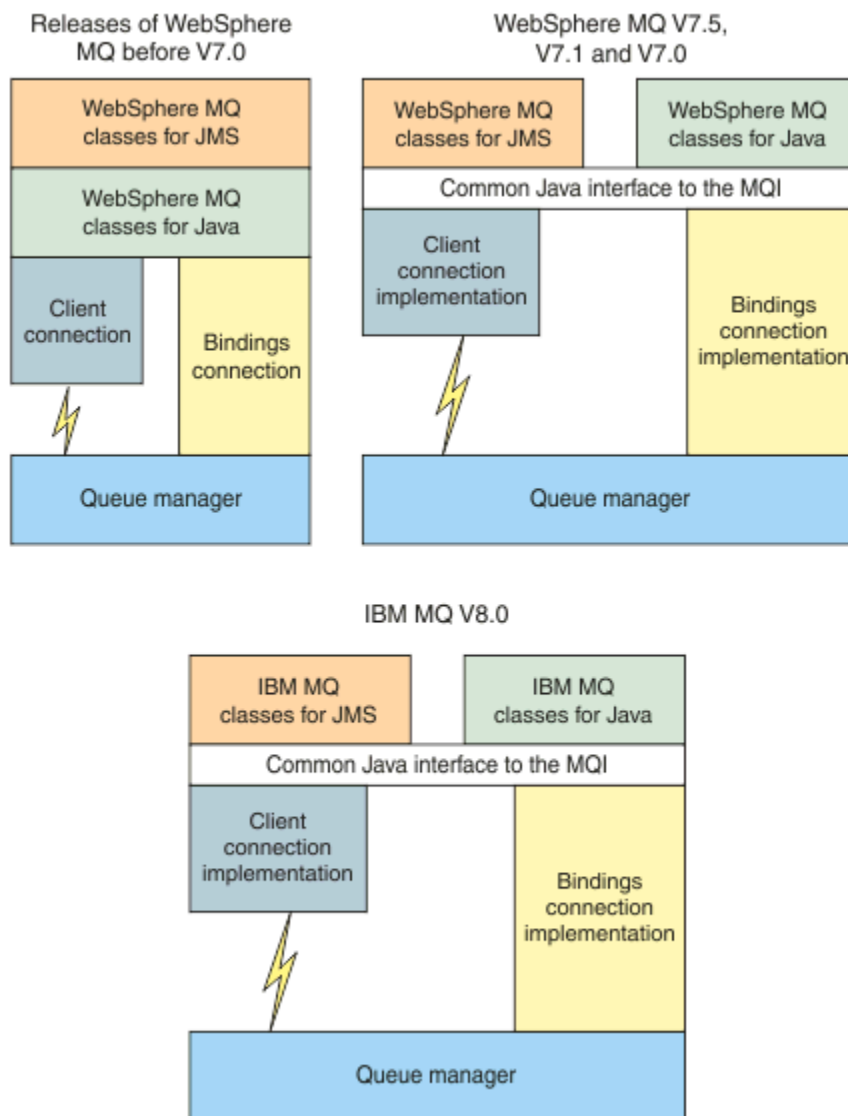
Aplikace Java EE používají objekty MDB k asynchronnímu zpracování zpráv. Objekt MDB se chová jako modul listener pro zprávy produktu JMS a je implementován metodou onMessage(), která definuje způsob zpracování zprávy. Objekt MDB je implementován v kontejneru EJB aplikačního serveru. Přesný způsob, jakým je objekt MDB konfigurován, závisí na tom, jaký aplikační server používáte, ale informace o konfiguraci musí určovat, ke kterému správci front se má připojit, jak se připojit ke správci front, který cíl má být sledován pro zprávy, a transakční chování objektu MDB. Tyto informace pak budou použity kontejnerem EJB. Je-li doručena zpráva splňující kritéria výběru objektu typu message-driven bean v zadaném místě určení, kontejner EJB použije k načtení zprávy ze správce front produktu IBM MQ classes for JMS a poté předá zprávu do objektu MDB voláním jeho metody onMessage().

## **Relace s IBM MQ classes for Java**

Protože produkty IBM WebSphere MQ 7.0, IBM WebSphere MQ classes for JMS a IBM WebSphere MQ classes for Java jsou implementovány jako rovnocenné uzly. Tato implementace se liší od předchozích verzí, kde implementace produktu IBM WebSphere MQ classes for JMS závisí na produktu IBM WebSphere MQ classes for Java.

Pro verze starší než IBM WebSphere MQ 7.0 bylo IBM WebSphere MQ classes for JMS implementováno téměř výhradně jako vrstva kódu v horní části IBM WebSphere MQ classes for Java. Toto uspořádání způsobilo zmatení mezi vývojáři aplikací, protože nastavení polí nebo volání metod ve třídě MQEnvironment může způsobit nechtěné a neočekávané dopady na chování kódu, které je zapsáno pomocí produktu IBM MQ classes for JMS. Kromě toho implementace produktu IBM MQ classes for JMS měla některá omezení v oblastech, ve kterých není rozhraní API produktu JMS přirozené, pokud jde o IBM MQ classes for Java, a tato omezení vedla k určitým problémům s výkonem běhového prostředí.

V produktu IBM WebSphere MQ 7.0 již implementace produktu IBM MQ classes for JMS není závislá na produktu IBM MQ classes for Java. IBM MQ classes for Java a IBM MQ classes for JMS jsou rovnocenné uzly, které používají společné rozhraní Java pro rozhraní MQI. Toto uspořádání umožňuje větší rozsah optimalizace výkonu a znamená, že nastavení polí nebo volání metod ve třídě MQEnvironment nemá žádný vliv na chování kódu, který je napsán pomocí produktu IBM MQ classes for JMS, na chování za běhu. Obrázek 56 na stránce 151 zobrazuje vztah mezi IBM MQ classes for JMS a IBM MQ classes for Java v předchozích vydáních produktu IBM WebSphere MQ classes for JMS a ve vydáních před IBM WebSphere MQ 7.0 a jak se tento vztah změnil pro pozdější vydání.



Obrázek 56. Vztah mezi IBM MQ classes for JMS a IBM MQ classes for Java

Chcete-li zachovat kompatibilitu s verzemi dřívějšími než IBM WebSphere MQ 7.0, mohou třídy uživatelské procedury kanálu, které jsou zapsány v produktu Java, stále používat rozhraní IBM MQ classes for Java, i když jsou třídy uživatelské procedury kanálu volány z produktu IBM MQ classes for JMS. Použití rozhraní IBM MQ classes for Java však znamená, že vaše aplikace jsou stále závislé na souboru JAR IBM MQ classes for Java, `com.ibm.mq.jar`. Pokud nechcete `com.ibm.mq.jar` v cestě ke třídě, můžete místo toho použít sadu rozhraní v balíku `com.ibm.mq.exits`.

V produktu IBM WebSphere MQ 7.0 můžete vytvářet a konfigurovat spravované objekty produktu JMS pomocí Průzkumníka IBM MQ.

## IBM MQ Poskytovatel systému zpráv

Poskytovatel systému zpráv produktu IBM MQ má tři režimy provozu: normální režim, normální režim s omezeními a režim migrace.

Poskytovatel systému zpráv produktu IBM MQ má tři režimy provozu:

- Normální režim poskytovatele systému zpráv produktu IBM MQ
- IBM MQ s omezeními
- Režim migrace poskytovatele systému zpráv produktu IBM MQ

Normální režim poskytovatele systému zpráv produktu IBM MQ používá všechny funkce správce front produktu IBM MQ k implementaci produktu JMS. Tento režim je optimalizovaný pro použití rozhraní API a funkčnosti JMS 2.0.

Normální režim poskytovatele systému zpráv produktu IBM MQ s omezeními používá rozhraní API produktu JMS 2.0, ale ne nové funkce produktu IBM MQ 8.0, jako jsou sdílené odběry, odložené doručení nebo asynchronní odeslání.

Režim migrace poskytovatele systému zpráv produktu IBM MQ je založen na funkci IBM WebSphere MQ 6.0 a používá pouze funkce, které byly k dispozici ve správci front produktu IBM WebSphere MQ 6.0 za účelem implementace produktu JMS. Můžete se připojit k produktu IBM WebSphere MQ 7.0 nebo k novějším správcům front, kteří používají režim migrace, ale nemůžete použít žádné optimalizační nastavení produktu IBM WebSphere MQ 7.0. Tento režim umožňuje připojení k jednomu z níže uvedených verzí správce front:

1. Produkt IBM WebSphere MQ 7.0 nebo novější správce front ve vazbách nebo v režimu klienta, ale tento režim používá pouze ty funkce, které byly k dispozici pro správce front produktu IBM WebSphere MQ 6.0.
2. IBM WebSphere MQ 6.0 nebo předchozí správce front v režimu klienta.

Chcete-li se připojit k produktu IBM Integration Bus pomocí produktu IBM MQ Enterprise Transport, použijte režim migrace. Používáte-li produkt IBM MQ Real-Time Transport, je režim migrace automaticky vybrán, protože jste explicitně vybrali vlastnosti v objektu továrny připojení. Připojení k produktu IBM Integration Bus pomocí produktu IBM MQ Enterprise Transport odpovídá obecným pravidlům pro výběr režimu, který je popsán v části [Konfigurace vlastnosti produktu JMS PROVIDERVERSION](#).

### Související úlohy

[Konfigurace prostředků JMS](#)

z/OS

## IBM MQ for z/OS koncepce

Některé z koncepcí používaných produktem IBM MQ for z/OS jsou jedinečné pro platformu z/OS. Například mechanismus protokolování, techniky správy úložného prostoru, dispozice jednotky zotavení a skupiny sdílení front jsou poskytovány pouze s produktem IBM MQ for z/OS. V tomto tématu naleznete další informace o těchto koncepcích.

### Související pojmy

[“Správce front v systému z/OS” na stránce 153](#)

Než budete moci nechat své aplikační programy používat produkt IBM MQ ve svém systému z/OS, je třeba nainstalovat produkt IBM MQ for z/OS a spustit správce front. Správce front vlastní a spravuje sadu prostředků, které používá produkt IBM MQ.

[“Inicializátor kanálu na systému z/OS” na stránce 154](#)

Inicializátor kanálu poskytuje a spravuje prostředky, které umožňují IBM MQ distribuovaných front.

Produkt IBM MQ používá *agenty kanálů zpráv* (MCA) k odesílání zpráv z jednoho správce front do jiného.

[“Výrazy a úlohy pro správu IBM MQ for z/OS” na stránce 156](#)

Toto téma se používá jako úvod do terminologie a úloh, které jsou specifické pro produkt IBM MQ for z/OS.

[“Sdílené fronty a skupiny sdílení front” na stránce 158](#)



Můžete použít sdílené fronty a skupiny sdílení front za účelem implementace vysoké dostupnosti prostředků produktu IBM MQ . 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 203](#)

Tento oddíl popisuje řazení do front v rámci skupiny, funkce IBM MQ for z/OS je jedinečná pro platformu z/OS . Tato funkce je k dispozici pouze pro správce front definované pro skupinu sdílení front.

[“Správa úložišť na systému z/OS” na stránce 216](#)

Produkt IBM MQ for z/OS vyžaduje trvalé a dočasné datové struktury a používá k ukládání těchto dat sady stránek a vyrovnávací paměti. Tato témata poskytují více podrobností o tom, jak produkt IBM MQ používá tyto sady stránek a vyrovnávací paměti.

[“přihlášení IBM MQ for z/OS” na stránce 220](#)

Produkt IBM MQ uchovává *protokoly* změn dat a významných událostí, jakmile k nim dojde. Tyto protokoly lze v případě potřeby použít k obnovení dat do předchozího stavu.

[“Zotavení a restartování v systému z/OS” na stránce 242](#)

Použijte odkazy v tomto tématu k vyhledání informací o funkcích produktu IBM MQ for z/OS pro restart a zotavení.

[“Koncepte zabezpečení v produktu IBM MQ for z/OS” na stránce 258](#)

Toto téma slouží k pochopení důležitosti zabezpečení pro produkt IBM MQ a o důsledcích neodpovídajícího nastavení zabezpečení v systému.

[“Dostupnost v systému z/OS” na stránce 265](#)

Produkt IBM MQ for z/OS má mnoho funkcí pro vysokou dostupnost. Toto téma popisuje některé aspekty dostupnosti.

[“Dispozice jednotky zotavení v systému z/OS” na stránce 269](#)

Určité transakční aplikace mohou při připojení ke správci front ve skupině sdílení front (QSG) při připojení ke správci front v rámci skupiny sdílení front (QSG) při připojení ke správci front používat místo názvu správce front místo názvu skupiny sdílení front jako název skupiny sdílení front. To umožňuje, aby zotavení transakce bylo pružnější a robustnější tím, že odeberete požadavek na opětovné připojení ke stejnému správci front v QSG.

### **Související odkazy**

[“Definování systému v systému z/OS” na stránce 231](#)

Produkt IBM MQ for z/OS používá mnoho výchozích definic objektů a poskytuje ukázkou JCL pro vytvoření těchto výchozích objektů. Toto téma slouží k pochopení těchto výchozích objektů a ukázky JCL.

[“Monitorování a statistika v systému IBM MQ for z/OS” na stránce 268](#)

IBM MQ for z/OS má sadu zařízení pro monitorování správce front a shromažďování statistik.

## **z/OS Správce front v systému z/OS**

Než budete moci nechat své aplikační programy používat produkt IBM MQ ve svém systému z/OS , je třeba nainstalovat produkt IBM MQ for z/OS a spustit správce front. Správce front vlastní a spravuje sadu prostředků, které používá produkt 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.

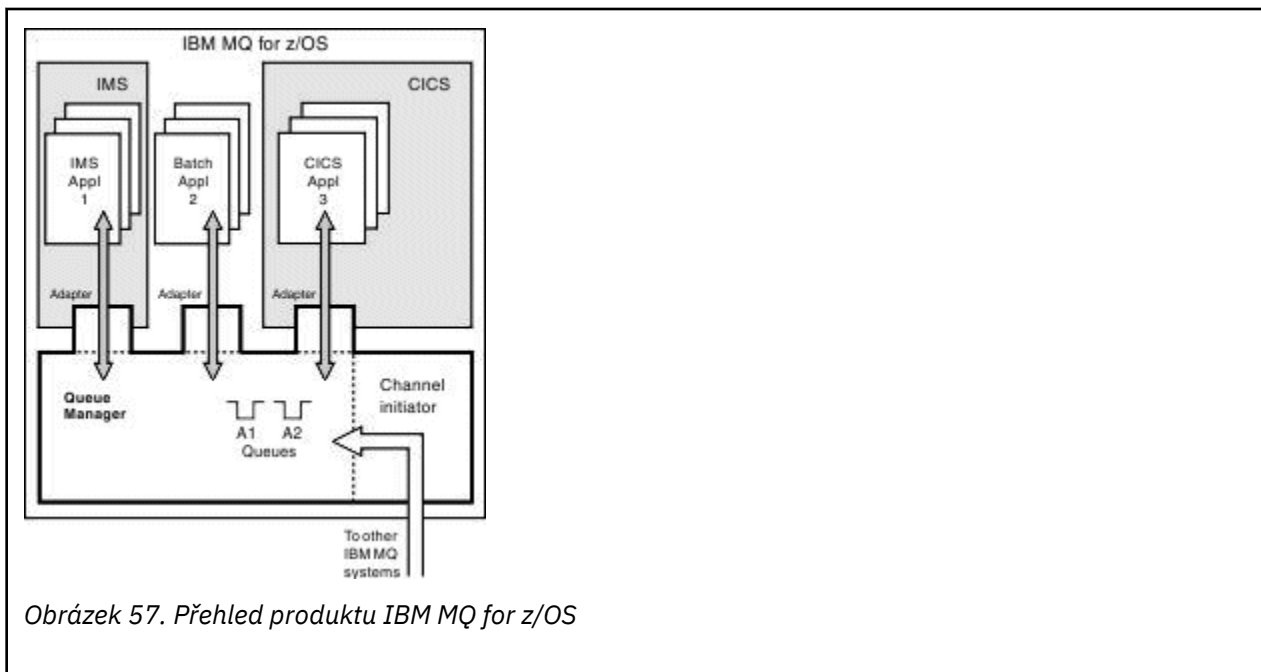
Mezi prostředky spravované správcem front patří následující:

- Sady stránek, které obsahují definice objektu IBM MQ a data zpráv
- Protokoly, které se používají k obnově zpráv a objektů v případě selhání správce front
- Paměť procesoru

- Produkt Connections, jehož prostřednictvím mají různá prostředí aplikací ( CICS, IMSa Batch) přístup k rozhraní API produktu IBM MQ
- Inicializátor kanálu IBM MQ , který umožňuje komunikaci mezi produktem IBM MQ v systému z/OS a dalšími systémy.

Název správce front má název a aplikace se k němu mohou připojit pomocí tohoto názvu.

Obrázek 57 na stránce 154 ilustruje správce front, který ukazuje připojení k různým prostředím aplikací a inicializátor kanálu.



Obrázek 57. Přehled produktu IBM MQ for z/OS

## Subsystém správce front v systému z/OS

V systému z/OS se 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 produktu z/OS , které obsahují informace o protokolech a které obsahují 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.

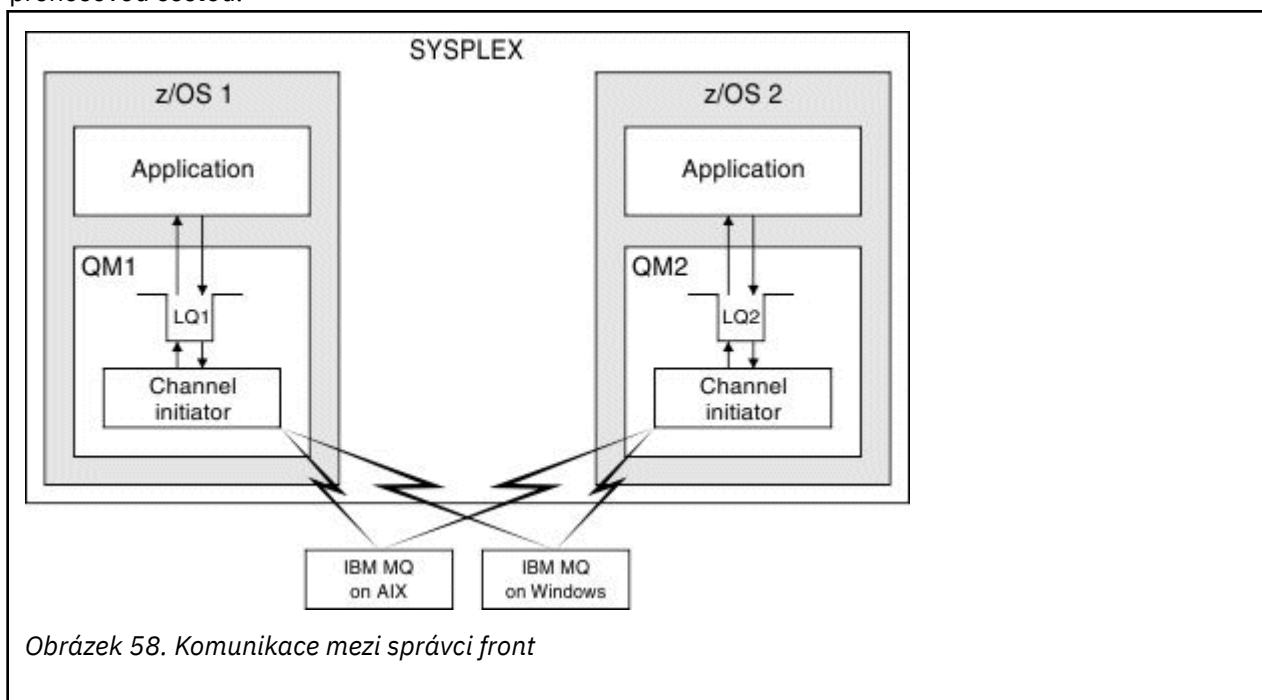
## z/OS Inicializátor kanálu na systému z/OS

Inicializátor kanálu poskytuje a spravuje prostředky, které umožňují IBM MQ distribuovaných front. Produkt IBM MQ používá *agenty kanálů zpráv* (MCA) k odesílání zpráv z jednoho správce front do jiného.

Chcete-li odeslat zprávy ze správce front A do správce front B, musí být *odesílající* MCA ve správci front A nastaven komunikační spojení se správcem front B. *Přijímající* MCA musí být spuštěn ve správci front B, aby bylo možné přijímat zprávy z komunikačního spojení. Tato jednosměrná cesta skládající se z odesílající sběrnice MCA, komunikačního propojení a přijímajícího agenta MCA se označuje jako *kanál*. Odesílající agent MCA přijme zprávy z přenosové fronty a odešle je pomocí kanálu do přijímajícího agenta MCA. Přijímající agent MCA přijímá zprávy a vkládá je do cílových front.

V produktu IBM MQ for z/OS se odesílá a přijímá MCA všechna spuštění uvnitř inicializátoru kanálu (inicializátor kanálu je také známý jako *modul pro přesouvání* ). Inicializátor kanálu se spouští jako adresní prostor produktu z/OS pod kontrolou správce front. K jednomu správci front může být připojen pouze jeden iniciátor kanálu a je spuštěn ve stejném obrazu produktu z/OS jako správce front. Současně mohou být spuštěny tisíce procesů MCA v rámci iniciátoru kanálu.

Obrázek 58 na stránce 155 zobrazuje dva správce front v rámci prostředí sysplex. Každému správci front je přiřazen inicializátor kanálu a lokální fronta. Zprávy odeslané správci front v systémech AIX a Windows jsou umístěny do lokální fronty, odkud jsou načítány aplikací. Zprávy odpovědi jsou vráceny podobnou přenosovou cestou.



Obrázek 58. Komunikace mezi správci front

Inicializátor kanálu také obsahuje další procesy, které se zabývají správou kanálů. Tyto procesy zahrnují:

#### Moduly listener

Tyto procesy naslouchají příchozím požadavkům na kanál v komunikačním subsystému, jako je například TCP, a při přijetí příchozího požadavku spustit jmenovanou sběrnici MCA.

#### Supervizor

Spravuje adresní prostor inicializátoru kanálu, například je zodpovědný za restartování kanálů po selhání.

#### Server názvů

Používá se k vyřešení názvů TCP na adresy.

#### Úlohy TLS

Ty se používají k provedení šifrování a dešifrování a kontrole seznamů odvolaných certifikátů.

### z/OS Záznamy SMF pro inicializátor kanálu

Inicializátor kanálu (CHINIT) může vytvářet statistické záznamy SMF a evidenční záznamy s informacemi o úlohách a kanálech.

CHINIT mohou vytvářet záznamy statistiky SMF a záznamy evidence s následujícími typy informací:

- Úlohy: dispečer, adaptér, server Domain Name Server (DNS) a SSL. Tyto úlohy formují to, co se nazývá statistika CHINIT.
- Kanály: poskytuje informace o účtování podobné informacím, které jsou k dispozici s příkazem DIS CHSTATUS. To se nazývá evidence kanálu.

Produkt IBM MQ for Multiplatforms poskytuje podobné informace při zápisu zpráv PCF do systému SYSTEM.ADMIN.STATISTICS.QUEUE. Další informace o tom, jak jsou informace o statistice zaznamenány v produktu IBM MQ for Multiplatforms, najdete v tématu [Data zprávy o statistice kanálu](#).

#### Data statistiky

Tyto informace můžete použít k vyhledání následujících informací:

- Zda potřebujete více úloh CHINIT, jako je například počet SSL TCB a kolik CPU tyto úlohy používají.
- Průměrná doba pro požadavky na tyto úlohy.
- Nejdelší požadavek na dobu trvání v intervalu a čas dne, kdy se tento požadavek vyskytl, pro úlohy DNS a SSL. Můžete korelovat tento čas dne s problémy, se kterými se můžete setkat s kanálem.

## Účetní údaje

Tyto informace můžete použít ke sledování využití kanálu a zjištění informací o následujících tématech:

- Kanály s nejvyšší propustností.
- Rychlost, jakou byly zprávy odeslány, a rychlost odesílání dat v MB/sekundy.
- Dosažená velikost dávky. Je-li dosažená velikost dávky blízká velikosti dávky určené pro kanál, může být kanál blízko jeho limitu pro odesílání zpráv.

Příkazy `START TRACE` a `STOP TRACE` slouží k řízení kolekce trasování evidence a trasování statistiky. Můžete použít volby `STATCHL` a `STATACLS` na kanálu a správci front, abyste mohli řídit, zda kanály produkuje data SMF.

z/OS

## Výrazy a úlohy pro správu IBM MQ for z/OS

Toto téma se používá jako úvod do terminologie a úloh, které jsou specifické pro produkt IBM MQ for z/OS.

Některé z termínů a úloh vyžadovaných 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](#)
- [Přizpůsobování prostředí správce front](#)
- [Restartování a obnova](#)
- [Zabezpečení](#)
- [Dostupnost](#)
- [Manipulace s objekty](#)
- [Monitorování a statistika](#)
- [Prostředí aplikace](#)

## Sdílené fronty

Fronty mohou být *nesdílené*, vlastněné a přístupné pouze jednomu správci front nebo *sdíleno*, jejichž vlastníkem je *skupina sdílení front*. Skupina sdílení front se skládá z určitého počtu správců front spuštěných v rámci jednoho prostředí sysplex z/OS, který může přistupovat ke stejným definicím objektů a datům zpráv produktu IBM MQ souběžně. V rámci skupiny sdílení front jsou sdílitelné definice objektů uloženy ve sdílené databázi Db2. Zprávy sdílené fronty jsou drženy v jedné nebo více strukturách prostředku Coupling Facility (struktury CF). Pokud jsou data zprávy příliš velká pro uložení přímo ve struktuře (více než 63 kB), nebo pokud je zpráva dostatečně velká, že pravidla definovaná pro instalaci ji vybírají pro odlehčování, informace o řízení zpráv jsou stále uloženy v položce prostředku Coupling Facility, ale data zprávy jsou odložena do sdílené datové sady zpráv (SMDS) nebo do sdílené databáze produktu Db2. Sdílené datové sady zpráv, sdílená databáze produktu Db2 a struktury prostředku Coupling Facility jsou prostředky, které jsou spravovány společně 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 takovým způsobem, že ji lze načíst poté, co další operace získá zprávu ze stejné fronty. Je-li zpráva odebrána z fronty, prostor v sadě stránek, který uchovává data, je později uvolněn pro opětovné použití. Počet zpráv uložených ve frontě se zvyšuje, a proto se velikost prostoru využitého v sadě stránek a počet zpráv ve frontě zmenšuje, prostor použitý v sadě stránek se snižuje.

Chcete-li snížit náklady na zápis dat a čtení dat ze sad stránek, správce front uloží aktualizace do vyrovnávací paměti procesoru. Množství paměti použité pro vyrovnávací paměť přístupu k sadě stránek je řízeno prostřednictvím objektů IBM MQ nazývaných *fondy vyrovnávacích pamětí*.

Další informace o sadách stránek a fondech vyrovnávacích pamětí naleznete v tématu [Správa úložiště](#).

## Protokolování

Jakékoli změny v objektech zadržených na sadách stránek a operacích na trvalých zprávách se zaznamenají jako záznamy protokolu. Tyto záznamy protokolu jsou zapsány do datové sady protokolu zvané *aktivní protokol*. Název a velikost datové sady aktivního protokolu jsou uloženy v datové sadě nazvané *zaváděcí datová sada* (BSDS).

Když se aktivní datová sada protokolu zaplní, správce front se přepne na jinou datovou sadu protokolu, takže protokolování může pokračovat, a zkopíruje obsah úplných aktivních dat protokolu do datové sady *protokolu archivace*. Informace o těchto akcích, včetně názvu datové sady archivního protokolu, jsou uloženy v datové sadě zaváděcího programu. Koncepčně existuje kroužek aktivních datových sad žurnálu, které správce front prochází; když je zaplněn aktivní protokol, data protokolu jsou odložena do protokolu archivace a datová sada aktivního protokolu je k dispozici pro opětovné použití.

Další informace o protokolu a datových sadách pro samozavedení naleznete v tématu [“přihlášení IBM MQ for z/OS”](#) na stránce 220.

## Přizpůsobení prostředí správce front

Při spuštění správce front dojde ke čtení sady inicializačních parametrů, které řídí způsob, jakým správce front pracuje. Kromě toho se čtou datové sady obsahující příkazy IBM MQ a příkazy, které obsahují, jsou spuštěny. Obvykle tyto datové sady obsahují definice systémových objektů požadovaných pro spuštění produktu IBM MQ a vy je můžete upravit tak, aby byly definovány nebo inicializovány objekty produktu IBM MQ nezbytné pro vaše provozní prostředí. Při čtení těchto datových sad jsou všechny objekty definované pomocí těchto datových sad uloženy buď na sadě stránek, nebo v produktu Db2.

Další informace o inicializačních parametrech a systémových objektech najdete v tématu [“Definování systému v systému z/OS”](#) na stránce 231.

## Zotavení a restartování

Kdykoli během operace IBM MQ mohou být provedeny změny v paměti procesoru, které ještě nebyly zapsány do sady stránek. Tyto změny jsou zapsány do sady stránek, která byla naposledy použita úlohou na pozadí v rámci správce front.

Je-li správce front ukončen nestandardně, může fáze zotavení při restartu správce front obnovit ztracené změny sady stránek, protože v záznamech protokolu jsou uchovena trvalá data zprávy. To znamená, že produkt IBM MQ může obnovit trvalá data zprávy a změny objektu až do bodu selhání.

Pokud správce front, který je členem skupiny sdílení front, narazí na 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 restartu naleznete v příručce [“Zotavení a restartování v systému z/OS”](#) na stránce 242.

## Zabezpečení

Můžete použít externího správce zabezpečení, jako např. Security Server (dříve známé jako RACF) chcete-li chránit prostředky, které produkt IBM MQ vlastní a spravuje z přístupu neoprávněných uživatelů. Pro zabezpečení kanálu můžete také použít protokol TLS (Transport Layer Security). TLS je zahrnuto jako součást produktu IBM MQ.

Další informace o zabezpečení produktu IBM MQ naleznete v tématu [“Koncepce zabezpečení v produktu IBM MQ for z/OS”](#) na stránce 258.

## Dostupnost

Existuje několik funkcí produktu IBM MQ, které jsou navrženy ke zvýšení dostupnosti systému v případě selhání správce front nebo selhání komunikačního subsystému. Další informace o těchto funkcích viz [“Dostupnost v systému z/OS”](#) na stránce 265.

## Manipulace s objekty

Když je správce front spuštěn, můžete manipulovat s objekty produktu IBM MQ buď prostřednictvím rozhraní konzoly produktu z/OS, nebo pomocí obslužného programu pro administraci, který používá služby ISPF v rámci TSO. Oba mechanismy umožňují definovat, měnit nebo odstraňovat objekty IBM MQ. Můžete také řídit a zobrazovat stav různých funkcí produktu IBM MQ a správce front.

You can also manipulate IBM MQ objects using the IBM MQ Explorer, a graphical user interface that provides a visual way of working with queues, queue managers, and other objects.

Další informace o těchto zařízeních naleznete v tématu [Vydávání příkazů](#).

## Monitorování a statistika

Pro monitorování vašich správců front a inicializátorů kanálů je k dispozici několik zařízení. Můžete také shromažďovat statistiky pro vyhodnocení výkonu a účetní účely.

Další informace o těchto zařízeních naleznete v tématu [“Monitorování a statistika v systému IBM MQ for z/OS”](#) na stránce 268.

## Aplikační prostředí

Po spuštění správce front se k němu mohou aplikace připojit a spustit je pomocí rozhraní API produktu IBM MQ. Může se jednat o aplikace CICS, IMS, dávky nebo WebSphere Application Server. Aplikace produktu IBM MQ mohou také přistupovat k aplikacím v systémech CICS a IMS, které nejsou informovány o produktu IBM MQ, pomocí mostů CICS a IMS.

Další informace o těchto zařízeních naleznete v tématu [“IBM MQ a další produkty z/OS”](#) na stránce 272.

Informace o zápisu aplikací IBM MQ naleznete v následující dokumentaci:

- [Vývoj aplikací](#)
- [Použití C++](#)
- [Použití produktu IBM MQ classes for Java](#)

## Sdílené fronty a skupiny sdílení front

Můžete použít sdílené fronty a skupiny sdílení front za účelem implementace vysoké dostupnosti prostředků produktu IBM MQ. Sdílené fronty a skupiny sdílení front jsou funkce jedinečné pro produkt IBM MQ for z/OS na platformě z/OS.

Tato sekce popisuje atributy a výhody a nabízí informace o tom, jak může několik správců front sdílet stejné fronty a zprávy v těchto frontách.

## Související pojmy

[“Co je sdílená fronta?” na stránce 159](#)

Sdílená fronta je typem lokální fronty. K zprávám v této frontě může přistupovat jeden nebo více správců front, kteří jsou v prostředí sysplex.

[“Co je skupina sdílení front?” na stránce 160](#)

Skupina správců front, kteří mohou přistupovat ke stejným sdíleným frontám, se nazývá skupina sdílení front. Každý člen skupiny sdílení front má přístup ke stejné sadě sdílených front.

[“Kde jsou zadrženy sdílené zprávy fronty?” na stránce 162](#)

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 178](#)

Sdílená fronta umožňuje, aby aplikace produktu IBM MQ byly rozšiřitelné, vysoce dostupné a aby bylo možné implementovat vyrovnávání pracovní zátěže.

[“Distribuované fronty a skupiny sdílení front” na stránce 197](#)

Distribuované fronty a skupiny sdílení front jsou dvě techniky, které můžete použít ke zvýšení dostupnosti vašich aplikačních systémů. Použijte toto téma k vyhledání dalších informací o těchto technikách.

[“Ovlivňování distribuce pracovní zátěže se sdílenými frontami” na stránce 201](#)

Prostřednictvím tohoto tématu můžete porozumět faktorům, které ovlivňují distribuci pracovní zátěže se sdílenými frontami ve skupině sdílení front.

## Související odkazy

[“Kde najdete další informace o sdílených frontách a skupinách sdílení front” na stránce 202](#)

Pomocí tabulky v tomto tématu naleznete více informací o tom, jak produkt IBM MQ for z/OS používá sdílené fronty a skupiny sdílení front.

## Co je sdílená fronta?

Sdílená fronta je typem lokální fronty. K zprávám v této frontě může přistupovat jeden nebo více správců front, kteří jsou v prostředí sysplex.

## Skupina sdílení front

Správci front, kteří mohou přistupovat ke stejné sadě sdílených front, tvoří skupinu s názvem *skupina sdílení front*.

## Ke zprávám může přistupovat libovolný správce front

Jakýkoliv správce front ve skupině sdílení front může přistupovat ke sdílené frontě. To znamená, že můžete vložit zprávu do sdílené fronty do jednoho správce front a získat stejnou zprávu z fronty z jiného správce front. To poskytuje rychlý mechanismus pro komunikaci v rámci skupiny sdílení front, která nevyžaduje, aby kanály byly mezi správci front aktivní.

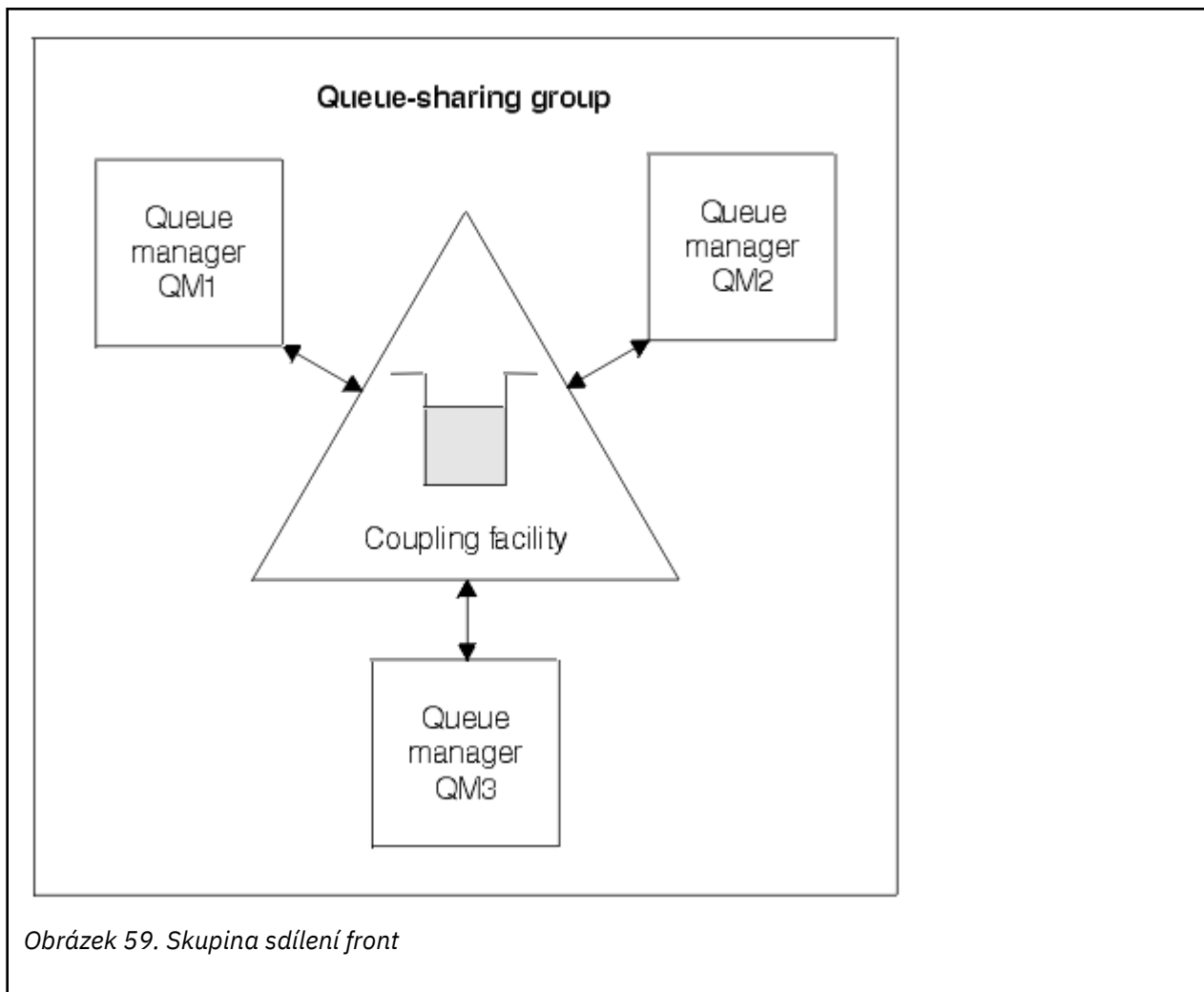
Produkt IBM WebSphere MQ 7.1 a pozdější podporují odlehčování zpráv do produktu Db2 nebo do sdílené datové sady zpráv (SMDS). Vypokládání zpráv libovolné velikosti lze konfigurovat.

Ve starších verzích produktu IBM MQ mají velké zprávy (> 63 kB) zástupný symbol uložený v prostředku Coupling Facility (4 K) a jejich data zprávy uložena v produktu Db2.

Produkt Obrázek 59 na stránce 160 zobrazuje tři správce front a prostředek CF, který tvoří skupinu sdílení front. Všichni tři správci front mají přístup 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 mohou přistupovat ke všem sdíleným frontám, aplikace nezávisí na dostupnosti specifického správce front; všechny správce front ve skupině sdílení front mohou tuto frontu zařadit do fronty.

To dává větší dostupnost, protože všichni ostatní správci front ve skupině sdílení front mohou pokračovat ve zpracování fronty, pokud má některý z správců front problém.



## 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 v rámci skupiny sdílení front. To znamená, že je třeba provést méně definic.

Naproti tomu definice nesdílené fronty je uložena na sadě stránek nula správce front, který je vlastníkem fronty (jak je popsáno v tématu [Sady stránek](#)).

Sdílenou frontu nelze definovat, pokud byla fronta s tímto názvem již 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.

### **z/OS** Co je skupina sdílení front?

Skupina správců front, kteří mohou přistupovat ke stejným sdíleným frontám, se nazývá skupina sdílení front. Každý člen skupiny sdílení front má přístup ke stejné sadě sdílených front.

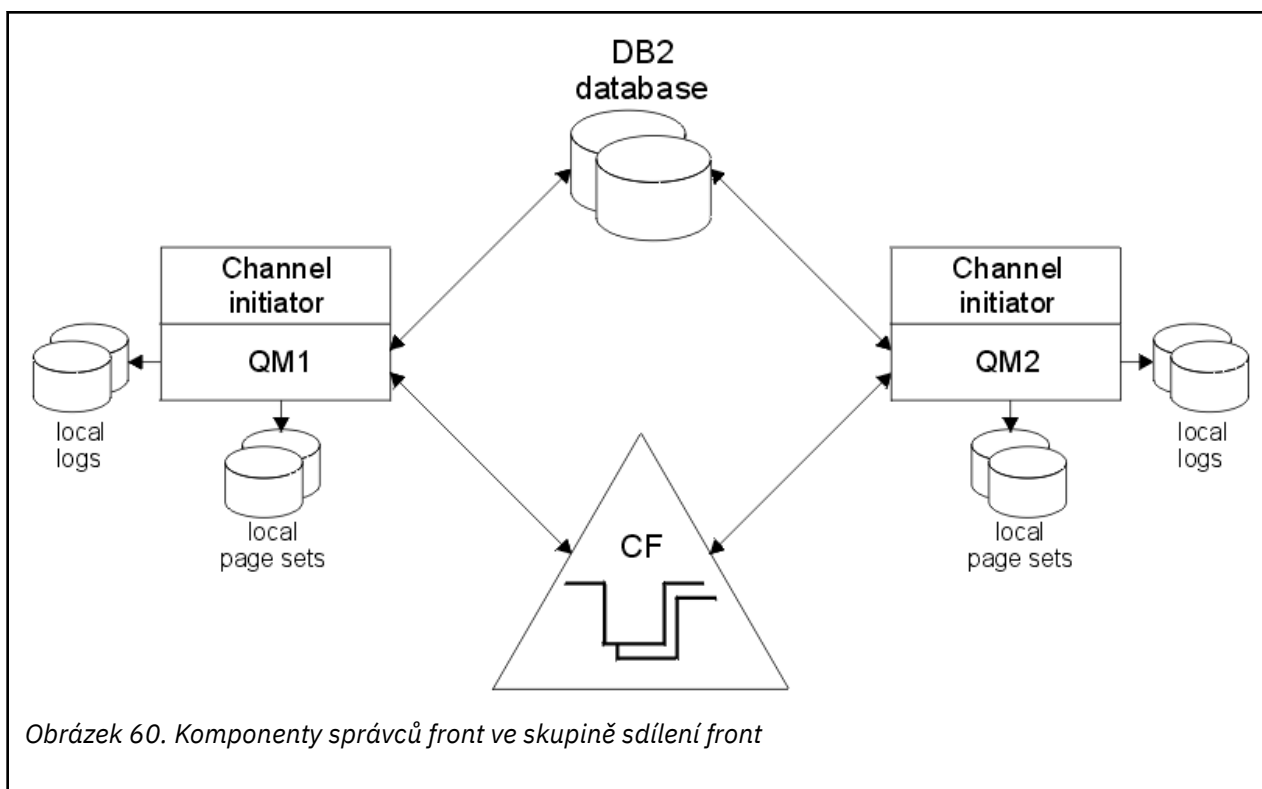
Názvy skupin sdílení front jsou tvořeny nejvýše čtyřmi znaky. Název musí být v síti jedinečný a nesmí být shodný s žádným názvem správce front.



Obrázek 60 na stránce 161 ilustruje skupinu sdílení front, která obsahuje dva správce front. Každý správce front má inicializátor kanálu a jeho vlastní sady lokálních stránek a datové sady protokolů.

Každý člen skupiny sdílení front se musí také připojit k systému Db2 . Všechny systémy Db2 musí být ve stejné skupině sdílení dat Db2 , aby mohli správci front přistupovat ke sdílenému úložišti Db2 používanému k uchování definic sdílených objektů. Jedná se o definice libovolného typu objektu IBM MQ (například fronty a kanály), které jsou definovány pouze jednou, a každý správce front v této skupině je může použít. Ty se nazývají *globální* definice a jsou popsány v tématu Soukromé a globální definice.

Více než jedna skupina sdílení front může odkazovat na konkrétní skupinu sdílení dat. Uvedete název subsystému Db2 a která skupina sdílení dat používá správce front v systémových parametrech systému IBM MQ při spuštění.



Obrázek 60. Komponenty správců front ve skupině sdílení front

Když se správce front připojí ke skupině sdílení front, má přístup ke sdíleným objektům definovaným pro tuto skupinu a tento správce front můžete použít k definování nových sdílených objektů v rámci skupiny. Jsou-li ve skupině definovány sdílené fronty, můžete pomocí tohoto správců front vkládat zprávy do těchto sdílených front a získávat zprávy z těchto sdílených front. Všechny správce front ve skupině mohou načítat zprávy uložené ve sdílené frontě.

Můžete zadat příkaz MQSC jednou a nechat jej provést ve všech správcích front v rámci skupiny sdílení front, jako kdyby byl zadán v každém správci front jednotlivě. Pro tento atribut se používá atribut *rozsah příkazu* . Tento atribut je popsán v části Směrování příkazů do různých správců front.

Je-li správce front spuštěn jako člen skupiny sdílení front, musí být možné rozlišit mezi objekty produktu IBM MQ definované soukromě k tomuto správci front a objektům produktu IBM MQ definovaným globálně, které jsou k dispozici pro všechny správce front ve skupině sdílení front. Pro tento atribut se používá atribut *dispozice skupiny sdílení front* . Tento atribut je popsán v tématu Soukromé a globální definice.

Můžete definovat jednu sadu profilů zabezpečení, které řídí přístup k objektům produktu IBM MQ kdekoli v rámci skupiny. To znamená, že počet profilů, které musíte definovat, se výrazně sníží.

Správce front může náležet pouze do jedné skupiny sdílení front a všichni správci front ve skupině musí být ve stejném prostředí sysplex. Určíte, která skupina sdílení front bude správce front patřit do parametrů systému při spuštění.

## 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. Viz [IBM MQ V8 Funkce a vylepšení](#), kapitola 8.

**Důležité:** IBM z16 je plánováno jako poslední generace produktu IBM Z<sup>®</sup> 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 164.

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 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 512 bajtů dat.

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 179.

**Poznámka:** Při použití IBM MQ můžete šifrovat strukturu prostředku Coupling Facility. Další informace viz [Šifrování dat struktury prostředku Coupling Facility](#) .

## 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.

## 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 164

Můžete zvolit místo uložení dat zprávy pro zprávu sdílené fronty, a to buď v tabulce Db2 , nebo do sdílené datové sady zpráv (SMDS) zpráv. Můžete také vybrat, které zprávy se odlehčují, na základě velikosti zprávy a aktuálního využití struktury prostředku Coupling Facility (CF).

“Správa prostředí sdílené datové sady zpráv (SMDS)” na stránce 166

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 místo uložení dat zprávy pro zprávu sdílené fronty, a to buď v tabulce Db2 , nebo do sdílené datové sady zpráv (SMDS) zpráv. Můžete také vybrat, které zprávy se odlehčují, na základě velikosti zprávy a aktuálního využití struktury prostředku Coupling Facility (CF).

Data zprávy pro sdílené fronty lze odložit ze zařízení CF a uložit buď do tabulky Db2 , nebo do datové sady spravované produktem IBM MQ , která se nazývá *sdílená datová sada zpráv* (SMDS).

Pro zprávy, které jsou větší než velikost záznamu prostředku CF 63 kB, mohou mít data odložování zpráv do SMDS výrazné zlepšení výkonu ve srovnání s odlehčováním na Db2.

Každá zpráva o sdílené frontě je stále spravována pomocí položky seznamu ve struktuře prostředku Coupling Facility, ale když jsou data zprávy odložena do SMDS, položka prostředku Coupling Facility obsahuje pouze některé řídicí informace a seznam odkazů na příslušné bloky na disku, kde je zpráva uložena. Použití tohoto mechanismu znamená, že množství paměti prvků prostředku Coupling Facility požadované pro každou zprávu je pouze zlomkem skutečné velikosti zprávy.

#### **Výběr místa, kde jsou uloženy zprávy sdílené fronty**

Výběr SMDS nebo sdíleného úložiště zpráv obslužného programu Db2 je řízen parametrem **OFFLOAD(SMDS|DB2)** v definici **CFSTRUCT** . Výchozí hodnota je **OFFLOAD(SMDS)** .

Tento parametr také vyžaduje, aby **CFSTRUCT** používal **CFLEVEL (5)** nebo vyšší. Na strukturu CF na této úrovni se mohou připojovat pouze správci front v produktu IBM WebSphere MQ 7.1 nebo vyšší.

Strukturu lze změnit pouze na **CFLEVEL (5)** , pokud jsou všichni správci front ve skupině sdílení front IBM WebSphere MQ 7.1 nebo vyšší.

Argument **OFFLOAD** je platný pouze od **CFLEVEL (5)** . Další podrobnosti viz [DEFINE CFSTRUCT](#) .

Produkt **OFFLOAD(DB2)** je podporován především pro účely migrace.

#### **Výběr, které sdílené fronty zpráv jsou odloženy**

Data zprávy jsou odložena do SMDS nebo Db2 na základě velikosti dat zprávy a aktuálního použití struktury prostředku Coupling Facility. Jsou zde tři pravidla a každé pravidlo určuje odpovídající dvojici parametrů. Tyto parametry jsou odpovídající procentní částí prahové hodnoty využití struktury prostředku Coupling Facility ( **OFFLDnTH** ) a omezení velikosti zpráv ( **OFFLDnSZ** ).

Aktuální implementace těchto tří pravidel se zadává pomocí následujících párů klíčových slov:

- OFFLD1TH a OFFLD1SZ
- OFFLD2TH a OFFLD2SZ
- OFFLD3TH a OFFLD3SZ

| <b>Dvojice pravidel</b> | <b>Výchozí hodnota</b>          | <b>Popis</b>                                                                                                   |
|-------------------------|---------------------------------|----------------------------------------------------------------------------------------------------------------|
| Dvojice pravidel 1      | OFFLD1TH(70)<br>a OFFLD1SZ(32K) | Je-li struktura prostředku Coupling Facility více než 70% úplných dat odlehčování pro zprávy přesahující 32 kB |

| Dvojice pravidel   | Výchozí hodnota             | Popis                                                                                                                           |
|--------------------|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------|
| Dvojice pravidel 2 | OFFLD2TH(80) a OFFLD2SZ(4K) | Je-li struktura prostředku Coupling Facility více než 80% úplných dat odlehčování pro zprávy přesahující 4 kB                   |
| Dvojice pravidel 3 | OFFLD3TH(90) a OFFLD3SZ(0K) | Je-li struktura prostředku Coupling Facility více než 90% úplných dat odlehčování pro zprávy přesahující 0 kB (všechny zprávy), |

Má-li pravidlo odlehčování hodnotu OFFLD x SZ hodnoty 64K , znamená to, že pravidlo není v platnosti. V takovém případě budou zprávy odloženy pouze v případě, že je v platnosti jiné pravidlo odlehčování, nebo je-li zpráva větší než 63.75 kB, a proto je tato zpráva příliš velká a nelze ji uložit do struktury.

Každá odložená zpráva stále vyžaduje 0.75 KB úložiště ve spojovacím zařízení.

Tři pravidla odlehčování, která mohou být určena pro každou strukturu, mají být použita následujícím způsobem.

- Výkon

- Pokud je ve struktuře aplikace dostatek místa, data zprávy by měla být odložena pouze v případě, že je příliš velká pro uložení ve struktuře nebo pokud překročí některou nižší prahovou hodnotu velikosti zpráv tak, aby hodnota výkonu při ukládání ve struktuře nestála za hodnotu struktury prostoru, kterou by potřebovala.
- Je-li požadována určitá prahová hodnota velikosti zpráv, je konvenčně určena pomocí prvního pravidla odlehčování.

- kapacita

- Je-li ve struktuře aplikace málo místa, maximální objem dat zprávy by měl být odložen tak, aby se co nejlépe používalo zbývající prostor.
- Třetí pravidlo odlehčování je konvenčně použito k označení toho, že když je struktura téměř plná, většina zpráv by měla být odložena, takže položky ve struktuře aplikace budou zpravidla minimální velikosti (vyžadující přibližně 0.75K bajtů).
- Parametr prahové hodnoty využití by měl být vybrán na základě velikosti struktury aplikace a maximálního počtu nevyřízených požadavků. Je-li například maximální očekávaný počet očekávaných položek 1M zpráv, pak velikost úložiště struktury požadované pro tento počet zpráv bude přibližně 0.75G bajtů. To znamená například, že je-li struktura přibližně 10G bajtů, prahová hodnota využití pro odlehčování všech zpráv musí být nastavena na 92% nebo nižší.
- Struktura struktury je rozdělena na prvky a položky, a i když může být celkově dostatek prostoru, jedna z nich může být spuštěna před druhou. Systém poskytuje schopnost AUTOALTER upravit poměr, je-li to nezbytné, ale to není příliš citlivé, takže množství skutečně dostupného prostoru může být poněkud 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 stanovena na 80%.

- Přejídný přechod:

- Vzhledem k tomu, že množství prostoru ponechané ve struktuře prostředku Coupling Facility se snižuje, bylo by nežádoucí mít velkou náhlou změnu charakteristik výkonu. Je také nežádoucí, aby správa prostředku Coupling Facility měla náhlé změny prahové hodnoty v obvyklém poměru mezi položkami a prvky, které se používají.
- Druhé pravidlo odlehčování je konvenčně použito k zajištění mezilehlého polštáře mezi výkonností a nepodloženými pravidly odlehčování. Může být nastavena tak, že způsobí značné zvýšení aktivity odlehčování, pokud prostor použitý ve struktuře prostředku Coupling Facility překročí střední prahovou hodnotu. To znamená, že zbývající prostor se používá pomaleji a poskytuje

automatické zpracování změny prostředku CF více času na přizpůsobení se vyšším úrovním využití.

Pokud strukturu prostředku Coupling Facility nelze rozbalit a je třeba uložit alespoň nějaký předem určený počet zpráv, třetí pravidlo může být změněno tak, aby bylo zajištěno, že odlehčování dat pro všechny zprávy začíná na odpovídající prahové hodnotě, aby se zajistilo, že prostor je 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  $1\,000\,000 * 0.75$  kB, což je 768 MB, 18.75% 4 GB. V tomto případě musí být prahová hodnota pro odlehčování všech zpráv nastavena na přibližně 80%, nikoli na 90%. Tím se získají parametry OFFLD3TH(80) a OFFLD3SZ(OK). Je třeba také upravit ostatní parametry odlehčování zátěže.

Je-li zjištěno, že odlehčování velmi malých zpráv má významný dopad na výkon, ale relativní dopad je menší pro větší zprávy, pak mohou být prahové hodnoty využití pro ostatní pravidla zredukovány, aby bylo možné odlehčit větší počet zpráv dříve, a ponechat tak více prostoru ve struktuře pro malé zprávy, než je třeba je odsunout.

Pokud například zprávy přesahující 32KB často nastávají, ale výkon při jejich načítání (jak je určen statistikou RMF nebo výkonem aplikací) je velmi podobný, je-li to pro jejich uchování ve spojovacím zařízení velmi podobné, pak by prahová hodnota pro první pravidlo mohla být nastavena na 0%, chcete-li tyto zprávy převést na hodnotu 0%. To poskytnete parametry OFFLD1TH(0) a OFFLD1SZ(32K). Opět by bylo nutné upravit ostatní parametry odlehčování.

Pokud existuje mnoho zpráv okolo specifických středních velikostí, například 16 KB a 6 KB, může být užitečné změnit volbu velikosti zprávy pro druhé pravidlo tak, aby větší z nich byly odloženy při poměrně nízké prahové hodnotě využití, čímž se ušetří značný prostor, ale menší se budou stále ukládat pouze do spojovacího zařízení.

## **Správa prostředí sdílené datové sady zpráv (SMDS)**

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 SMDS”](#) na stránce 177

### **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 správci 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 vlastnícímu 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 vlastní správce front pro zpracování restartu.

### PRÁZDNÝ

Datová sada neobsahuje žádné zprávy. Datová sada je uvedena do tohoto stavu, pokud je normálně zavřena vlastním správcem 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 vlastním správcem 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 vlastním správcem front.
- Ověření záhlaví datové sady se nezdaří poté, co bylo úspěšně otevřeno libovolným správcem front.
- Pokud vlastní 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 vlastnímu správci front provádět zpracování restartu. Po dokončení zpracování restartu se stav změní na AKTIVNÍ a všichni ostatní správci 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 správci 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 správci 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 uložené 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 vlastnícímu 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 rozbálí 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 prvků, 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.

Vyrovňovací 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")<sup>1</sup>.

- 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 pamětí 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.

---

<sup>1</sup>  $(\text{Hits} / (\text{Hits} + \text{Misses})) * 100$

## 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. 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 přidělit 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** vlastníci front.

### Datovou sadu jen pro čtení nelze otevřít

Pokud správce front nemůže přidělit 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 uzavřena a uvolněna všemi správci front, takže může být například přesunut na nový svazek, než bude znovu zpřístupněna.

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 přidané 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 obnovení 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 změní 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 prostředku 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 uvidíte, 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 SMDS

Toto téma popisuje a poskytuje přístup k příkazům, které souvisejí se sdílenými datovými sadami zpráv.

Zobrazit a změnit volby obslužného programu **CFSTRUCT** týkající se velkých zpráv odlehčování zpráv (**OFFLOAD** a odlehčování pravidel) a sdílených datových sad zpráv (**DSGROUP**, **DSBLOCK**, **DSBUFS**, **DSEXPAND**):

- [ZOBRAZENÍ CFSTRUCT](#)
- [DEFINE CFSTRUCT](#)
- [ALTER CFSTRUCT](#)
- [ODSTRANIT CFSTRUCT](#)

Zobrazit stav **CFSTRUCT** týkající se diskového odlehčování zpráv (**OFFLDUSE**):

- [ZOBRAZIT STAV CFSTATUS](#)

Zobrazit a změnit volby přepisu datové sady (**DSEXPAND** a **DSBUFS**) pro jednotlivé správce front:

- [Zobrazit sadu SMDS](#)
- [ALTER SMDS](#)

Zobrazte nebo upravte stav a dostupnost datových sad v rámci skupiny sdílení front:

- [ZOBRAZENÍ TYPU CFSTATUS \(SMDS\)](#)
- [Resetovat SMDS](#)

Zobrazit informace o využití prostoru sady dat SMDS a informace o využití vyrovnávací paměti pro správce front:

- [ZOBRAZIT TYP POUŽITÍ \(SMDS\)](#)

Zobrazit nebo upravit stav a dostupnost připojení (**SMDSCONN**) do datových sad z jednotlivých správců front:

- [ZOBRAZENÍ SMDS](#)
- [ZAČÁTEK SMDSCONN](#)
- [STOP SMDSCONN](#)

Zálohovat a obnovit sdílené zprávy, včetně velkých dat zprávy v SMDS, je-li to nezbytné:

- [BACKUP CFSTRUCT](#)
- [RECOVER CFSTRUCT](#)

## **Výhody používání sdílených front**

Sdílená fronta umožňuje, aby aplikace produktu 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 a přidává nové instance serverové aplikace, nebo dokonce přidává nový obraz produktu z/OS se správcem front (ve skupině sdílení front) a kopií aplikace.
- Je vysoce dostupný.
- Přirozeně provádí *stažení* vyrovnávání pracovní zátěže na základě dostupné procesorové kapacity jednotlivých správců front ve skupině sdílení front.

### **Použití sdílených front pro vysokou dostupnost**

Následující příklady ilustrují, jak můžete použít sdílenou frontu ke zvýšení dostupnosti aplikací.

Představte si scénář IBM MQ, ve kterém klientské aplikace spuštěné v síti chtějí vytvářet požadavky na serverové aplikace spuštěné na serveru z/OS. Aplikace klienta vytvoří zprávu požadavku a umístí ji do fronty požadavků. Klient potom čeká na odpověď ze serveru, odeslanou na frontu pro odpověď uvedenou v deskriptoru zprávy požadavku.

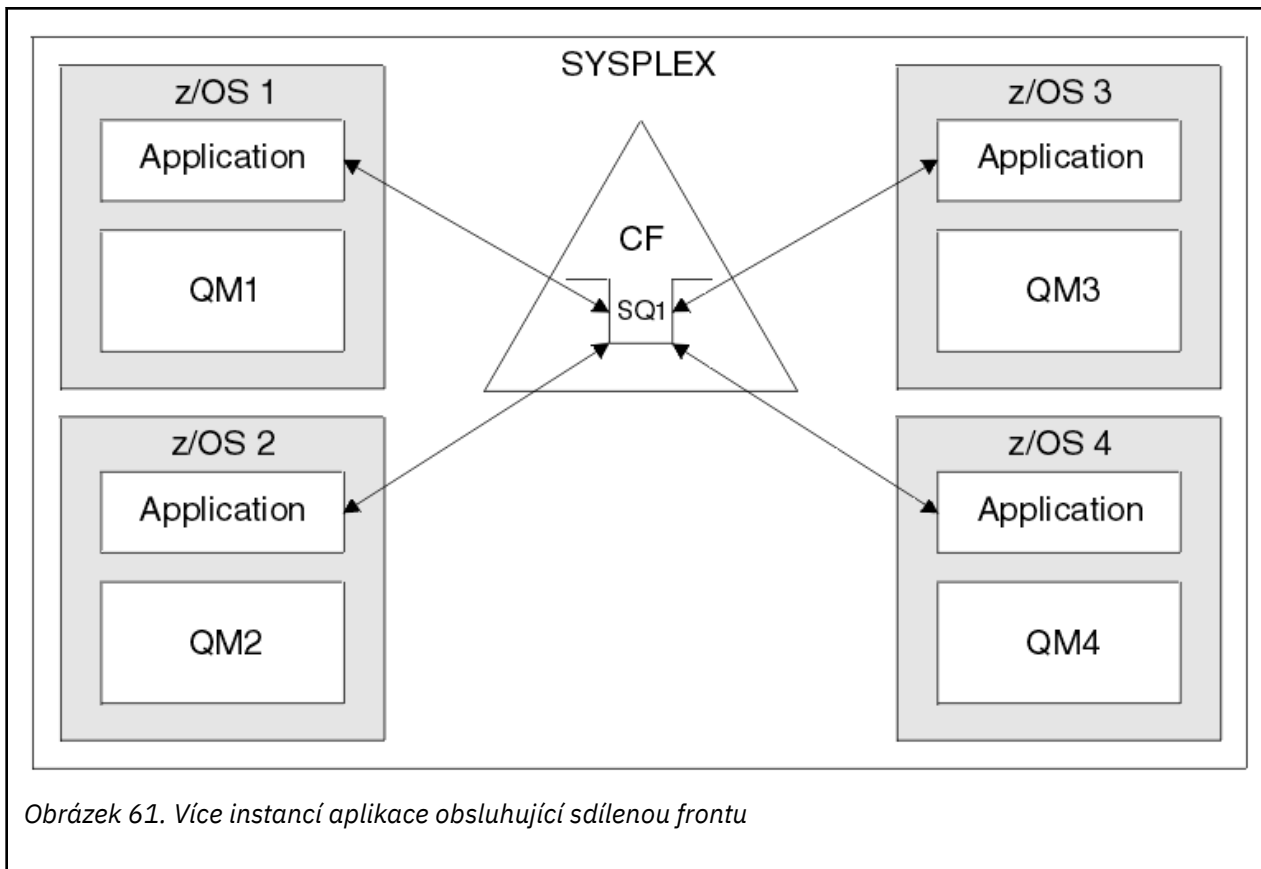
Produkt IBM MQ spravuje přenos zprávy požadavku z počítače klienta do vstupní fronty serveru v systému z/OS a z odezvy serveru zpět na klienta. Definováním vstupní fronty serveru jako sdílené fronty mohou být všechny zprávy zařazené do fronty načítány do libovolného správce front v rámci skupiny sdílení front. To znamená, že můžete nakonfigurovat správce front na každém obrazu produktu z/OS v prostředí sysplex a tím, že je připojíte ke stejné skupině sdílení front, může každý z nich přistupovat ke zprávám na vstupní frontě serveru.

Zprávy ve vstupní frontě serveru jsou stále k dispozici i v případě nestandardního ukončení jednoho ze správců front nebo z administrativních důvodů je třeba zastavit. 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 produktu z/OS v prostředí sysplex, abyste mohli poskytovat vyšší kapacitu a dostupnost serverové aplikace, jak ukazuje [Obrázek 61 na stránce 179](#).

Jedna instance aplikace serveru načte zprávu požadavku ze sdílené fronty a na základě obsahu provede její zpracování a vytvoří výsledek, který se odešle zpět klientovi jako zprávu IBM MQ. Zpráva odpovědi je určena pro frontu odpovědi a správce front uvedenou v deskriptoru zprávy požadavku.

Existuje celá řada voleb, které můžete použít ke konfiguraci návratové cesty. Další informace o těchto volbách viz [“Distribuované fronty a skupiny sdílení front” na stránce 197](#).



Obrázek 61. Více instancí aplikace obsluhující sdílenou frontu

## Obnova partnerských serverů

Chcete-li dále zvýšit dostupnost zpráv ve skupině sdílení front, produkt IBM MQ zjistí, zda je jiný správce front ve skupině odpojen od nestandardně odpojovacího zařízení a dokončí pracovní jednotky, které dosud nejsou ve stavu nevyřízené, pokud je to možné. Tato funkce je známá jako *partnerská obnova*.

Předpokládejme, že správce front byl nestandardně ukončen v místě, kde aplikace načetla zprávu požadavku z fronty v bodu synchronizace, ale dosud neumístila zprávu odezvy nebo nepotvrdila jednotku práce. Jiný správce front v dané skupině sdílení front zjistil selhání a zazálohuje příslušné jednotky práce prováděné ve správci front, který selhal. To znamená, že zpráva požadavku je vložena do fronty požadavků a je k dispozici pro jednu z dalších instancí serveru ke zpracování, aniž by se čekalo na restartování správce front, který selhal.

Pokud produkt IBM MQ nemůže automaticky vyřešit jednotku práce, můžete sdílenou část vyřešit ručně, chcete-li povolit dalšímu zpracování této práce jinému správci front ve skupině sdílení front.

### **z/OS** 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.

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<sup>®</sup> 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 nakonfigurovaná 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 **SCMMAXSIZE** 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 nepříliš 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 uložená 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<sup>®</sup> 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 186
- [“Vylepšený výkon-základní konfigurace”](#) na stránce 192

**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 181.

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 uvezení 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 nevytvorí 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. Zvětšíte velikost struktury aplikace.
2. Přidejte pravidla odlehčování do struktury aplikace tak, aby se zprávy předávaly do SMDS, jakmile se fronta začne zaplňovat.
3. Přidružíte 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<sup>®</sup> 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.

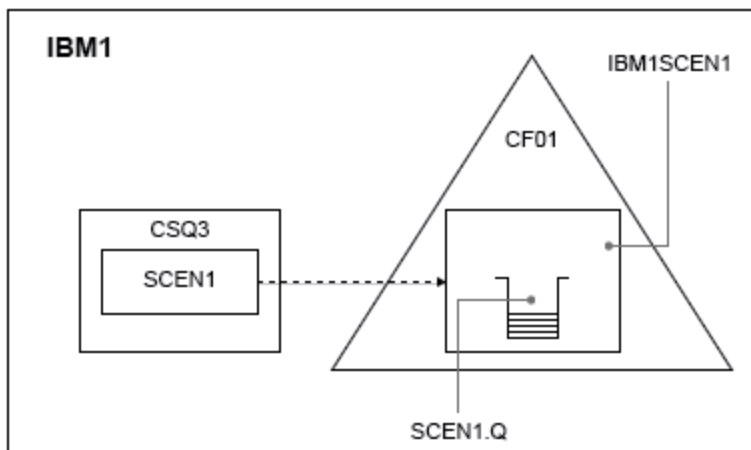
Očekáváte, že se rychlost odesílání zpráv do sdílené fronty v krátkodobém až střednědobém horizontu zdvojnásobí. Ačkoli váš požadavek, aby byl systém schopen tolerovat dvouhodinový výpadek, stále

existuje, v prostředí CF není k dispozici dostatek skutečného úložiště pro zdvojnásobení velikosti struktury. Vzhledem k tomu, že prostředí 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 62 na stránce 187](#).



Obrázek 62. 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)
OFFFLD1SZ(64K) OFFFLD1TH(70)
OFFFLD2SZ(64K) OFFFLD2TH(80)
OFFFLD3SZ(64K) OFFFLD3TH(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. Zadaním 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 179

Použití paměti paměťové třídy (SCM) může být výhodné při použití se sdílenými frontami IBM MQ for z/OS .

 *Přidání SMDS a SCM do počáteční struktury*

Jak přidáte SMDS a SCM pro nouzové úložiště na systému IBM MQ.

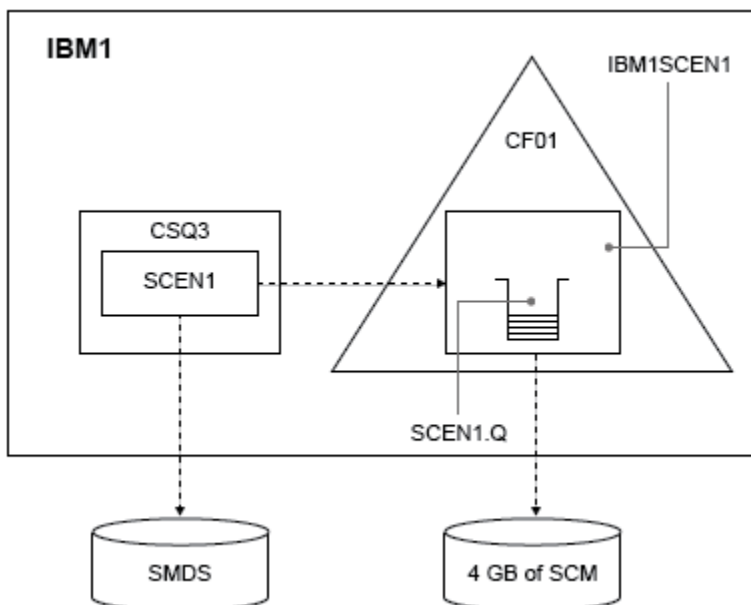
## Informace o této úloze

**Důležité:** IBM z16 je plánováno jako poslední generace produktu IBM Z<sup>®</sup> 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 úlohy používá základní konfiguraci popsanou v části “Nouzové úložiště-základní konfigurace” na stránce 186. Scénář popisuje přidání sdílených datových sad zpráv (SMDS) a poté SCM do počáteční struktury.

Tato konečná konfigurace je znázorněna v souboru Obrázek 63 na stránce 189.



Obrázek 63. 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.SCSQAUTH,DISP=SHR
//SYSUT1 DD DISP=OLD,DSN=CSQSMDS.SCEN1.CSQ3.SMDS
//SYSPRINT DD SYSOUT=*
```

2. Zadáním 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 190.

 *Optimalizace využití paměti paměťové třídy*

Jak zlepšíte využití paměti paměťové třídy (SCM).

**Důležité:** IBM z16 je plánováno jako poslední generace produktu IBM Z<sup>®</sup> 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 : IXCLO053
DISPOSITION : KEEP
ACCESS TIME : NOLIMIT
MAX CONNECTIONS : 32
CONNECTIONS : 1
CONNECTION NAME ID VERSION SYSNAME JOBNAME ASID STATE

CSQEIBM1CSQ301 01 00010059 SC61 CSQ3MSTR 0091 ACTIVE
```

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.

*Tabulka 17. Porovnání výsledků pro scénář nouzového úložiště*

| 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.

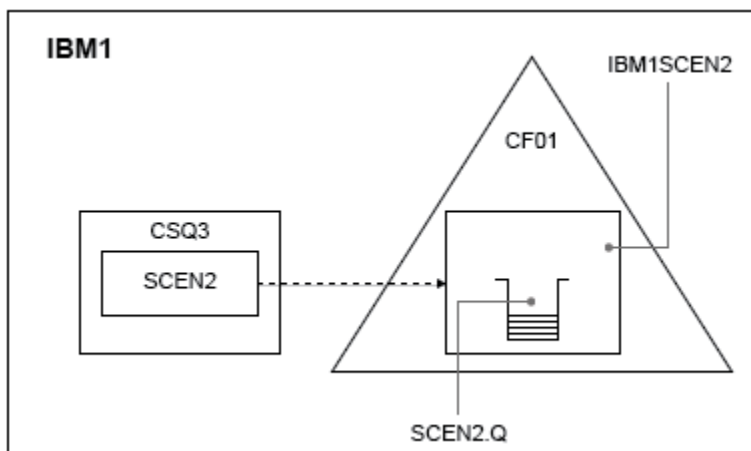
Tento scénář popisuje použití SCM ke zvýšení počtu zpráv, které lze uložit do sdílené fronty, aniž by vznikly náklady na výkon při používání SMDS.

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 ilustrovaná v souboru [Obrázek 64](#) na stránce 193.





Obrázek 64. 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)
OFFLD1SZ(64K) OFFLD1TH(70)
OFFLD2SZ(64K) OFFLD2TH(80)
OFFLD3SZ(64K) OFFLD3TH(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 vložením 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 MQRC\_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 195.

### Související pojmy

[“Použití paměti paměťové třídy se sdílenými frontami”](#) na stránce 179

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.

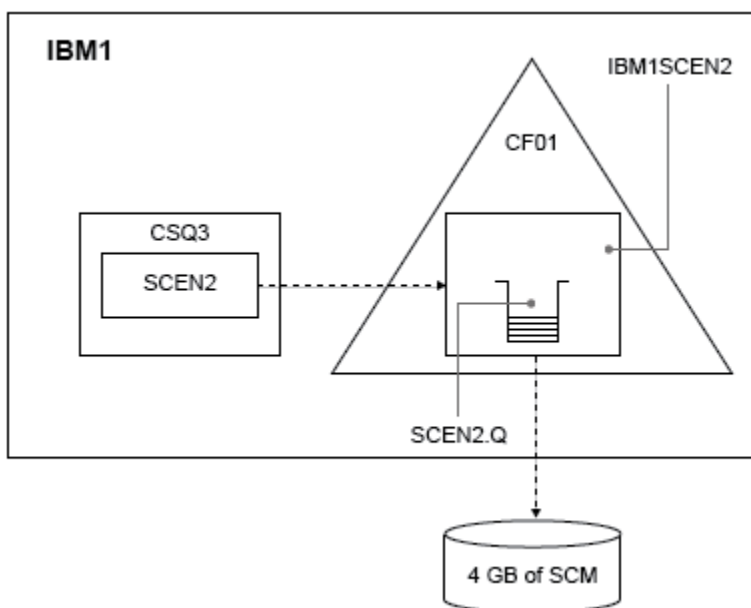
## Informace o této úloze

**Důležité:** IBM z16 je plánováno jako poslední generace produktu IBM Z<sup>®</sup> 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 úlohy používá základní konfiguraci popsanou v části [“Vylepšený výkon-základní konfigurace”](#) na stránce 192. Scénář popisuje přidání SCM do počáteční struktury.

Tato konečná konfigurace je znázorněna v souboru [Obrázek 65](#) na stránce 196.



Obrázek 65. 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 192:
  - 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:

$$(2558 + 44,770 + 204,718) * 256 = 61.5 \text{ MB}$$

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 192.

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](#).

## Distribuované fronty a skupiny sdílení front

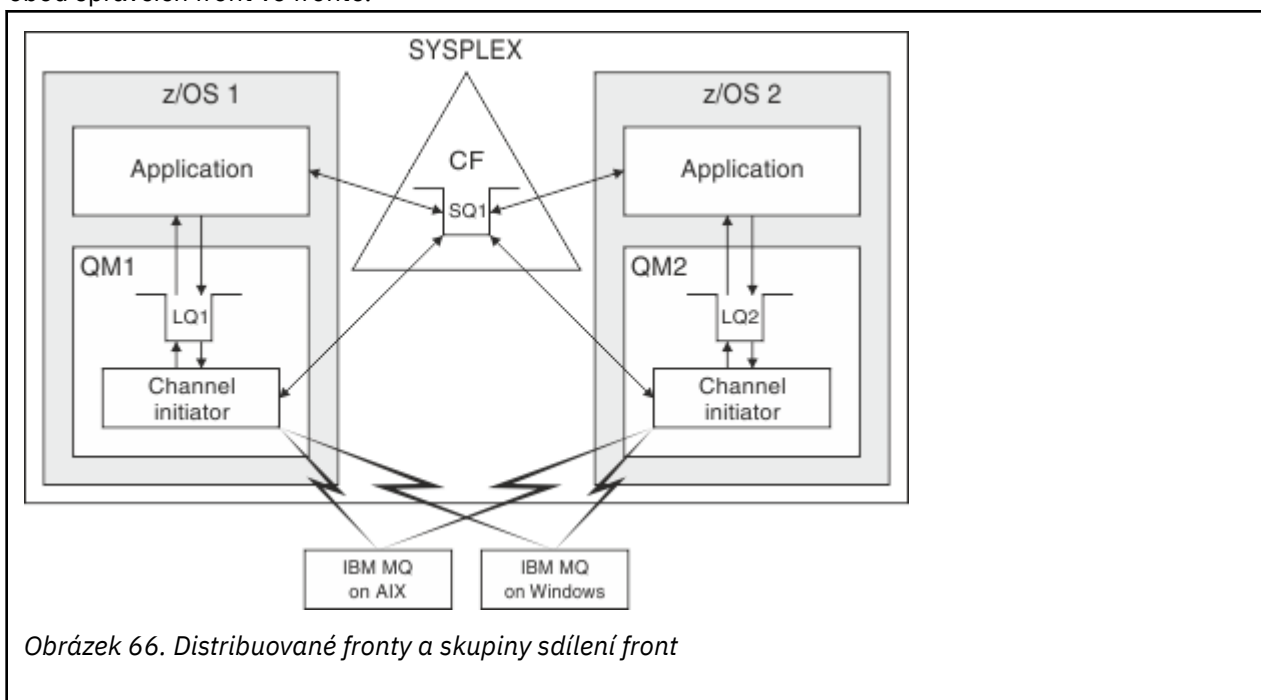
Distribuované fronty a skupiny sdílení front jsou dvě techniky, které můžete použít ke zvýšení dostupnosti vašich aplikačních systémů. Použijte toto téma k vyhledání dalších informací o těchto technikách.

Chcete-li doplnit vysokou dostupnost zpráv ve sdílených frontách, distribuovaná komponenta řazení do fronty produktu IBM MQ má další funkce, které poskytují následující:

- Vyšší dostupnost pro síť.

- Vyšší kapacita pro příchozí síťová připojení ke skupině sdílení front.

Produkt [Obrázek 66](#) na stránce 198 ilustruje distribuované fronty a skupiny sdílení front. Zobrazuje dva správce front v rámci prostředí sysplex, kteří patří do stejné skupiny sdílení front. Oba mohou přistupovat ke sdílené frontě SQ1. Správci front v síti (například v produktu AIX a v produktu Windows ) mohou vkládat zprávy do této fronty prostřednictvím inicializátoru kanálu správce front. Klonování aplikací na obou správcích front ve frontě.



Obrázek 66. Distribuované fronty a skupiny sdílení front

### Související pojmy

“Sdílené kanály” na stránce 198

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 200

Zařazení do front v rámci skupiny umožňuje přenos zpráv mezi správci front v rámci skupiny sdílení front.

“Klastry a skupiny sdílení front” na stránce 201

Toto téma vám pomůže pochopit, jak můžete používat skupiny 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 spuštění 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*.

### Vyrovňování 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 uchovávané 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 uchovávané 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 vyrovnávání 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.

### Použití front v rámci skupiny

Zařazení do front v rámci skupiny umožňuje přenos zpráv mezi správci front v rámci skupiny sdílení front.

Rychlý přenos zpráv mezi správci front v rámci skupiny sdílení front můžete provádět bez definování kanálů. To používá systémovou frontu s názvem `SYSTEM.QSG.TRANSMIT.QUEUE`, což je sdílená přenosová fronta. Každý správce front v rámci skupiny sdílení front spouští úlohu s názvem agent fronty v rámci skupiny, který čeká na příchod zpráv do této fronty, která je určena pro příslušného správce front. Je-li taková zpráva zjištěna, je odebrána z fronty a umístěna do správné cílové fronty.

Je použita standardní pravidla rozpoznávání názvů, ale pokud je povoleno ukládání do front v rámci skupiny (IGQ) a cílový správce front je v rámci skupiny sdílení front, je `SYSTEM.QSG.TRANSMIT.QUEUE` se používá k přenosu zprávy na správného správce cílové fronty místo použití přenosové fronty a kanálu.

Ukládání do front v rámci skupiny lze povolit prostřednictvím atributu správce front. Zařazení do front v rámci skupiny přesouvá přechodné zprávy mimo synchronizační bod a trvalé zprávy v rámci synchronizačního bodu. Pokud narazí na problém doručující zprávy do cílové fronty, fronty v rámci skupiny se ji pokusí umístit do fronty nedoručených zpráv. Je-li fronta nedoručených zpráv plná nebo nedefinovaná, budou přechodné zprávy zrušeny, ale trvalé zprávy jsou vráceny a vráceny do systému `SYSTEM.QSG.TRANSMIT.QUEUE` a agent IGQ se pokusí doručit zprávy, dokud nebude úspěšný.

Příchozí sdílený kanál, který přijímá zprávu určenou pro frontu v jiném správci front v rámci skupiny sdílení front, může používat řazení do fronty v rámci skupiny pro *směrovací uzel* do správného místa určení.



Může nastat situace, kdy má lokální správce front vložit zprávu přímo do cílové fronty, je-li cílová fronta sdílenou frontou, a nikoli první, která byla přenesena do cílového správce front. K řízení tohoto parametru můžete použít atribut správce front SQQMNAME. Nastavíte-li hodnotu proměnné SQQMNAME na hodnotu USE, bude příkaz MQOPEN proveden ve správci front uvedeném názvem ObjectQMgr. Je-li však cílová fronta sdílenou frontou a nastavíte hodnotu proměnné SQQMNAME na IGNORE a ObjectQMgrnázev 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 do fronty zprávu, bude zpráva předána do určeného názvu ObjectQMgrbuď prostřednictvím IGQ, nebo kanálu produktu MQ.

Zařazení do fronty v rámci skupiny (IGQ) podporuje velké zprávy, přičemž největší hodnota je 100 MB *minus* délka záhlaví přenosové fronty.

Použijete-li tuto funkci, uživatelé musí mít stejný přístup k frontám ve všech správci front ve skupině sdílení front.

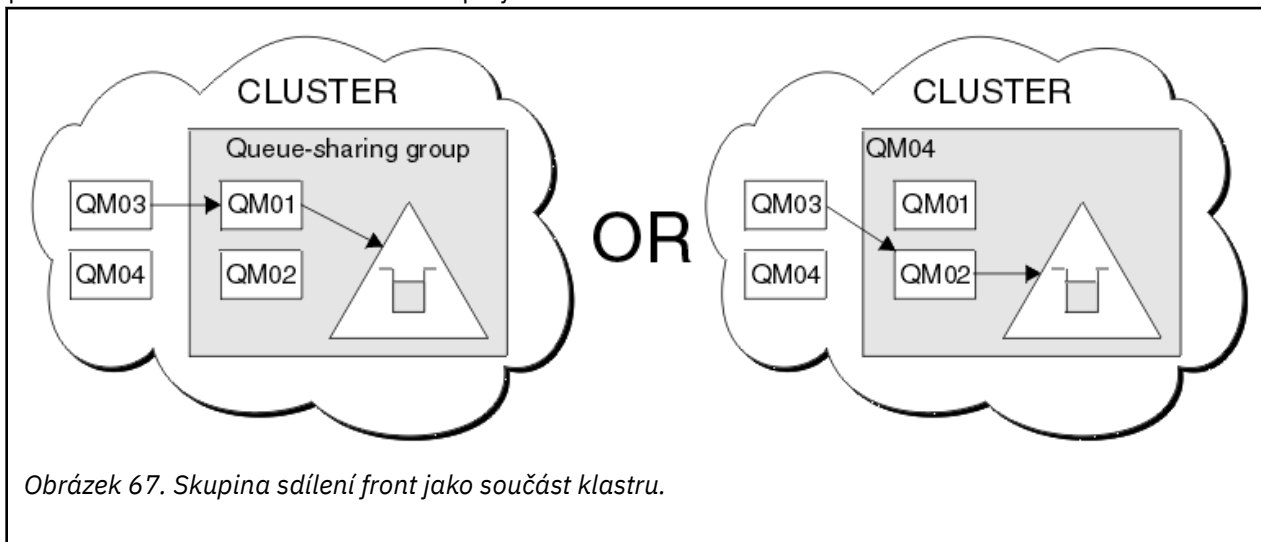
## **z/OS** **Klastry a skupiny sdílení front**

Toto téma vám pomůže pochopit, jak můžete používat skupiny sdílení front s klastry.

Sdílenou frontu můžete zpřístupnit pro klastr v jediné definici. Chcete-li to provést, zadejte při definování sdílené fronty název klastru.

Uživatelé v síti vidí sdílenou frontu jako hostitele každého správce front v rámci skupiny sdílení front (sdílená fronta není deklarována jako hostovaná skupinou sdílení front). Klienti mohou spouštět relace s libovolnými členy skupiny sdílení front, aby ukládali zprávy do stejné sdílené fronty.

Příkaz [Obrázek 67 na stránce 201](#) zobrazuje, jak členové klastru mohou přistupovat ke sdílené frontě prostřednictvím libovolného člena skupiny sdílení front.



Obrázek 67. Skupina sdílení front jako součást klastru.

## **z/OS** **Ovlivňování distribuce pracovní zátěže se sdílenými frontami**

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. Rozdělení pracovní zátěže ve skupině sdílení front (QSG) však může být ovlivněno *způsobem založeným na stažení*. Volba, která služby správce front obsluhuje frontu (přijímá zprávu napsanou do sdílené fronty), je ovlivněna dostupnou kapacitou zpracování každého správce front ve skupině sdílení front a cíli správy pracovní zátěže definovanými v rámci prostředí sysplex.

Je však důležité uvědomit si, že správce front, který provádí MQPUT zprávy, může mít také velký vliv při rozhodování o tom, který správce front dostane zprávu.

## Lokální správce front je s větší pravděpodobností proveden příkaz MQGET.

V případě aplikace provádějící požadavek MQPUT je správce lokální fronty připojen jako správce front, k němuž je aplikace připojena.

Přesně to, který správce front obsluhuje MQGET zprávy provedením příkazu MQGET v zastoupení při získávání aplikace, je ovlivněn následujícími pokyny.

Je-li zpráva vložena do prázdné sdílené fronty, je lokální správce front obvykle odeslán dříve, než se oznámí kterýkoli z ostatních správců front ve skupině sdílení front. Je-li lokální správce front v pozici, aby mohl zprávu zpracovat, obdrží od prostředku CF (coupling facility) seznam oznámení o přechodu seznamu před jiným správcem front v rámci skupiny sdílení front. (Oznámení o přechodu seznamu je oznámení, že sdílená fronta změnila stav z prázdné na neprázdný.)

Možné scénáře, v tomto případě, jsou následující:

1. Volání MQPUT dočasné zprávy mimo synchronizační bod a *rychlé odeslání pro čekání metody getter*.

Existuje-li aplikace s volbou *MQGET s čekáním* na lokálním správcem front pro danou frontu, bude příkaz MQPUT zprávy předán přímo do vyrovnávací paměti aplikace a nebude do ní zapsán. To platí pro sdílené a nesdílené fronty. Tato funkce se často nazývá *fast put to a waiting getter* mechanismus. V případě sdílených front není oznámen žádný další správce front v QSG, protože neexistuje žádný přechod z prázdné do neprázdné fronty. To znamená, že například za předpokladu, že tento správce front může poskytovat služby všechny operace vložení z této aplikace a za předpokladu, že žádné další aplikace nevkládají do fronty žádné zprávy, nebude žádný jiný správce front v rámci skupiny sdílení front pomáhat při vyprazdňování této fronty. Pokud však není k dispozici příkaz MQGET s čekáním na lokálního správce front a zpráva je vložena do sdílené fronty, bude prostředek CF v souladu se svými pravidly pro oznámení změny seznamu upozorňovat ostatní správce front v dané skupině sdílení front.

2. MQPUT trvalé nebo synchronizující zprávy v rámci synchronizačního bodu.

Je-li v tomto případě aplikace s volbou *MQGET s čekáním* na lokálním správcem front, bude zpráva vložena do sdílené fronty a prostředek CF informuje ostatní správce front ve skupině sdílení front podle svých pravidel pro oznámení přechodů seznamu. Lokální správce front však nečeká na oznámení o přechodu z prostředku CF, ale nejprve všechny lokální příkazy *MQGET s čekáním* a obvykle provede získání této zprávy jménem aplikace předtím, než může kterýkoli jiný správce front v rámci skupiny sdílení front odpovědět na oznámení prostředku CF. To je závislé na tom, jak je lokální správce front zaneprázdněn. Jinak každý správce front oznámený prostředkem CF kvůli příchodu zprávy do prázdné fronty se pokusí službu získat jako první. První správce front, který má odpovědět, zpracuje novou zprávu.

3. Pokud fronta není vyčerpána zprávy, kde prostředek CF odeslal oznámení o změně stavu z prázdné na neprázdný pro danou frontu, budou mít všechny připojené správce front příležitost pomoci při zpracování fronty. V této události se říká, že pracovní zátěž je *založená na tahu*.

Tento návrh umožňuje lepší výkon v čisté tažné distribuci pracovní zátěže. Cílem je využít služeb vysoké dostupnosti, které jsou nabízeny ve frontách v rámci prostředku CF a které umožňují správcem front provádět MQGET bez nutnosti odkazovat na prostředek CF, a aby bylo možné zpracovat pracovní zátěž zpráv co nejefektivněji.

Alternativní přístupy mohou být přijaty tam, kde je kladen důraz na rovnováhu pracovní zátěže než dříve popisovaná zlepšení výkonnosti. Například je třeba zajistit, aby žádná ze získání aplikací nebyla připojena ke stejnému správcem front, ke kterému je aplikace připojena. Při použití tohoto návrhu jsou všechny zprávy vloženy do fronty a všichni správce front v QSG jsou upozorněni, když se fronta přesune z prázdné do neprázdné, v souladu s algoritmem CF pro zpracování takových přechodů. Kromě toho nelze použít mechanismus *fast put to waiting Getter*.

## Kde najdete další informace o sdílených frontách a skupinách sdílení front

Pomocí tabulky v tomto tématu naleznete více informací o tom, jak produkt IBM MQ for z/OS používá sdílené fronty a skupiny sdílení front.

Tabulka 18. Kde najdete další informace o sdílených frontách a skupinách sdílení front

| Téma                                                                                 | Kde hledat                                                                       |
|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------|
| Obnova skupiny sdílení front                                                         | <a href="#">“Zotavení a restartování v systému z/OS” na stránce 242</a>          |
| Zabezpečení skupiny sdílení front                                                    | <a href="#">“Koncepte zabezpečení v produktu IBM MQ for z/OS” na stránce 258</a> |
| Definice soukromých a globálních objektů<br>Směrování příkazů do jiné fronty správci | <a href="#">Zadání příkazů</a>                                                   |
| Plánování prostředku CF prostředí                                                    | <a href="#">Definování prostředků prostředku Coupling Facility</a>               |
| Plánování prostředí SMDS                                                             | <a href="#">Plánování prostředí sdílených datových sad zpráv (SMDS)</a>          |
| Plánování<br>Prostředí produktu Db2                                                  | <a href="#">Plánování vašeho prostředí Db2</a>                                   |
| Nastavení sdílených front<br>Parametry systému                                       | <a href="#">“Sdílené fronty a skupiny sdílení front” na stránce 158</a>          |
| Obslužné programy<br>Migrace front                                                   | <a href="#">Použití obslužných programů produktu IBM MQ</a>                      |
| Zprávy konzoly                                                                       | <a href="#">Zprávy pro IBM MQ for z/OS</a>                                       |
| Příkazy MQSC                                                                         | <a href="#">Příkazy MQSC</a>                                                     |
| IBM MQKlastry                                                                        | <a href="#">Konfigurace klastru správců front</a>                                |
| IBM MQ distribuované fronty<br>Názvy kanálů                                          | <a href="#">Úvod do distribuované správy front</a>                               |
| Zapisování aplikací                                                                  | <a href="#">Přehled návrhu aplikací</a>                                          |
| Volání MQCONN                                                                        | <a href="#">MQCONN</a>                                                           |

z/OS

## Použití front v rámci skupiny

Tento oddíl popisuje řazení do front v rámci skupiny, funkce IBM MQ for z/OS je 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 158](#).

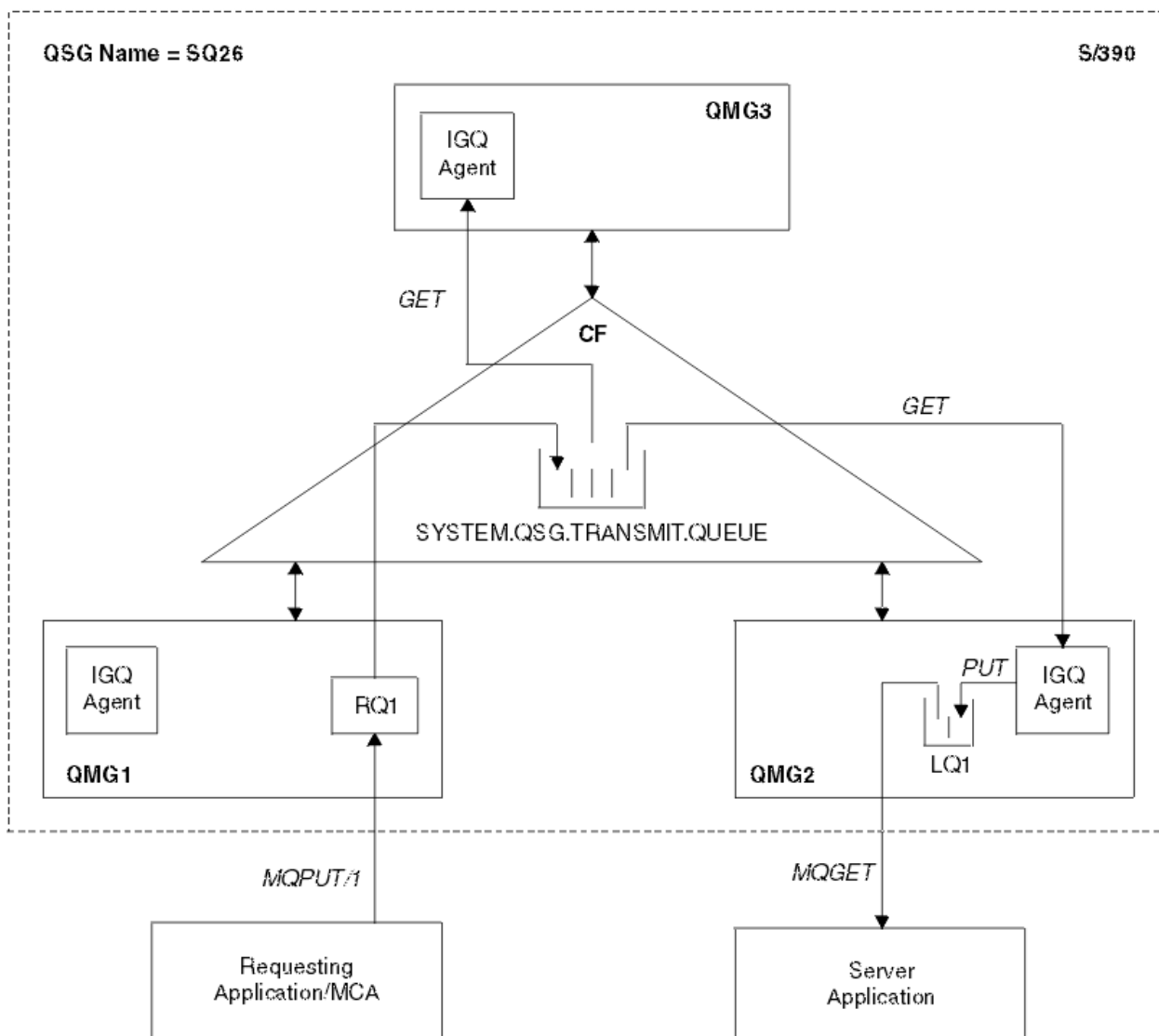
z/OS

## Koncepce front v rámci skupiny

Řazení do front v rámci skupiny lze použít k zajištění efektivnějšího doručení malých zpráv do front uložených ve vzdálených správcích front v rámci skupiny sdílení front.

Řazení do front v rámci skupiny (IGQ) může způsobit potenciálně rychlý a levnější přenos malých zpráv mezi správci front v rámci skupiny sdílení front (QSG), aniž by bylo nutné definovat kanály mezi správci front.

Následující diagram znázorňuje typický příklad použití front v rámci skupiny.



Obrázek 68. Příklad řazení do front v rámci skupiny

Diagram zobrazuje:

- Agenti IGQ běží na třech správcích front (QMG1, QMG2a QMG3), které jsou definovány do skupiny sdílení front s názvem SQ26.
- Sdílená přenosová fronta SYSTEM.QSG.TRANSMIT.QUEUE , která je definována v prostředku CF (coupling facility).
- Definice vzdálené fronty, která je definována ve správci front QMG1.
- Lokální fronta, která je definovaná ve správci front QMG2.
- Žádost o aplikaci (tato aplikace může být agentem MCA (Message Channel Agent), která je připojena ke správci front QMG1.
- Serverová aplikace, která je připojena ke správci front QMG2.
- Zpráva požadavku je umístěna na SYSTEM.QSG.TRANSMIT.QUEUE.

## Řazení do front v rámci skupiny a agent front v rámci skupiny

Agent IGQ se spustí během inicializace správce front. Když aplikace otevřou a umíjí zprávy do vzdálených front, určuje lokální správce front, zda se pro přenos zpráv použije řazení do front v rámci skupiny. Má-li být použito řazení do fronty v rámci skupiny, umístí lokální správce front zprávu do systému SYSTEM.QSG.TRANSMIT.QUEUE. Agent IGQ v cílovém vzdáleném správci front načte zprávu a umístí ji do cílové fronty.

### Terminologie vytváření front v rámci skupiny

Vysvětlení terminologie: fronty v rámci skupiny, sdílená přenosová fronta pro použití ve frontě v rámci skupiny a správce front v rámci skupiny.

### Použití front v rámci skupiny

Zařazení do front v rámci skupiny může mít za následek potenciálně rychlý a méně nákladný přenos zpráv mezi správci front v rámci skupiny sdílení front, aniž by bylo nutné definovat kanály.

### Sdílená přenosová fronta pro použití ve frontě v rámci skupiny

Každá skupina sdílení front má sdílenou přenosovou frontu s názvem SYSTEM.QSG.TRANSMIT.QUEUE pro použití ve frontě v rámci skupiny. Je-li povoleno ukládání do front v rámci skupiny, SYSTEM.QSG.TRANSMIT.QUEUE se zobrazí v cestě rozpoznání názvu při otevírání vzdálených front. Když aplikace (včetně agentů kanálu zpráv (MCA)) umísťuje zprávy do vzdálené fronty, správce lokální fronty určuje způsobilost zpráv pro rychlý přenos a umísťuje je do systému SYSTEM.QSG.TRANSMIT.QUEUE.

### Agent zařazení do fronty 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 načítá vhodné zprávy z této fronty a doručuje je do cílových front.

Agent IGQ pro každého správce front je vždy spuštěn, protože správce front používá řazení v rámci skupiny pro vlastní vnitřní zpracování.

### Výhody řazení do front v rámci skupiny

Výhody ukládání do front v rámci skupiny jsou: zredukované definice systému, snížená správa systému, vylepšený výkon, podpora migrace a doručování zpráv při vícenásobným přeskokování mezi správci front v rámci skupiny sdílení front.

Výhody ukládání do front v rámci skupiny jsou:

#### **Redukované definice systému**

Zařazení do front v rámci skupiny odstraňuje potřebu definovat kanály mezi správci front ve skupině sdílení front.

#### **snížení administrace systému**

Vzhledem k tomu, že mezi správci front ve skupině sdílení front nejsou definovány žádné kanály, není pro správu kanálů nutné žádný požadavek.

#### **Zlepšení výkonu**

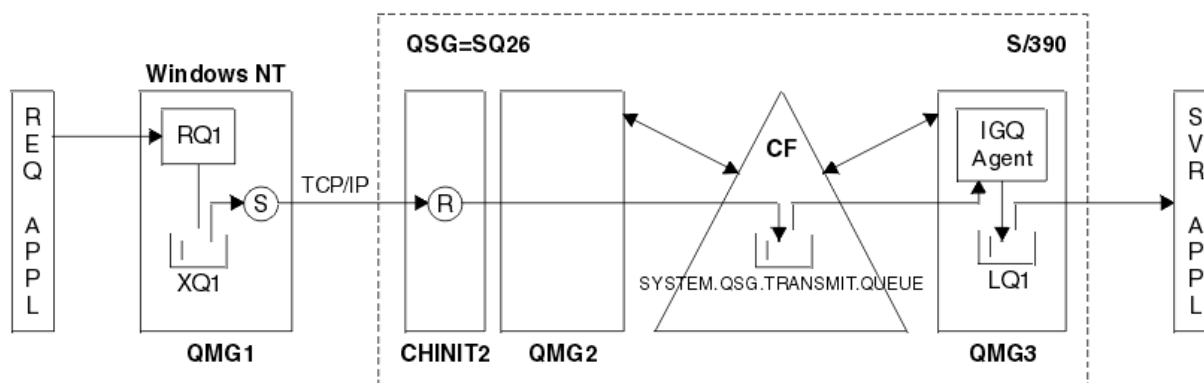
Vzhledem k tomu, že pro doručení zprávy do cílové fronty je zapotřebí pouze jeden agent IGQ (místo dvou zprostředkujících odesílacích a přijímacích agentů), může být doručování zpráv používajících řazení do front v rámci skupiny levnější než doručování zpráv pomocí kanálů. Ve frontě front v rámci skupiny je pouze přijímající komponenta, protože byla odebrána potřeba odesílací komponenty. Toto uložení je způsobeno tím, že zpráva je k dispozici pro agenta IGQ v cílovém správci front k doručení do cílové fronty po dokončení operace vložení v lokálním správci front a v případě zpráv vložených do rozsahu synchronizačního bodu potvrzena.

## Podporuje migraci

Aplikace externí do skupiny 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 to, že zprávy přicházející do přijímacího kanálu, které jsou určeny pro frontu ve vzdáleném správci front, mohou být transparentně odeslány do cílové fronty pomocí front v rámci skupiny. Toto zařízení umožňuje implementovat aplikace mezi skupinou sdílení front bez nutnosti měnit všechny systémy, které jsou externí pro skupinu sdílení front.

Typická konfigurace je ilustrována následujícím diagramem, v němž:

- Žádající aplikace připojená ke správci front QMG1 potřebuje odeslat zprávu do lokální fronty ve správci front QMG3.
- Správce front QMG1 je připojen pouze ke správci front QMG2.
- Správci front QMG2 a QMG3, které byly dříve připojeny pomocí kanálů, jsou nyní členy skupiny sdílení front SQ26.



Obrázek 69. Příklad podpory migrace

Tok operací je následující:

1. Žádající aplikace vloží do vzdálené definice fronty RQ1 zprávu určenou pro lokální frontu LQ1 ve vzdáleném správci front QMG3.
2. Správce front QMG1, který je spuštěn na pracovní stanici s produktem Windows NT, umístí zprávu do přenosové fronty XQ1.
3. Odesílatel MCA (S) na QM1 přenáší zprávu pomocí TCP/IP do přijímače MCA (R) na inicializátoru kanálu CHINIT2.
4. Přijímací MCA (R) na iniciátoru 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. Serverová aplikace načte zprávu z cílové lokální fronty a zpracuje ji.

## Doručování zpráv při vícenásobným přeskakování mezi správci front ve skupině sdílení front

Předchozí diagram v části Podporuje migraci také ilustruje doručení zpráv při vícenásobným přeskakování mezi správci front ve skupině sdílení front. Zprávy přicházející do správce front v rámci skupiny sdílení front, které jsou určeny pro frontu v jiném správci front ve skupině sdílení front, lze snadno přenést do cílové fronty v cílovém správci front pomocí front v rámci skupiny.

## z/OS Omezení řazení do front v rámci skupiny

Omezení front v rámci skupiny jsou: zprávy vhodné pro přenos pomocí front v rámci skupiny, počet agentů front v rámci skupiny v rámci správce front a spuštění a zastavení agenta pro řazení do front v rámci skupiny.

Toto téma popisuje omezení front v rámci skupiny.

## Zprávy způsobilé pro přenos pomocí front v rámci skupiny

Vzhledem k tomu, že řazení do front v rámci skupiny používá sdílenou přenosovou frontu, která je definována v prostředku Coupling Facility (CF), je řazení do front ve skupině omezeno na doručení zpráv o maximální podporované délce zpráv pro sdílené fronty po odečtení délky záhlaví přenosové fronty (MQXQH).

## Počet agentů front v rámci skupiny pro správce front

V rámci skupiny sdílení front je spuštěn pouze jeden agent IGQ na správce front.

## Spuštění a zastavení agenta intra-group queuing Agent

Agent IGQ se spustí během inicializace správce front a ukončí se během ukončování činnosti správce front. Je navržen tak, aby byl dlouhodobě spuštěný, zotavující se (v případě abnormálního ukončení), úloha. Pokud dojde k chybě s definicí SYSTEM.QSG.TRANSMIT.QUEUE (například, je-li tato fronta zablokována), agent IGQ bude pokračovat v pokusech. Pokud agent IGQ zjistí chybu, která má za následek běžné ukončení činnosti agenta, zatímco je správce front stále aktivní, lze jej restartovat zadáním příkazu ALTER QMGR IGQ (ENABLED). Tento příkaz se vyhýbá potřebě recyklovat správce front.

## Nastavení atributu IGQ správce front na hodnotu ENABLED nebo DISABLED

Je-li atribut IGQ správce front nastaven na hodnotu ENABLED nebo DISABLED, mohou být existující popisovače objektů zneplatněny s kódem příčiny MQRC\_OBJECT\_CHANGED. Další informace najdete v tématu [Začínáme s frontami v rámci skupiny](#).

## Začínáme s frontami v rámci skupiny

Můžete povolit, zakázat a používat řazení do front ve skupině, jak je popsáno v tomto tématu.

### Povolení řazení do front v rámci skupiny

Chcete-li ve správci front povolit řazení do front v rámci skupiny, je třeba provést následující kroky:

- Definujte sdílenou přenosovou frontu s názvem SYSTEM.QSG.TRANSMIT.QUEUE. Definice této fronty může být nalezena v souboru thlqual.SCSQPROC(CSQ4INSS), ukázka CSQINP2 pro objekty SYSTEM pro skupiny sdílení front. Tato fronta musí být definována se správnými atributy, jak je uvedeno v souboru thlqual.SCSQPROC(CSQ4INSS), pro řádné fungování front ve skupině.
- Protože se agent IGQ vždy spouští při inicializaci správce front, je pro zpracování příchozích zpráv vždy k dispozici řazení do front v rámci skupiny. Agent IGQ zpracuje všechny zprávy, které jsou umístěny na SYSTEM.QSG.TRANSMIT.QUEUE. Chcete-li však povolit řazení do fronty v rámci skupiny pro odchozí zpracování, musí být atribut IGQ správce front nastaven na hodnotu ENABLED.

**Důležité:** Je-li atribut IGQ správce front nastaven na hodnotu ENABLED, mohou být existující popisovače objektů zneplatněny s kódem příčiny MQRC\_OBJECT\_CHANGED. Další informace viz [“Specifické vlastnosti front v rámci skupiny”](#) na stránce 215. Jak je popsáno v sekci 'Odezva programátora' pro tento kód příčiny, aplikace musí být kódovány, aby zvládl tuto situaci (viz [2041 \(07F9\) \(RC2041\): MQRC\\_OBJECT\\_CHANGED](#) pro více podrobností).

Kromě toho je IGQ navržen jako dlouhotrvající a sebeobnova, která začíná během inicializace a končí ukončením práce systému, viz [“Omezení řazení do front v rámci skupiny”](#) na stránce 206, kde jsou další informace.

### Zakázání front v rámci skupiny

Chcete-li zakázat řazení do fronty v rámci skupiny pro odchozí přenos zpráv, nastavte atribut IGQ správce front na hodnotu DISABLED. Je-li pro určitého správce front zakázáno řazení do fronty v rámci skupiny, může agent IGQ v daném správci front stále zpracovávat příchozí zprávy, které byly umístěny do systému SYSTEM.QSG.TRANSMIT.QUEUE u správce front, který má pro odchozí přenos povoleno ukládání do front v rámci skupiny.

**Důležité:** Je-li atribut IGQ správce front nastaven na hodnotu ENABLED, mohou být existující popisovače objektů zneplatněny s kódem příčiny MQRC\_OBJECT\_CHANGED. Další informace viz [“Specifické vlastnosti front v rámci skupiny”](#) na stránce 215. Jak je popsáno v sekci 'Odezva programátora' pro tento kód příčiny, aplikace musí být kódovány, aby zvládl tuto situaci (viz [2041 \(07F9\) \(RC2041\): MQRC\\_OBJECT\\_CHANGED](#) pro více podrobností).

Kromě toho je IGQ navržen jako dlouhotrvající a sebeobnova, která začíná během inicializace a končí ukončením práce systému, viz [“Omezení řazení do front v rámci skupiny”](#) na stránce 206 , kde jsou další informace.

### **Použití front v rámci skupiny**

Je-li povoleno ukládání do front v rámci skupiny, je k dispozici pro použití a správce front jej bude používat vždy, když je to možné. To znamená, že když aplikace vkládá zprávu do vzdálené definice fronty, do plně kvalifikované vzdálené fronty nebo do fronty klastru, správce front určí, zda je zpráva vhodná k doručení s využitím front v rámci skupiny, a pokud ano, umístí zprávu do systému SYSTEM.QSG.TRANSMIT.QUEUE. Není třeba měnit uživatelské aplikace ani aplikační fronty, protože u vhodných zpráv správce front používá SYSTEM.QSG.TRANSMIT.QUEUE, přednostně do jakékoli jiné přenosové fronty.

### **z/OS Konfigurace front v rámci skupiny**

Kromě typické konfigurace front v rámci skupiny jsou možné další konfigurace.

[Obrázek 68 na stránce 204](#) popisuje typickou konfiguraci.

#### **Související pojmy**

[“distribuované řazení do front s frontou v rámci skupiny \(více doručovacích cest\)”](#) na stránce 208

U aplikací, které zpracovávají krátké zprávy, může být možné nakonfigurovat řazení do fronty v rámci skupiny pouze pro doručování zpráv mezi správci front v rámci skupiny sdílení front.

[“Klastrování s frontami v rámci skupiny \(více doručovacích cest\)”](#) na stránce 210

Je možné nakonfigurovat správce front tak, aby byli v klastru, stejně jako ve skupině sdílení front.

[“Klastrování, řazení do front v rámci skupiny a distribuované fronty”](#) na stránce 212

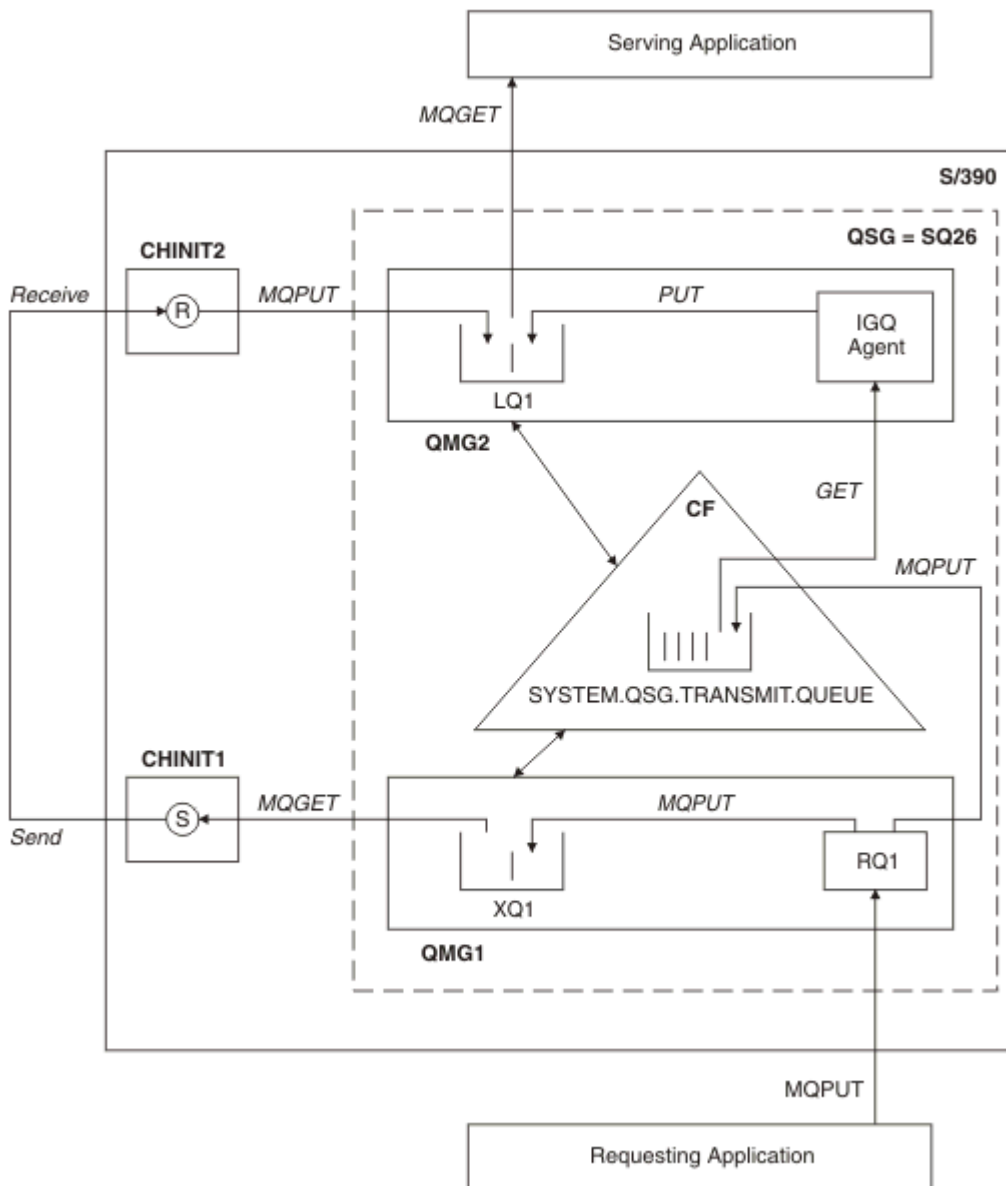
Je možné nakonfigurovat správce front, který je členem klastru, stejně jako skupinu sdílení front a je připojen k distribuovanému správci front s použitím dvojice kanálu odesílatele/příjemce.

### **z/OS distribuované řazení do front s frontou v rámci skupiny (více doručovacích cest)**

U aplikací, které zpracovávají krátké zprávy, může být možné nakonfigurovat řazení do fronty v rámci skupiny pouze pro doručování zpráv mezi správci front v rámci skupiny sdílení front.

Volba ukládání do front v rámci skupiny může být řízena úrovní typu CFSTRUCT. (3 místo 4 nebo 5). Maximální délka zprávy, která je nastavena na SYSTEM.QSQ.TRANSMIT.QUEUE.





Obrázek 70. Příklad konfigurace

## Otevřít/Vložit zpracování

1. Je důležité si uvědomit, že když žádající aplikace otevře vzdálenou frontu `RQ1`, dojde k rozpoznání názvu pro nesdílenou přenosovou frontu `XQ1` i pro sdílenou přenosovou frontu `SYSTEM.QSG.TRANSMIT.QUEUE`.
2. Když žádající aplikace vloží zprávu do vzdálené fronty na základě toho, zda je povoleno řazení do front v rámci skupiny pro odchozí přenos ve správci front, a pokud jde o charakteristiky 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 `XQ1` a všechny malé zprávy do přenosové fronty `SYSTEM.QSG.TRANSMIT.QUEUE`.
3. Je-li přenosová fronta `XQ1` plná nebo není k dispozici, požadavky na velké zprávy se synchronně nezdaří s vhodným návratovým kódem a kódem příčiny. Požadavky na malé zprávy však budou i nadále úspěšné a budou umístěny do přenosové fronty `SYSTEM.QSG.TRANSMIT.QUEUE`.
4. Je-li přenosová fronta `SYSTEM.QSG.TRANSMIT.QUEUE` je plná nebo nemůže být vložena, požadavky na malé zprávy se synchronně nezdaří s vhodným návratovým kódem a kódem příčiny. Požadavky na

velké zprávy jsou však nadále úspěšné a jsou umístěny do přenosové fronty XQ1. V tomto případě není proveden 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 vkládá zprávy do přenosové fronty XQ1.
3. Odesílatel MCA (S) ve správci front QMG1 načte zprávy z přenosové fronty XQ1 a odešle je do správce front QMG2.
4. Příjemce MCA (R) ve správci front QMG2 přijímá zprávy a umísťuje je do cílové fronty LQ1.
5. Obsluha aplikace načte a poté zpracuje zprávy z fronty LQ1.

### **Tok pro malé zprávy**

1. Požadující aplikace vkládá malé zprávy do vzdálené fronty RQ1.
2. Správce front QMG1 vkládá zprávy do přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE.
3. IGQ ve správci front QMG2 načte zprávy a umístí je do cílové fronty LQ1.
4. Aplikace obsluhy načítá zprávy z fronty LQ1.

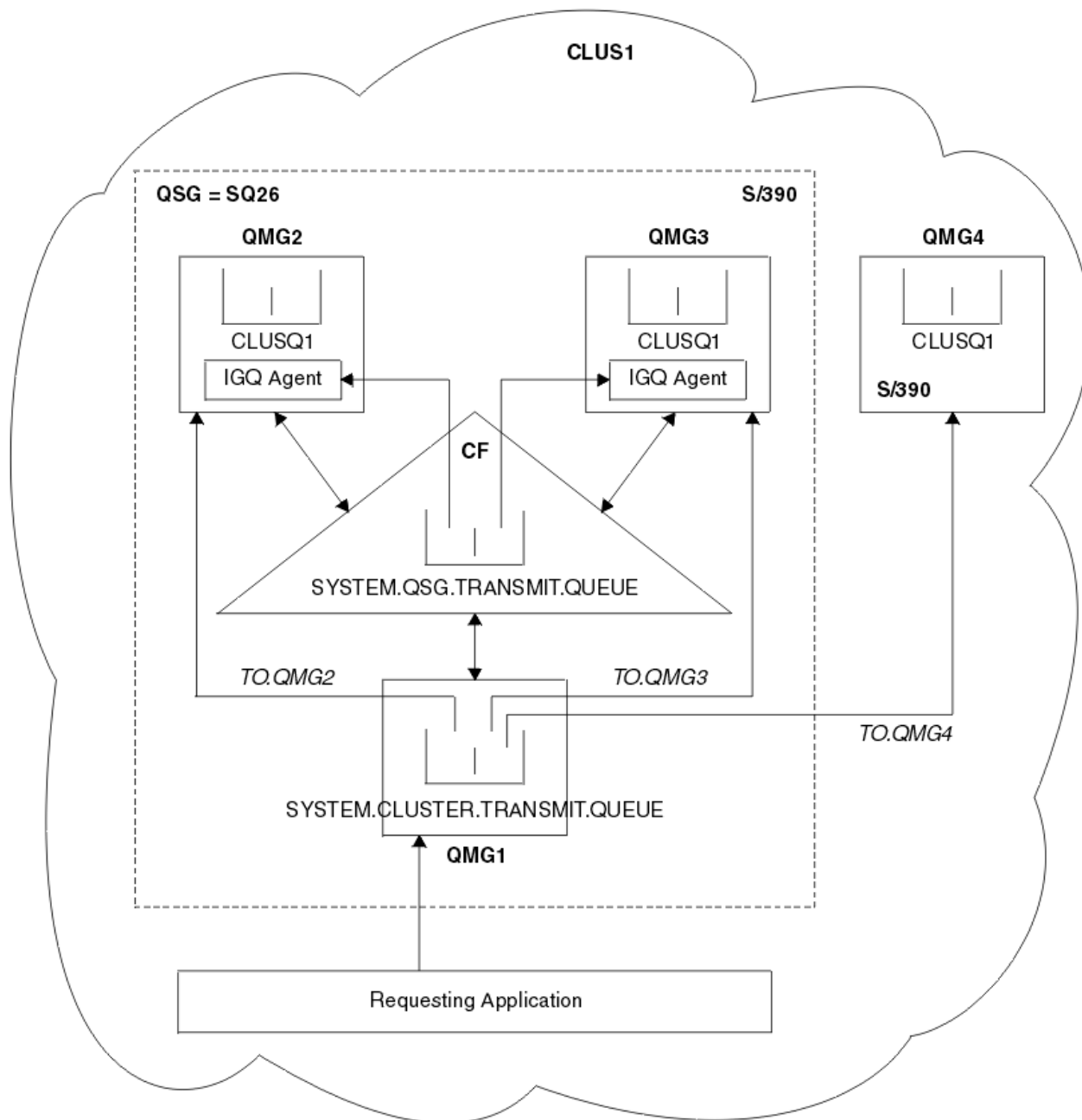
### **Body k poznámce**

1. Žádající aplikace si nemusí být vědoma základního mechanismu použitého pro doručování zpráv.
2. Potenciálně rychlejší mechanismus doručování zpráv lze dosáhnout pro malé zprávy.
3. Pro doručení zprávy je k dispozici více cest (to znamená normální trasu kanálu a směrování ve frontě v rámci skupiny).
4. Přenosová cesta front v rámci skupiny je potenciálně rychlejší, je vybrána jako předvolba pro normální přenosovou cestu kanálu. V závislosti na charakteristice zprávy může být doručení zprávy rozděleno mezi dvě cesty. Proto mohou být zprávy doručovány mimo pořadí (ačkoliv je toto doručení také možné, pokud jsou zprávy doručovány pouze pomocí normální přenosové cesty kanálu).
5. Když byla zvolena trasa a zprávy byly umístěny do přenosových front, použije se pro doručení zprávy pouze zvolená přenosová cesta. Všechny nezpracované zprávy na SYSTEM.QSG.TRANSMIT.QUEUE není přesměrována do přenosové fronty XQ1.

### **Klastrování s frontami v rámci skupiny (více doručovacích cest)**

Je možné nakonfigurovat správce front tak, aby byli v klastru, stejně jako ve skupině sdílení front.

Jsou-li zprávy odesílány do fronty klastru a lokální a vzdálené cílové správce front jsou ve stejné skupině sdílení front, bude pro doručování malých zpráv použito řazení do fronty v rámci skupiny pro doručování malých zpráv (pomocí příkazu SYSTEM.QSG.TRANSMIT.QUEUE) a doručení velkých zpráv, pokud fronta front ve skupině podporuje velikost zprávy. Také SYSTEM.CLUSTER.TRANSMIT.QUEUE se používá pro doručování zpráv do libovolného správce front, který je v klastru, ale mimo skupinu sdílení front. Tato konfigurace ilustruje následující diagram (nejsou zobrazeny iniciátory kanálu).



Obrázek 71. Příklad klastrování při řazení do fronty v rámci skupiny

Diagram zobrazuje:

- Čtyři správci front z/OS QMG1, QMG2, QMG3 a QMG4 konfigurované v klastru CLUS1.
- Správci front QMG1, QMG2 a QMG3 konfigurované ve skupině sdílení front SQ26.
- Agenti IGQ běží na správci front QMG2 a QMG3.
- Lokální SYSTEM.CLUSTER.TRANSMIT.QUEUE definovaná v QMG1.

**Poznámka:** Pro srozumitelnost je SYSTEM.CLUSTER.TRANSMIT.QUEUE na ostatních správci front nebyla zobrazena.

- Sdílený SYSTEM.QSG.TRANSMIT.QUEUE definovaná v prostředku CF, který je ve struktuře IBM MQ nakonfigurované s atributem RECOVER (YES) s 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 ve správcích front QMG2, QMG3a QMG4.

Předpokládejme, že žádající aplikace otevře frontu klastru s volbou MQOO\_BIND\_NOT\_FIXED tak, aby byl cílový správce front pro frontu klastru vybrán v čase vložení.

Je-li vybraný cílový správce front QMG2:

- Všechny velké zprávy odeslané žádající aplikací jsou:
  - Vložte do SYSTEM.CLUSTER.TRANSMIT.QUEUE na QMG1, protože SYSTEM.QSG.TRANSMIT.QUEUE se nachází ve struktuře CFLEVEL (3); proto podporuje pouze zprávy až do velikosti 63 kB.
  - Přeneseno do fronty klastru CLUSQ1 na QMG2 pomocí kanálu klastru TO.QMG2
- Všechny malé zprávy odeslané žádající aplikací jsou
  - Dejte do sdílené přenosové fronty SYSTEM.QSG.TRANSMIT.QUEUE. Tato fronta se nachází ve struktuře nakonfigurované s atributem RECOVER (YES), takže se používá pro trvalé i dočasné malé zprávy.
  - Načteno agentem IGQ na QMG2
  - Umístit do fronty klastru CLUSQ1 na QMG2

Pokud je vybraný cílový správce front QMG4:

- Vzhledem k tomu, že QMG4 není členem skupiny sdílení front SQ26, všechny zprávy odeslané žádající aplikací jsou
  - Vložte do SYSTEM.CLUSTER.TRANSMIT.QUEUE na QMG1
  - Přeneseno do fronty klastru CLUSQ1 na systému QMG4 pomocí kanálu klastru TO.QMG4 .

## Body k poznámce

- Žádající aplikace si nemusí být vědoma základního mechanismu použitého pro doručování zpráv.
- Potenciálně rychlejší mechanismus doručení je dosažen pro přenos malých přechodných zpráv mezi správci front ve skupině sdílení front (i v případě, že se v klastru nacházejí tytéž správce front).
- Pro doručení zprávy je k dispozici více cest (to znamená jak trasu klastru, tak i trasu ve frontě v rámci skupiny).
- Přenosová cesta k frontě v rámci skupiny je pravděpodobně rychlejší a je vybrána jako předvolba pro přenosovou cestu 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 dodání je možné bez ohledu na volbu MQOO\_BIND\_ \* uvedenou v aplikaci. Řazení do front v rámci skupiny distribuuje zprávy stejným způsobem jako klastrování, v závislosti na tom, zda je funkce MQOO\_BIND\_NOT\_FIXED, MQOO\_BIND\_ON\_OPEN, MQOO\_BIND\_ON\_GROUP nebo MQOO\_BIND\_AS\_Q\_DEF uvedena v otevřeném stavu.
- Když byla zvolena trasa a zprávy byly umístěny do přenosových front, použije se pro doručení zprávy pouze zvolená přenosová cesta. Všechny nezpracované zprávy na SYSTEM.QSG.TRANSMIT.QUEUE není přesměrována do systému SYSTEM.CLUSTER.TRANSMIT.QUEUE.

## z/OŠ **Klastrování, řazení do front v rámci skupiny a distribuované fronty**

Je možné nakonfigurovat správce front, který je členem klastru, stejně jako skupinu sdílení front a je připojen k distribuovanému správci front s použitím dvojice kanálu odesílatele/příjemce.

Tato konfigurace je kombinací distribuovaných front s frontami v rámci skupiny a klastrování s frontami v rámci skupiny.

Řazení do front v rámci skupiny je popsáno v tématu [“distribuované řazení do front s frontou v rámci skupiny \(více doručovacích cest\)”](#) na stránce 208.

Klastrování s frontami v rámci skupiny je popsáno v tématu [“Klastrování s frontami v rámci skupiny \(více doručovacích cest\)”](#) na stránce 210.

## Zprávy front v rámci skupiny

Tento oddíl popisuje zprávy, které jsou vloženy do SYSTEM.QSG.TRANSMIT.QUEUE.

### Struktura zprávy

Podobně jako všechny ostatní zprávy, které jsou vloženy do přenosových front, se jedná o zprávy, které jsou vloženy do SYSTEM.QSG.TRANSMIT.QUEUE má předponu v záhlaví přenosové fronty (MQXQH).

### Trvalost zpráv

V produktu IBM WebSphere MQ 5.3 a vyšším podporují sdílené fronty jak trvalé, tak dočasné zprávy.

Pokud je správce front ukončen v době, kdy agent IGQ zpracovává přechodné zprávy, nebo pokud se agent IGQ ukončí nestandardně během zpracování zpráv v polovině zpracování, může dojít ke ztrátě netrvalých zpráv, které se zpracovávají. Pokud je jejich obnova povinná, aplikace musí vytvořit opatření pro obnovu netrvalých zpráv.

Pokud se požadavek na vložení přechodné zprávy, vydaný agentem IGQ, neočekávaně nezdaří, bude nezpracovaná zpráva ztracena.

### Doručení zpráv

Agent IGQ načte a doručí všechny přechodné zprávy mimo rozsah synchronizačního bodu a všechny trvalé zprávy v rozsahu bodu synchronizace. V takovém případě se agent IGQ chová jako koordinátor synchronizačního bodu. Agent IGQ proto zpracovává přechodné zprávy stejně jako rychlé zprávy, přechodné zprávy jsou zpracovány na kanálu zpráv. Viz [Rychlé, přechodné zprávy](#).

### Dávkové zpracování 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 skládající se z trvalých zpráv, když nejsou k dispozici žádné další zprávy k načtení na SYSTEM.QSG.TRANSMIT.QUEUE.

### Velikost zpráv

Maximální velikost zprávy, kterou lze vložit do SYSTEM.QSG.TRANSMIT.QUEUE je maximální podporovaná délka zprávy pro sdílené fronty minus délka záhlaví přenosové fronty (MQXQH).

### Výchozí hodnota perzistence zpráv a výchozí priorita zpráv

Pokud je SYSTEM.QSG.TRANSMIT.QUEUE je v cestě rozpoznání názvu fronty zavedené v otevřeném čase, pak pro zprávy, které jsou vloženy s výchozí perzistencí a výchozí prioritou (nebo s výchozí perzistencí nebo výchozí prioritou), jsou použita běžná pravidla ve výběru fronty, která má výchozí prioritu a hodnoty perzistence, které se používají. (Další informace o pravidlech výběru fronty) naleznete v sekci [Zprávy produktu IBM MQ](#).

### Související pojmy

[“Nedoručené/nezpracované zprávy” na stránce 213](#)

Toto téma popisuje, co se stane s nedoručenou a nezpracovanou zprávou v systému SYSTEM.QSG.TRANSMIT.QUEUE.

[“Zprávy sestavy-řazení v rámci skupin do fronty” na stránce 214](#)

Toto téma popisuje zprávy sestavy: Potvrzení přijetí, potvrzení o doručení, sestava vypršení platnosti a hlášení výjimek.

## Nedoručené/nezpracované zprávy

Toto téma popisuje, co se stane s nedoručenou a nezpracovanou zprávou v systému SYSTEM.QSG.TRANSMIT.QUEUE.

Pokud nemůže agent IGQ doručit zprávu do cílové fronty, agent IGQ:

- Honoruje volbu sestavy MQRO\_DISCARD\_MSG (pokud pole voleb sestavy v produktu MQMD pro nedoručenou zprávu označuje, že musí) a zahodí nedoručenou zprávu.
- Pokusí se umístit nedoručenou zprávu do fronty nedoručených zpráv pro cílového správce front, pokud tato zpráva dosud nebyla zahozena. Agent IGQ používá předpony zprávy s hlavičkou fronty nedoručených zpráv (MQDLH).

Není-li fronta nedoručených zpráv definována nebo pokud nedoručenou zprávu nelze vložit do fronty nedoručených zpráv, a pokud nedoručená zpráva:

- persistent, agent IGQ zálohuje aktuální dávku trvalých zpráv, které zpracovává, a zadává do stavu opakování. Další informace viz téma [“Specifické vlastnosti front v rámci skupiny”](#) na stránce 215.
- Dočasný, agent IGQ vyřadí zprávu a pokračuje ve zpracování další zprávy.

Je-li správce front ve skupině sdílení front ukončen před tím, než má přidružený agent IGQ čas ke zpracování všech svých zpráv, nezpracovatelné zprávy zůstanou na SYSTEM.QSG.TRANSMIT.QUEUE, dokud nebude správce front spuštěn. Agent IGQ pak načte a doručí zprávy do cílových front.

Pokud prostředek Coupling Facility selže před všemi zprávami na systému SYSTEM.QSG.TRANSMIT.QUEUE byla zpracována, všechna nezpracovaná netrvalá zpráva se ztratí.

Produkt IBM doporučuje, aby aplikace neukládala zprávy přímo do přenosových front. Pokud aplikace vložila zprávy přímo do SYSTEM.QSG.TRANSMIT.QUEUE, agent IGQ nemusí být schopen tyto zprávy zpracovat a zůstávají na SYSTEM.QSG.TRANSMIT.QUEUE. Uživatelé pak musí použít své vlastní metody pro práci s těmito nezpracovanými zprávami.

## **Zprávy sestavy-řazení v rámci skupin do fronty**

Toto téma popisuje zprávy sestavy: Potvrzení přijetí, potvrzení o doručení, sestava vypršení platnosti a hlášení výjimek.

### **Potvrzení přijetí (COA) /potvrzení zpráv o doručení (COD)**

Zprávy COA a COD jsou generovány správcem front při použití front v rámci skupiny.

### **Zprávy sestavy vypršení platnosti**

Zprávy sestavy vypršení jsou generovány správcem front.

### **Výjimečná zpráva hlášení**

V závislosti na volbě sestavy MQR0\_EXCEPTION\_\* určené v poli *Volby sestav* v deskriptoru zpráv pro nedoručenou zprávu agent IGQ vygeneruje požadovanou sestavu výjimek a umístí ji do určené fronty pro odpovědi. Řazení do front v rámci skupiny lze použít k doručení zprávy o výjimce do cílové fronty pro odpověď.

Perzistence zprávy sestavy je stejná jako stálost nedoručené zprávy. Pokud se agentovi IGQ nepodaří přeložit název cílové fronty pro odpověď, nebo pokud selže vložení zprávy s odpovědí do přenosové fronty (pro následný přenos do cílové fronty odpovědi), pokusí se o vložení zprávy o výjimce do fronty nedoručených zpráv správce front, ve kterém je generována zpráva sestavy. Pokud to není možné, pak je-li nedoručená zpráva:

- persistent, agent IGQ vyřadí zprávu o výjimce, zálohuje aktuální dávku zpráv a zadá do stavu opakování. Další informace viz [“Specifické vlastnosti front v rámci skupiny”](#) na stránce 215.
- Dočasný, agent IGQ vyřadí hlášení o výjimce 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) mohou být nastaveny k řízení úrovně kontroly zabezpečení, která se provádí, když agent IGQ otevře cílové fronty.

### **Skupinové oprávnění v rámci skupiny (IGQAUT)**

Atribut IGQAUT může být nastaven tak, aby označoval typ kontrol zabezpečení, které mají být provedeny, a proto je třeba určit ID uživatelů, které má agent IGQ používat při vytváření autority pro vkládání zpráv do cílové fronty.

Atribut IGQAUT je analogický k atributu PUTAUT, který je k dispozici v definicích kanálů.

### **Identifikátor uživatele fronty v rámci skupiny (IGQUSER)**

Atribut IGQUSER lze použít k nominaci ID uživatele, které má být použito agentem IGQ, když se ustavuje oprávnění pro vkládání zpráv do cílové fronty.

Atribut IGQUSER je analogický k atributu MCAUSER, který je k dispozici na definicích kanálů.

Tento oddíl popisuje specifické vlastnosti front v rámci skupiny včetně zneplatnění obslužných rutin objektů, zotavení při vlastní obnově a opakovaného pokusu o práci s agentem fronty v rámci skupiny a také pro agenta a serializaci front v rámci skupiny.

**Zneplatnění popisovačů objektů (MQRC\_OBJECT\_CHANGED)**

Pokud se zjistí, že se atributy objektu změnilo po otevření objektu, správce front zneplatní popisovač objektu s atributem MQRC\_OBJECT\_CHANGED na dalším použití.

Zařazení do front v rámci skupiny zavádí následující pravidla pro zrušení platnosti popisovače objektu:

- Pokud je SYSTEM.QSG.TRANSMIT.QUEUE byla zahrnuta do cesty rozpoznání názvu během zpracování otevření, protože fronta v rámci skupiny byla v době otevření POVOLENA, ale řazení do fronty uvnitř skupiny bylo zjištěno, že je ve stavu vložení VYPNUTO, pak správce front zneplatní popisovač objektu a nezdaří se požadavek na vložení s MQRC\_OBJECT\_CHANGED.
- Pokud je SYSTEM.QSG.TRANSMIT.QUEUE nebyla zahrnuta do cesty rozpoznání názvu během zpracování otevření, protože vložení do fronty uvnitř skupiny bylo v otevřené době ZAKÁZÁNO, ale ve frontě v rámci skupiny bylo zjištěno, že je v čase vložení POVOLENO, pak správce front zneplatní popisovač objektu a nezdaří se požadavek na vložení s MQRC\_OBJECT\_CHANGED.
- Pokud je SYSTEM.QSG.TRANSMIT.QUEUE byla zahrnuta do cesty rozpoznání názvu během zpracování otevření, protože řazení do fronty v rámci skupiny bylo povoleno v otevřeném čase, ale SYSTEM.QSG.TRANSMIT.QUEUE byla změněna vložím času, poté správce front zneplatnil popisovač objektu a nezdařil se požadavek na vložení s MQRC\_OBJECT\_CHANGED.

**Vlastní obnova agenta intra-group queuing**

Pokud se agent IGQ nestandardně ukončí, zobrazí se zpráva CSQM067E a agent IGQ se spustí znovu.

**Zopakovat schopnost agenta pro řazení do front v rámci skupiny**

Pokud se agent IGQ setká s problémem při přístupu k systému SYSTEM.QSG.TRANSMIT.QUEUE (protože není definována, například nebo je definována s chybnými atributy, nebo je pro GET nebo z nějakého jiného důvodu blokována), agent IGQ přejde do stavu opakování.

Agent IGQ sleduje krátké a dlouhé počty opakování a intervaly. Hodnoty pro tyto počty a intervaly, které nelze změnit, jsou následující:

| <i>Tabulka 19. Počty krátkých a dlouhých opakování a intervaly opakování</i> |                      |
|------------------------------------------------------------------------------|----------------------|
| <b>Konstanta</b>                                                             | <b>Hodnota</b>       |
| 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 front v rámci skupiny a serializace**

Pokus agenta IGQ o serializaci přístupu ke sdíleným frontám v době, kdy stále probíhá peer zotavení, může selhat.

Dojde-li k selhání správce front ve skupině sdílení front, zatímco agent IGQ pracuje s nepotvrzenými zprávami ve sdílené frontě nebo ve frontách, ukončí se agent IGQ a pro správce front se selháním se provede obnova se sdílenou frontou. Vzhledem k tomu, že zotavení sdílené fronty typu peer je asynchronní činnost, ponechává před dokončením zotavení sdílené fronty front, aby došlo k restartování správce front, který selhal, a také u agenta IGQ pro tohoto správce front. Což zase ponechává možnost,

aby byly všechny potvrzené zprávy zpracovány před a mimo pořadí se zprávami, které jsou stále obnovovány. Aby se zajistilo, že zprávy nebudou zpracovány mimo pořadí, agent IGQ serializuje přístup ke sdíleným frontám vyvoláním volání rozhraní API MQCONN.

Pokus agenta IGQ o serializaci přístupu ke sdíleným frontám v době, kdy stále probíhá peer zotavení, může selhat. Je vydána chybová zpráva a agent IGQ je vložen do stavu opakování. Je-li zotavení typu peer správce front dokončeno, například při příštím pokusu, může být agent IGQ spuštěn.

## **z/OS** Správa úložišť na systému z/OS

Produkt IBM MQ for z/OS vyžaduje trvalé a dočasné datové struktury a používá k ukládání těchto dat sady stránek a vyrovnávací paměti. Tato témata poskytují více podrobností 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 216](#)

V tomto tématu se rozumí, jak produkt IBM MQ for z/OS používá sady stránek k ukládání zpráv.

[“Úložné třídy pro IBM MQ for z/OS” na stránce 217](#)

Paměťová třída je koncept produktu IBM MQ for z/OS, který umožňuje správci front mapovat fronty na sady stránek. Paměťové třídy můžete použít k řízení toho, které datové sady budou používat, které fronty.

[“Vyrovnávací paměti a fondy vyrovnávacích pamětí pro IBM MQ for z/OS” na stránce 218](#)

Produkt IBM MQ for z/OS používá vyrovnávací paměti a fondy vyrovnávacích pamětí k dočasnému ukládání dat do mezipaměti. Toto téma vám pomůže dále pochopit, jak jsou vyrovnávací paměti organizovány a používány.

### **Související odkazy**

[“Kde najít další informace o správě úložiště pro produkt IBM MQ for z/OS” na stránce 220](#)

Toto téma použijte jako referenci k vyhledání dalších informací o správě úložiště pro produkt IBM MQ for z/OS.

## **z/OS** Sady stránek pro IBM MQ for z/OS

V tomto tématu se rozumí, 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. Tyto údaje nejsou uloženy v sadách stránek správce front. Další informace o sdílených frontách viz [“Sdílené fronty a skupiny sdílení front” na stránce 158a](#) 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 od 00 do 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 souvisejících se správcem front. Pro normální provoz produktu IBM MQ je nezbytné, aby sada stránek nula nebyla plná, takže ji nepoužívejte k ukládání zpráv.

Chcete-li zvýšit výkon systému, měli byste také oddělit krátkodobou zprávu od dlouhotrvačných zpráv tím, že je umístíte na různé sady stránek.

Je třeba formátovat sady stránek a produkt IBM MQ pro něj poskytuje obslužný program FORMAT; viz téma [Formátování sad stránek \(FORMAT\)](#). Sady stránek musí být také definovány pro subsystém IBM MQ.

Produkt IBM MQ for z/OS lze nakonfigurovat tak, aby dynamicky rozšiřoval sadu stránek, pokud se stane úplným. IBM MQ pokračuje v rozbalování sady stránek, pokud je to požadováno, dokud 123 logických oblastí neexistuje, pokud je k dispozici dostatek paměťového prostoru. Jsou-li v tomto směru definovány lineární datové sady, oblasti pro rozšíření mohou přesahovat svazky, produkt IBM MQ však nemůže rozšiřovat sady stránek nad 64 GB.



Sady stránek nelze použít z jednoho správce front produktu IBM MQ v jiném správcí front IBM MQ nebo 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 správce front.

Není možné použít sady stránek větší než 4 GB ve správcí front se spuštěným vydáním starším než V6. Během období migrace, kdy je pravděpodobné, že se budete muset vrátit na předchozí vydání kódu:

- Neměňte sadu stránek 0, 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 umožňujících rozbalení nad rámec 4 GB naleznete v tématu [Definování sady stránek větší než 4 GB](#).

Je možné, aby administrátor dynamicky přidal sady stránek ke spuštěnému správcí front nebo odebral sady stránek ze spuštěného správce front (s výjimkou sady stránek nula). Příkaz DEFINE PSID lze spustit po dokončení restartování správce front, pouze v případě, že příkaz obsahuje klíčové slovo DSN.

## Úložné třídy pro IBM MQ for z/OS

Paměťová třída je koncept produktu IBM MQ for z/OS, který umožňuje správcí front mapovat fronty na sady stránek. Paměťové třídy můžete použít k řízení toho, které datové sady budou používat, které fronty.

### Představení paměťových tříd

*Paměťová třída* mapuje jednu nebo více front na sadu stránek. To znamená, že zprávy pro tuto frontu jsou uloženy v této sadě stránek.

Paměťové třídy vám umožňují řídit, kde jsou uložena nesdílená data zpráv pro administrativní účely, správu prostoru datové sady a správy zátěže nebo pro účely izolace aplikace. Paměťové třídy můžete také použít k definování skupiny XCF a názvu člena regionu produktu IMS, pokud používáte most IMS (popsané v části [“IBM MQ a IMS”](#) na stránce 273).

Sdílené fronty nepoužívají paměťové třídy k získání mapování sady stránek, protože zprávy na nich nejsou uloženy na sadách stránek.

### Jak fungují paměťové třídy

- Paměťová třídu definujete pomocí příkazu DEFINE STGCLASS, který určuje identifikátor sady stránek (PSID).
- Definujete-li frontu, zadejte do atributu STGCLASS danou paměťovou třídu.

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 na sadě stránek 21 (pokud zůstanou ve frontě dost dlouho na to, aby byly zapsány do DASD).

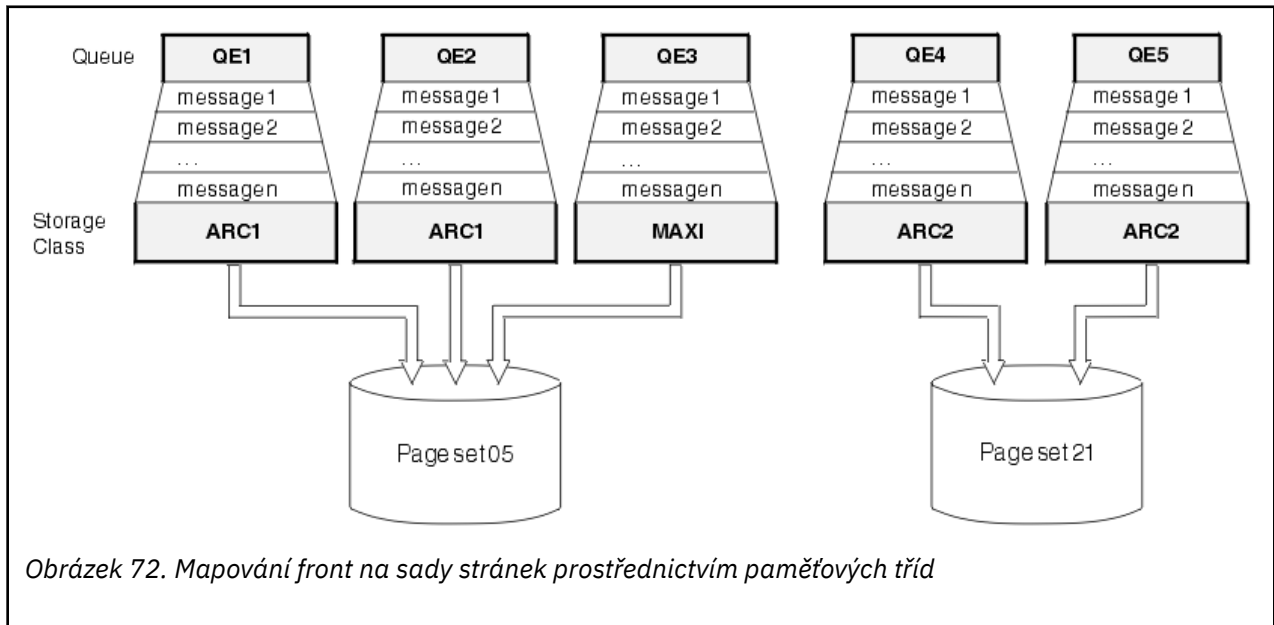
Více než jedna fronta může používat stejnou paměťovou třídu a vy můžete definovat tolik paměťových tříd, kolik chcete. Například můžete rozšířit předchozí příklad tak, aby zahrnoval více definic tříd a front úložiště, takto:

```

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 72](#) na stránce 218 jsou obě třídy úložiště ARC1 a MAXI přidruženy k sadě stránek 05. Proto jsou fronty QE1, QE2 a QE3 mapovány na sadu stránek 05. Podobně třída úložiště ARC2 přidružuje fronty QE4 a QE5 k sadě stránek 21.



Pokud definujete frontu, aniž byste určili třídu ukládání, produkt IBM MQ použije výchozí paměťovou třídu. Je-li zpráva vložena do fronty, která nepojmenovává neexistující paměťovou třídu, obdrží aplikace chybu. Musíte změnit definici fronty tak, aby jí bylo možné zadat název existující třídy úložiště, nebo vytvořit paměťovou třídu pojmenovanou touto frontou.

Paměťová třídu můžete změnit pouze v následujících případech:

- Všechny fronty, které používají tuto úložnou 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**

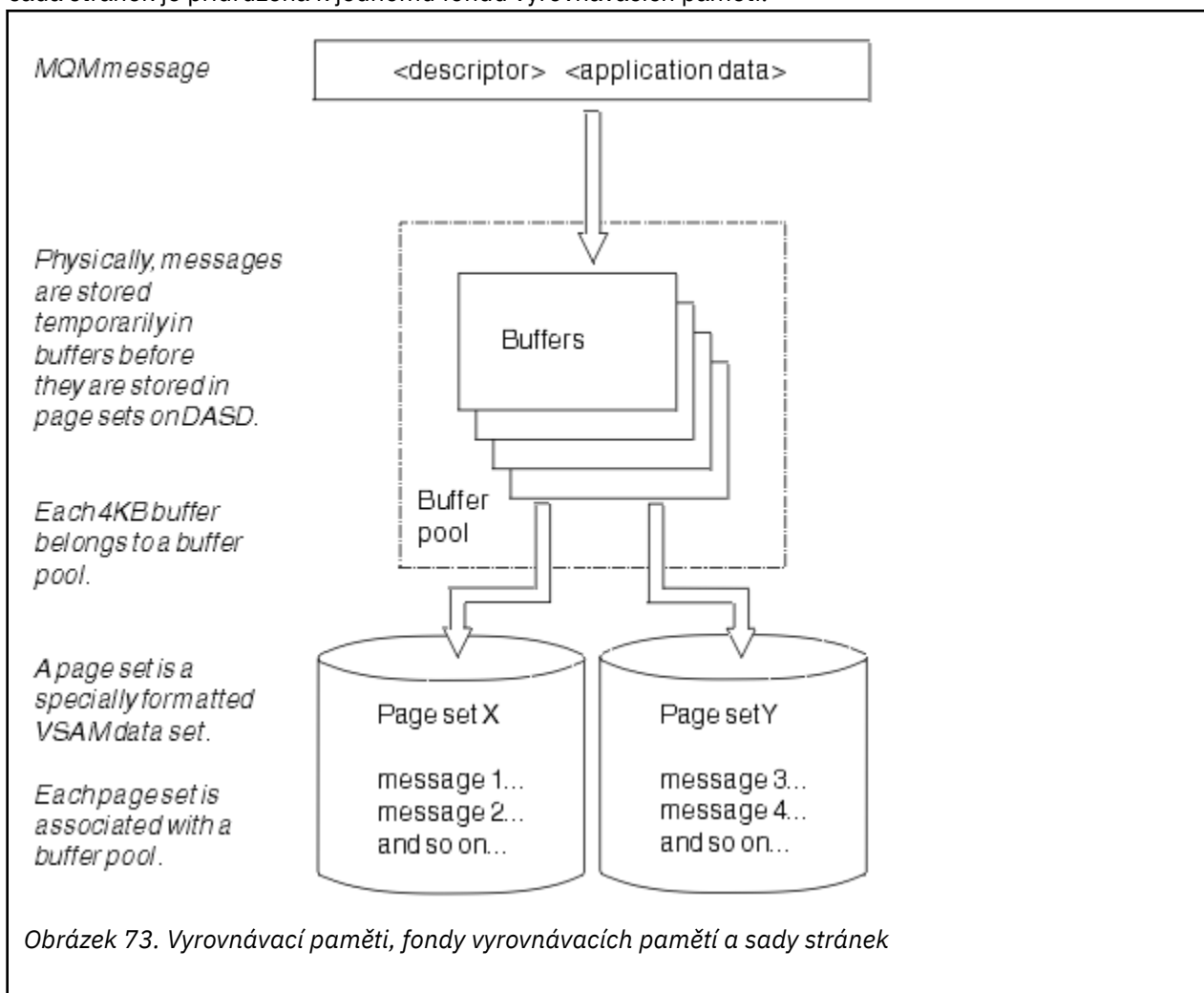
Produkt IBM MQ for z/OS používá vyrovnávací paměti a fondy vyrovnávacích pamětí k dočasnému ukládání dat do mezipaměti. Toto téma vám pomůže dále pochopit, jak jsou vyrovnávací paměti organizovány a používány.

Kvůli efektivitě používá produkt IBM MQ ukládání zpráv (a definic objektů) do mezipaměti před uložením do sad stránek v 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, což je komponenta produktu IBM MQ.

Vyrovnávací paměti jsou uspořádány do *fondů vyrovnávacích pamětí*. Pro každého správce front můžete definovat až 100 vyrovnávacích pamětí (0 až 99).

Doporučuje se použít minimální počet fondů vyrovnávacích pamětí konzistentní s objektem a oddělením typů zpráv popsanými v produktu Obrázek 73 na stránce 219a všechny požadavky na izolaci dat, které vaše aplikace může mít. Každá vyrovnávací paměť má délku 4 kB. Fondy vyrovnávacích pamětí používají standardně 31bitové úložiště, v tomto režimu je maximální počet vyrovnávacích pamětí určen množstvím 31bitového úložiště dostupného v adresním prostoru správce front; nevyužívá více než přibližně 70% pro vyrovnávací paměti. Případně lze přidělení fondu vyrovnávacích pamětí provést z 64bitového úložiště (použijte atribut LOCATION příkazu **DEFINE BUFFPOOL**). Použití UMÍSTĚNÍ (VÝŠE), aby bylo použito 64bitové úložiště, má dva výhody. Za prvé je k dispozici mnohem více 64 bitového úložiště, takže fondy vyrovnávací paměti mohou být mnohem větší, a za druhé, 31 bitové úložiště je zpřístupněno pro použití jinými funkcemi. Čím více vyrovnávacích pamětí máte, tím efektivnější je ukládání do vyrovnávací paměti a lepší výkon produktu IBM MQ.

Obrázek 73 na stránce 219 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 k úpravě velikosti fondu vyrovnávacích pamětí a umístění. Sady stránek lze dynamicky přidat pomocí příkazu **DEFINE PSID** nebo odstranit pomocí příkazu **DELETE PSID**.

Je-li fond vyrovnávacích pamětí příliš malý, IBM MQ vydá zprávu **CSQP020E**. Poté můžete dynamicky přidat více vyrovnávacích pamětí do postiženého fondu vyrovnávacích pamětí (povšimněte si, že je možné, že budete muset odebrat vyrovnávací paměti z jiných fondů vyrovnávacích pamětí, abyste mohli provést tuto operaci).

Počet vyrovnávacích pamětí ve fondu lze určit pomocí příkazu **DEFINE BUFFPOOL** a lze dynamicky měnit velikost fondů vyrovnávacích pamětí pomocí příkazu **ALTER BUFFPOOL**. Aktuální počet vyrovnávacích

paměti ve fondu je dynamicky určujete zobrazením sady stránek, která používá fond vyrovnávacích pamětí, pomocí příkazu **DISPLAY USAGE**.

Z výkonnostních důvodů neukládejte zprávy a definice objektů do stejného fondu vyrovnávacích pamětí. Použijte jeden fond vyrovnávacích pamětí (uvedte číslo 0) výhradně pro sadu stránek nula, kde jsou uchovány definice objektů. Podobně zachovejte krátkodobou zprávu a dlouze trvající zprávy v různých fondech vyrovnávacích pamětí, a proto i na 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.

## Kde najít další informace o správě úložiště pro produkt IBM MQ for z/OS

Toto téma použijte jako referenci k vyhledání dalších informací o správě úložiště pro produkt IBM MQ for z/OS.

Další informace o tématech v této sekci najdete v následujících zdrojích:

| Téma                                                          | Kde hledat                                                             |
|---------------------------------------------------------------|------------------------------------------------------------------------|
| Kolik paměťového prostoru potřebujete                         | <a href="#">Plánování požadavků na úložiště a výkon v systému z/OS</a> |
| Jak velké, aby vaše sady stránek a fondy vyrovnávacích pamětí | <a href="#">Plánování sad stránek a fondů vyrovnávacích pamětí</a>     |
| Správa sad stránek                                            | <a href="#">Správa sad stránek</a>                                     |
| Příkazy MQSC                                                  | <a href="#">Příkazy MQSC</a>                                           |

## přihlášení IBM MQ for z/OS

Produkt IBM MQ uchovává *protokoly* změn dat a významných událostí, jakmile k nim dojde. Tyto protokoly lze v případě potřeby použít k obnovení dat do předchozího stavu.

Datová sada *bootstrap data set* (BSDS) ukládá informace o datových sadách, které obsahují protokoly.

Protokol neobsahuje informace o statistice, trasování nebo vyhodnocení výkonu. Další podrobnosti o statistických a monitorovacích informacích, které produkt IBM MQ shromažďuje, naleznete v tématu [Monitorování a statistika](#).

Další informace o protokolování naleznete v následujících tématech:

- [“Soubory protokolu v produktu IBM MQ for z/OS” na stránce 221](#)
- [“Jak je protokol strukturován” na stránce 225](#)
- [“Jak jsou zapisovány protokoly produktu IBM MQ for z/OS” na stránce 225](#)
- [“Větší relativní adresa bajtů protokolu” na stránce 228](#)
- [“Datová sada zaváděcího programu” na stránce 229](#)


### Související úlohy

[Plánování vašeho protokolovacího prostředí](#)

[Nastavení protokolů pomocí modulu parametrů systému](#)

## Správa serveru z/OS

## Související odkazy

 [Zprávy pro IBM MQ for z/OS](#)

[Zdroje, ze kterých můžete zadávat příkazy MQSC v systému z/OS](#)

## Soubory protokolu v produktu IBM MQ for z/OS

Soubory protokolů obsahují informace potřebné pro zotavení transakcí. Aktivní soubory protokolů mohou být archivovány, takže můžete uchovávat data protokolu po dlouhou dobu.

### Co je soubor protokolu

Produkt IBM MQ zaznamenává všechny významné události tak, jak se vyskytují v *aktivním protokolu*. Protokol obsahuje informace potřebné k obnově:

- Trvalé zprávy
- Objekty produktu IBM MQ , například fronty.
- Správce front produktu IBM MQ

Aktivní protokol se skládá z kolekce datových sad (až 310), které se používají cyklicky.

Můžete povolit archivaci protokolu, takže když aktivní protokol zaplní kopii, je vytvořena v archivní datové sadě. Použití archivace umožňuje uchovávat data protokolu po delší dobu. Pokud nepoužíváte archivaci, budou přepsány protokoly se zpětnou smyčkou a dřívější data. Chcete-li obnovit sadu stránek nebo obnovit data ve struktuře prostředku CF, budete potřebovat data protokolu při provádění zálohy sady stránek nebo struktury. Archivní protokol může být vytvořen na disku nebo na pásce.

### Archivace

Protože aktivní protokol má pevnou velikost, IBM MQ zkopíruje obsah každé datové sady protokolu pravidelně do *protokolu archivace*, což je obvykle datová sada na paměťovém zařízení s přímým přístupem (DASD) nebo na magnetické pásce. Pokud došlo k selhání subsystému nebo transakce, produkt IBM MQ použije aktivní protokol a v případě potřeby protokol archivace pro zotavení.

Archivní žurnál může obsahovat až 1000 sekvenčních datových sad. You can catalog each data set using the z/OS integrated catalog facility (ICF).

Archivace je nezbytnou komponentou zotavení produktu IBM MQ . Je-li jednotka zotavení s dlouhou dobou zpracování, záznamy žurnálu v rámci této jednotky zotavení se mohou nacházet v protokolu archivace. V tomto případě obnova vyžaduje data z protokolu archivace. Pokud je však archivace zakázána, aktivní protokol s novými záznamy protokolu zabalí a přepíše předchozí 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 nepřepínejte archivaci**. Pokud tak učiníte, riskujete ztrátu dat po selhání systému nebo transakce. Pouze v případě, že pracujete v testovacím prostředí, můžete uvažovat o vypnutí archivace. Pokud to potřebujete provést, použijte makro CSQ6LOGP , které je popsáno v tématu [Použití CSQ6LOGP](#).

Chcete-li zabránit problémům s neplánovanými dlouhými jednotkami práce, IBM MQ vydá zprávu ([CSQJ160I](#) nebo [CSQJ161I](#)) je-li během aktivního zpracování odložovacího protokolu zjištěna dlouhodobě spuštěná jednotka práce.

### duální protokolování

V duálním protokolování je každý záznam protokolu zapsán do dvou různých aktivních datových sad protokolů, aby se minimalizovala pravděpodobnost problémů se ztrátou dat během restartu.

Produkt IBM MQ můžete nakonfigurovat tak, aby se spouštěl buď s *jediným protokolováním*, nebo *duálním protokolováním*. Při jednom protokolování se záznamy protokolu zapisují jednou do aktivní datové sady protokolu. Každá aktivní datová sada protokolu je jednoduchou datovou sadou VSAM s jedinou fyzickou oblastí (LDS). Při duálním protokolování je každý záznam protokolu zapsán do dvou různých aktivních datových sad protokolů. Duální protokolování minimalizuje pravděpodobnost vzniku problémů se ztrátou dat během restartu.

## posunování protokolu

Protokol se posunuje, aby záznamy protokolu pro některé jednotky práce byly zapsány dále v protokolu. Tím se sníží množství dat protokolu, která musí být přečtena při opětovném spuštění správce front nebo odvolání, dlouhodobě spuštěných nebo dlouhodobých nejistých jednotek práce.

Je-li jednotka práce považována za příliš dlouhou, je zápis každého záznamu protokolu zapsán dále v protokolu. Tato technika je známá jako *posunování*. Po zpracování celé jednotky práce se jednotka práce nachází ve stavu *posunut*. Jakákoli aktivita vrácení nebo restartování související s odsunovanou pracovní jednotkou může používat posunované záznamy protokolu místo použití původní jednotky záznamů pracovního protokolu.

Deinstalace dlouhotrvajícího pracovní jednotky je funkce procesu kontrolního bodu. V čase kontrolního bodu je každá aktivní jednotka práce zkontrolována, aby se zjistilo, zda je třeba jej vyšídit. Pokud byla jednotka práce přes dva předchozí kontrolní body od doby, kdy byla vytvořena, nebo od doby, kdy byla naposledy chovaná, je jednotka práce vhodná k posunování. To znamená, že jedna jednotka práce může být posunuta více než jednou. To je známé jako *vícenásobně posunovaná jednotka práce*.

Pracovní jednotka se posunuje každé tři kontrolní body. Kontrolní bod se však provádí asynchronně k přepínači protokolu (nebo při zápisu záznamu protokolu, který způsobil překročení parametru LOGLOAD).

Existuje pouze jeden kontrolní bod, který se provádí v daném okamžiku, takže může být více protokolů-přepínačů před dokončením kontrolního bodu.

To znamená, že pokud není k dispozici dostatek aktivních protokolů, nebo pokud jsou příliš malé, před dokončením všech protokolů nemusí být dokončeno posunování velké jednotky práce.

Zpráva [CSQR027I](#) má za následek, že posunování není možné dokončit.

Je-li archivace protokolu vypnuta, vyskytne se ABEND 5C6 s příčinou 00D1032A, pokud dojde k pokusu o vrácení pracovní jednotky, pro kterou došlo k selhání posunování. Abyste se vyhnuli tomuto problému, měli byste použít OFFLOAD=YES.

Protokolování posunování je vždy aktivní a spustí se, zda je povolena archivace protokolu, či nikoli.

**Poznámka:** Ačkoli jsou všechny záznamy protokolu pro pracovní jednotku odchleny, celý obsah každého záznamu není posunut, ale pouze ta část, která je nezbytná pro odvolání. To znamená, že množství zapsaných dat protokolu je udržováno na minimu, a že záznamy s posunem nelze použít, pokud dojde k selhání sady stránek. Dlouhá běžící jednotka práce je taková, která byla spuštěna pro více než tři kontrolní body správce front.

Další informace o posunování protokolu naleznete v tématu [Správa protokolů](#).

## Kompresce 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.

Kompresce protokolu může být použita ke snížení množství dat zapsaných do protokolu pro trvalé zprávy v soukromých frontách. Množství komprese, které je dosaženo, závisí na typu dat obsažených ve zprávách. Například RLE (Run Length Encoding) pracuje tak, že zhušťuje opakované instance bajtů, které mohou efektivně poskytovat dobré výsledky pro strukturovaná data nebo data orientovaná na záznam.



**Upozornění:** Trvalé zprávy, které jsou vloženy do sdílené fronty, nejsou předmětem komprese protokolu.

Pole v sekci správce protokolů v záznamech SMF (System Management Facility 115) můžete použít k monitorování toho, jak velká komprese dat je dosažena. Další informace o prostředí SMF najdete v tématu [Použití nástroje pro správu systému](#) a [Zprávy o evidenci a statistice](#).

Komprese protokolu zvyšuje využití procesoru systému. Měli byste zvážit použití komprese pouze v případě, že propustnost správce front je omezena šířkou pásma I/O zapisujícími do datových sad protokolů, nebo pokud omezíte diskovou paměť potřebnou k uchování datových sad protokolu. Pokud používáte sdílené fronty, pak lze omezení šířky pásma I/O zmírnit přidáním dalších správců front do skupiny sdílení front a rozdělením pracovní zátěže do dalších správců front.

Volbu komprese protokolu lze povolit a zakázat, jak je požadováno, aniž by bylo třeba správce front zastavit a restartovat. Správce front může číst libovolné komprimované záznamy žurnálu bez ohledu na aktuální nastavení komprese protokolu.

Správce front podporuje 3 nastavení komprese protokolu.

### **ŽÁDNÉ**

Nepoužije se žádná komprese dat protokolu. Toto je výchozí hodnota.

### **RLE**

Komprese dat protokolu se provádí za použití kódování RLE (run-length encoding).

### **ANY**

Povolte správci front vybrat algoritmus komprese, který poskytuje největší stupeň komprese záznamu protokolu. Tato volba má za následek kompresi RLE.

Komprimace záznamů protokolu můžete ovlivnit pomocí jedné z následujících možností:

- Příkazy SET a DISPLAY LOG v MQSC; viz [SET LOG](#) a [DISPLAY LOG](#)
- Funkce Nastavit protokol a Zjistit protokol v rozhraní PCF; viz [Nastavit protokol](#) a [Dotaz na protokol](#)
- Makro CSQ6LOGP v modulu parametrů systému, viz [Použití CSQ6LOGP](#)

Kromě toho má obslužný program Log Print CSQ1LOGP podporu pro rozbalení všech komprimovaných záznamů protokolu.

## **Data protokolu**

Protokol může obsahovat až 18 milionů milionů milionů ( $1.8 \cdot 10^{19}$ ) bajtů. Každý bajt může být adresován jeho ofsetem od začátku protokolu a tento posun je znám jako jeho *adresa RBA (Relative Byte Address)*.

RBA se odkazuje na 6bajtové nebo osmibajtové pole poskytující úplný adresovatelný rozsah  $2^{48}$  bajtů, nebo  $2^{64}$  bajtů, v závislosti na tom, zda se používají 6bajtová nebo 8bajtová paměť RBA.

Když však produkt IBM MQ zjistí, že používaný rozsah je nad rámec F00000000000 (jsou-li používány 6bajtové RBA) nebo FFFF800000000000 (jsou-li používány 8bajtové protokoly RBS), jsou vydány zprávy [CSQI045](#), [CSQI046](#), [CSQI047a](#) [CSQJ032](#), což vás varuje, abyste resetovali protokol RBA protokolu.

Pokud hodnota RBA dosáhne hodnoty FFF800000000 (jsou-li používány 6bajtová RBA) nebo FFFFFFFC0000000000 (pokud se používá 8bajtový protokol RBA), bude správce front ukončen s kódem příčiny 00D10257.

Jakmile se zobrazí varovné zprávy o použitém rozsahu protokolu, měli byste naplánovat výpadek správce front, během kterého může být správce front převeden tak, aby používal osmibajtové protokoly RBA, nebo je možné obnovit protokol. Postup resetování protokolu je dokumentován v části [Reset protokolu správce front](#).

Pokud správce front používá 6bajtová RBA, zvažte možnost převodu správce front tak, aby používal 8bajtový protokol RBA, než resetuje protokol správce front, podle postupu popsání v tématu [Implementace větší relativních bajtových adres protokolu](#).

Protokol se skládá z *záznamů protokolu*, z nichž každá je sada dat protokolu, se kterými se zachází jako s jednou jednotkou. Záznam protokolu je identifikován buď pomocí RBA prvního bajtu jeho záhlaví, nebo podle jeho pořadového čísla záznamu protokolu (LRSN). Adresa RBA nebo LRSN jedinečně identifikuje záznam, který začíná v určitém bodě protokolu.

Informace o tom, zda používáte skupiny sdílení front, závisí na tom, zda používáte RBA nebo LRSN k identifikaci bodů protokolu. V prostředí sdílení front nelze použít relativní adresu úložiště k jedinečné identifikaci bodu protokolu, protože více správců front může aktualizovat stejnou frontu ve stejnou dobu a každý má vlastní protokol. Chcete-li tento problém vyřešit, je pořadové číslo záznamu protokolu odvozeno z hodnoty časového razítka a nemusí nutně představovat fyzický posun záznamu protokolu v rámci protokolu.

Každý záznam protokolu má záhlaví, které uvádí jeho typ, dílčí komponentu IBM MQ, která provedla záznam, a pro záznamy jednotky zotavení identifikátor jednotky zotavení.

Existují čtyři typy záznamů protokolu, které jsou popsány pod následujícími nadpisy:

- Jednotka záznamů protokolu o zotavení
- Záznamy kontrolního bodu
- Řídící záznamy sady stránek
- Záznamy o zálohování struktury prostředku CF

## Jednotka záznamů protokolu o zotavení

Většina záznamů žurnálu popisuje změny ve frontách produktu IBM MQ. Všechny tyto změny se provádějí v jednotkách zotavení.

Produkt IBM MQ používá speciální metody protokolování zahrnující příkaz *Vrátit zpět/Znovu* a *kompensační záznamy protokolu*, aby se zkrátila doba restartování a zlepšila dostupnost systému.

Jedním z důsledků této skutečnosti je, že doba restartu je omezena. Dojde-li k selhání během restartu, aby bylo možné správce front znovu spustit podruhé, nemusí být všechny aktivity nápravy, které byly dokončeny do bodu selhání v prvním restartu, znovu aplikovány během druhého restartu. To znamená, že po sobě jdoucí restarty nebudou trvat delší dobu, než se dokončí.

## Záznamy kontrolního bodu

Chcete-li zkrátit dobu restartu, produkt IBM MQ provádí pravidelné kontrolní body během normálního provozu. Jedná se o následující:

- Když byl zapsán předdefinovaný počet záznamů protokolu. Toto číslo je definováno operandem frekvence kontrolního bodu s názvem LOGLOAD makra parametru systému CSQ6SYSP, které je popsáno v tématu Použití CSQ6SYSP.
- Na konci úspěšného restartu.
- Při normálním ukončení.
- Při každém přepnutí IBM MQ do další aktivní datové sady protokolu v cyklu.

V době, kdy je proveden kontrolní bod, produkt IBM MQ vydá příkaz DISPLAY CONN (popsaný v části DISPLAY CONN). interně tak, aby byl do protokolu konzoly serveru z/OS zapsán seznam připojení, která jsou momentálně nejistá.

## Řídící záznamy sady stránek

Tyto záznamy zaregistrují sady stránek a fondy vyrovnávacích pamětí známé správcí front IBM MQ na každém kontrolním bodu a zaznamenávají informace o rozsazích protokolu, které jsou nezbytné k provedení obnovy média dané sady stránek v době kontrolního bodu.

Určité dynamické změny sad stránek a fondů vyrovnávacích pamětí jsou také zapsány jako řídicí záznamy sady stránek, aby mohly být změny obnoveny a automaticky obnoveny při příštím spuštění správce front.



## Záznamy zálohování struktury prostředku CF

Tyto záznamy obsahují data přečtená ze struktury seznamu prostředku Coupling Facility v odpovědi na příkaz BACKUP CFSTRUCT. V nepravděpodobném případě selhání struktury prostředku Coupling Facility jsou tyto záznamy použity společně s jednotkou záznamů o zotavení pomocí příkazu RECOVER CFSTRUCT k provedení zotavení mediálního zařízení struktury prostředku Coupling Facility do bodu selhání.

### Související úlohy

[Implementace větší relativní bajtové adresy protokolu](#)

## Jak je protokol strukturován

Toto téma použijte k pochopení terminologie používané k popisu záznamů protokolu.

Každá aktivní datová sada protokolu musí být lineární datovou sadou VSAM (LDS). Fyzická výstupní jednotka, zapsaná do aktivní datové sady protokolu, je 4 kB řídicí interval (CI). Každá KP obsahuje jeden záznam VSAM.

### Fyzické a logické záznamy protokolu

Jedna KP VSAM je *fyzický* záznam. Informace zaprotokolované v konkrétním čase tvoří záznam *logical* s délkou, která se liší nezávisle na prostoru, který je k dispozici 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 zapotřebí k uložení.

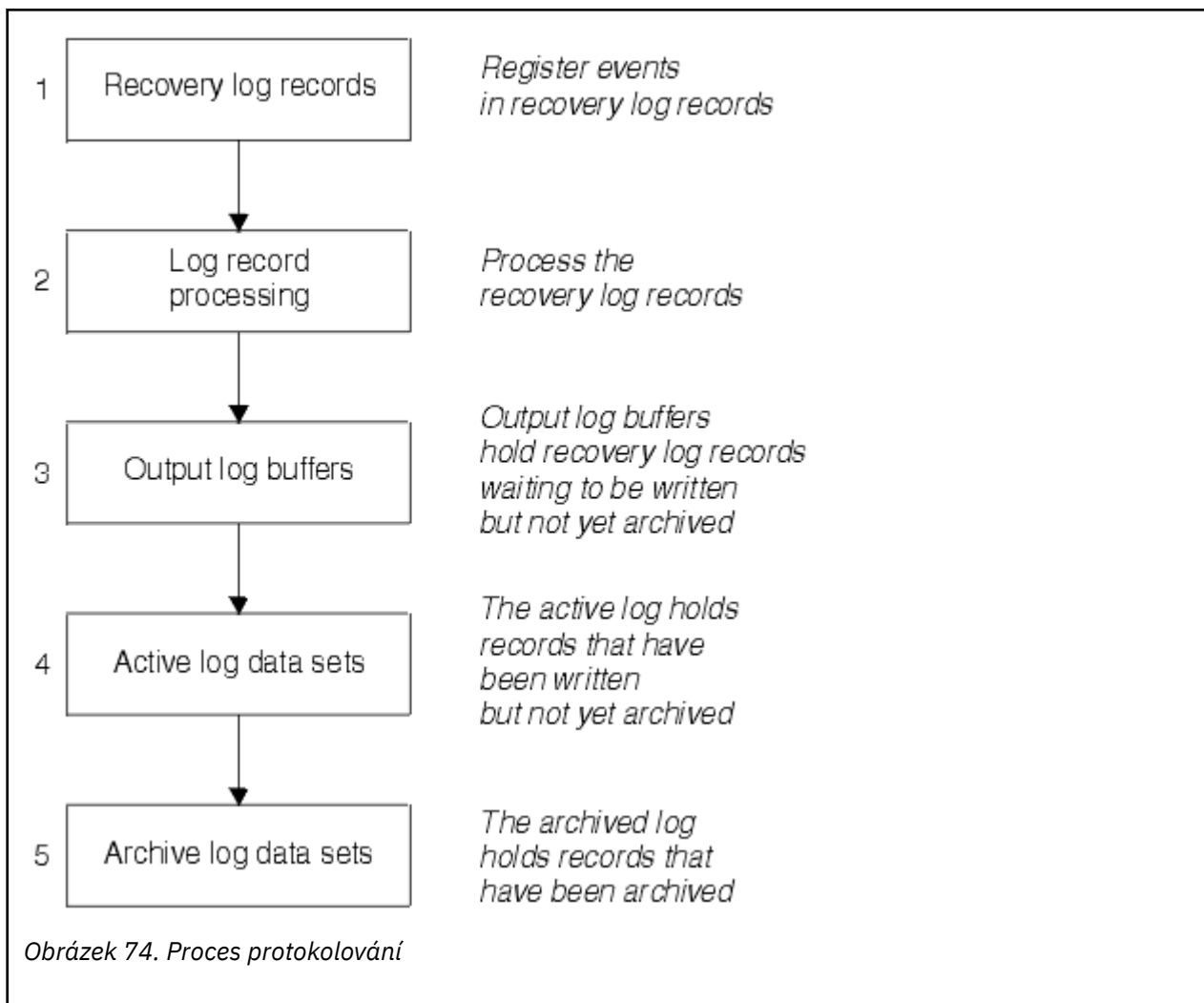
## Jak jsou zapisovány protokoly produktu IBM MQ for z/OS

Toto téma vám pomůže pochopit, jak produkt IBM MQ zpracovává záznamy souboru protokolu.

Produkt IBM MQ zapisuje každý záznam protokolu do datové sady DASD nazvané *aktivní protokol*. Je-li aktivní protokol plný, produkt IBM MQ zkopíruje svůj obsah do datové sady DASD nebo z datové sady, která se nazývá *archivní protokol*. Tento proces se nazývá *odlehčování*.

Obrázek 74 na stránce 226 ilustruje proces protokolování. Záznamy protokolu obvykle procházejí následujícím cyklem:

1. IBM MQ zaznamenává změny dat a významné události v záznamech protokolu o zotavení.
2. IBM MQ zpracuje záznamy protokolu o zotavení a rozdělí je do segmentů, je-li to nutné.
3. Záznamy protokolu jsou postupně umísťovány do *vyrovnávací paměti výstupního protokolu*, které jsou formátovány jako Interval ovládacích prvků VSAM (CI). Každý záznam protokolu je identifikován pomocí relativní bajtové adresy v rozsahu nula až  $2^{64} - 1$ .
4. KP jsou zapsány do sady předdefinovaných datových sad aktivního protokolu DASD, které se používají sekvenčně a recyklovány.
5. Je-li archivace aktivní, jakmile se každá aktivní datová sada protokolu zaplní, bude její obsah automaticky přenesen do nové datové sady protokolu archivace.



## Když je zapsán aktivní protokol

Vyrovňovací paměti protokolu v úložišti jsou zapsány do aktivní datové sady protokolu, kdykoli se vyskytne některá z následujících podmínek:

- Vyrovňovací paměti protokolu budou plné.
- Byla dosažena prahová hodnota zápisu (jak je uvedeno v makru CSQ6LOGP).
- Objevují se určité významné události, jako například bod potvrzení, nebo když je vydán příkaz IBM MQ BACKUP CFSTRUCT.

Při inicializaci správce front jsou datové sady aktivního žurnálu pojmenované v BSDS dynamicky přidělovány pro výhradní použití správcem front a zůstanou alokovány výhradně pro produkt IBM MQ, dokud není správce front ukončen.

## Dynamické přidání datových sad protokolu

Při spuštění správce front je možné dynamicky definovat nové datové sady aktivního protokolu. Tato funkce zmírňuje problém uváznutí správce front v případě, že archivace není schopna odlehčit aktivní protokoly kvůli přechodnému problému. Další informace naleznete v popisu příkazu [DEFINE LOG](#).

**Poznámka:** Chcete-li znovu definovat nebo odebrat aktivní protokoly, je třeba ukončit a restartovat správce front.

## Subsystém IBM MQ a subsystém správy úložišť

Parametry IBM MQ vám umožňují uvést třídy úložiště MVS/DFP SMS (Storage Management Subsystem) při dynamickém přidělování datových sad archivního protokolu IBM MQ. Produkt IBM MQ zahájí archivaci datových sad protokolu, ale pomocí serveru SMS lze provést přidělení datové sady archivu.

### Související odkazy

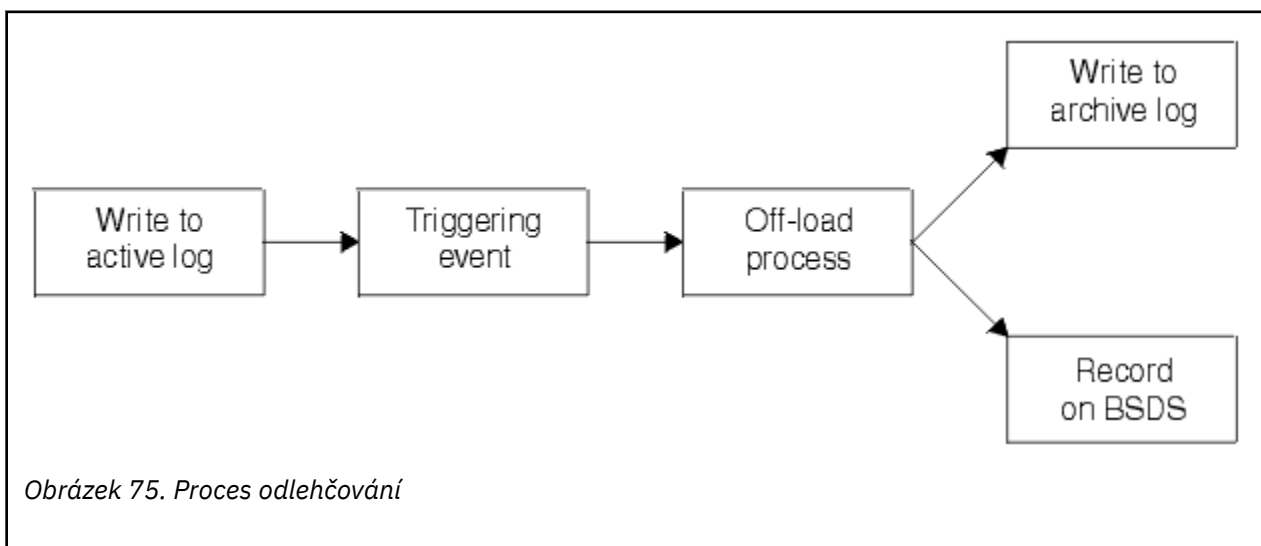
[“Je-li zapsán protokol archivace IBM MQ for z/OS” na stránce 227](#)

Toto téma slouží k pochopení procesu kopírování aktivních protokolů do protokolů archivace a při výskytu procesu.

### **z/OS** Je-li zapsán protokol archivace IBM MQ for z/OS

Toto téma slouží k pochopení procesu kopírování aktivních protokolů do protokolů archivace a při výskytu procesu.

Proces kopírování aktivních protokolů do archivačních protokolů se nazývá *odlehčování*. Vztah odsunutí na další události protokolování je schematicky zobrazen v produktu [Obrázek 75 na stránce 227](#).



### Spouštění procesu odlehčování

Proces odsunutí aktivního protokolu do protokolu archivace může být spuštěn několika událostmi. Příklad:

- Naplnění datové sady aktivního protokolu.
- Použijte příkaz MQSC ARCHIVE LOG.
- Došlo k chybě při zápisu do aktivní datové sady protokolu.

Datová sada se osekne před bodem selhání a záznam, který nebyl zapsán, se stane prvním záznamem nové datové sady. Přenášení se spustí pro oříznuté datové sady jako pro běžnou plnou datovou sadu žurnálu. Pokud existují duální aktivní protokoly, obě kopie jsou oříznuty tak, aby zůstaly dva záložní kopie synchronizované.

Zpráva CSQJ110E je vydána, když je poslední dostupný aktivní protokol o 5% plný a po 5% přírůstcích poté, kde se uvádí procentní část kapacity protokolu, která se používá. Dojde-li k zaplnění všech aktivních žurnálů, zastaví produkt IBM MQ zpracování, dokud nedojde k odlehčení, a zobrazí se tato zpráva:

```
CSQJ111A +CSQ1 OUT OF SPACE IN ACTIVE LOG DATA SETS
```

## Proces odlehčování

Jakmile se zaplní všechny aktivní protokoly, produkt IBM MQ spustí proces odlehčování a zastaví zpracování, dokud nebude dokončen proces odkládání. Pokud při zaplnění aktivních protokolů dojde k selhání zpracování odkládání, příkaz IBM MQ se ukončí.

Je-li aktivní protokol připraven k odlehčení zátěže, pošle se operátor konzole z/OS požadavek na připojení pásky nebo k přípravě jednotky DASD. Hodnota volby protokolování ARCWTOR (pro další informace viz [Použití CSQ6ARVP](#)) určuje, zda je požadavek přijat. Používáte-li pásku k odlehčování, zadejte parametr `ARCWTOR=YES`. Je-li hodnota `YES`, je před požadavkem uvedena `WTOR` (číslo zprávy `CSQJ008E`) sdělíte operátorovi, aby připravil datovou sadu protokolu archivu, která má být přidělena.

Operátor nemusí okamžitě odpovídat na tuto zprávu. Zpoždění odezvy však zpožďuje proces odkládání. Nemá vliv na výkon systému IBM MQ, pokud operátor zpožďuje odpověď tak dlouho, dokud produkt IBM MQ nevypracuje z aktivních protokolů.

Operátor může odpovědět zrušením procesu odkládání. V tomto případě, pokud je přidělení pro první kopii souborů s duálním archivem, je proces odlehčování pouze zpožděn, dokud nebude zaplněna další aktivní datová sada protokolu. Je-li alokace určena pro druhou kopii, proces archivace se přepne do režimu jedné kopie, ale pouze pro tuto datovou sadu.

## Přerušování a chyby při odlehčování

Požadavek na zastavení správce front se neprojevívá, dokud nebude zpracování odsunutí dokončeno. Pokud proces IBM MQ selže při provádění odlehčování, při restartu správce front se znovu spustí odlehčování.

## Zprávy během zpracování odkládání

Zanesené zprávy se odesílají na konzolu z/OS produktem IBM MQ a procesem odlehčování. Tyto zprávy můžete použít k vyhledání rozsahů RBA v různých datových sadách protokolů.

### **Větší relativní adresa bajtů protokolu**

Tato funkce zlepšuje dostupnost správce front tím, že zvyšuje časový interval, než budete muset resetovat protokol.


Data pro zotavení jsou zapsána do protokolu, aby byly při restartu správce front k dispozici trvalé zprávy. Termín log Relative Byte Address (log RBA) se používá jako odkaz na umístění dat jako ofset od začátku protokolu.


Před IBM MQ 8.0 se šestibajtová adresa RBA může adresovat až 256 terabajtům dat. Před zapsáním tohoto množství záznamů protokolu musíte znovu nastavit protokol správce front následujícím postupem dokumentovaným v části [Reset protokolu správce front](#).

Resetování protokolů správců front není rychlý proces a může vyžadovat rozšířený výpadek v důsledku nutnosti resetovat sady stránek jako součást procesu. U správce front s vysokým využitím může být tato operace obvykle prováděna jednou ročně.

V produktu IBM MQ 8.0 může být protokol RBA protokolu dlouhý 8 bajtů a správce front může nyní před protokolem RBA resetovat více než 64000 krát více dat (16 exabajtů). Dopad použití většího protokolu RBA je tak, že velikost dat protokolu se zvýší o několik bajtů.

## Kdy je tato funkce aktivována?

 Tuto funkci můžete kdykoli povolit, i když v ideálním případě byste měli naplánovat změnu a explicitně ji povolit v daném okamžiku, než se aktuální protokol RBA blíží ke konci rozsahu 6 bajtů protokolu RBA. Pokyny pro plánování migrace na 8 bajtů RBA najdete v tématu [Plánování zvýšení maximálního adresovatelného rozsahu protokolu](#).

 Správci front vytvořenému v produktu IBM MQ 9.2.5 nebo novější již mají tuto funkci zapnutou.

**Důležité:** Než budete moci tuto schopnost použít ve skupině sdílení front, všechny správce front ve skupině sdílení front se musí nacházet na jedné z následujících úrovní:

- V IBM MQ 9.0.n CD, IBM MQ 9.1.0 LTS nebo pozdější
- V IBM MQ 9.0.0 a byla spuštěna s **OPMODE= (NEWFUNC,800)** nebo **OPMODE= (NEWFUNC,900)**

Pokud se aktuální protokol RBA blíží ke konci rozsahu protokolu RBA, zvažte možnost převodu správce front tak, aby používal 8bajtový protokol RBA namísto resetování protokolu správce front. Převod správce front tak, aby používal 8 bajtů RBL, vyžaduje kratší výpadek než resetování protokolu a významně prodlužuje časový interval, než budete muset protokol resetovat.

Zpráva [CSQJ034I](#), vydaná během inicializace správce front, označuje konec rozsahu protokolu RBA pro správce front, jak je nakonfigurován, a lze jej použít k určení, zda se používá 6bajtová nebo 8bajtová adresa RBAs.

## Jak je tato funkce povolena?

Osmibajtová adresa RBA protokolu je povolena spuštěním správce front s formátem BSDS verze 2. V souhrnu, to je dosaženo tím, že:

1. Zajištění, aby všichni správci front ve skupině sdílení front splňovaly požadavky na povolení 8bajtového protokolu RBA
2. Vyčištění správce front čistě
3. Spuštění obslužného programu [BSDS conversion utility](#) pro vytvoření kopie BSDS ve formátu verze 2.
4. Restartování správce front s převedenými BSDS.

Jakmile byl správce front převeden na použití 8 bajtů RBL, nemůže se vrátit zpět na použití 6 bajtů protokolu RBA.

Podrobný postup, jak povolit osmibajtové protokoly RBA, najdete v tématu [Implementace větší relativní bajtové adresy žurnálu](#).

### Související úlohy

[Plánování zvýšení maximálního adresovatelného rozsahu protokolu](#)

### Související odkazy

[Obslužný program převodu BSDS \(CSQJUCNV\)](#)

## Datová sada zaváděcího programu

Konfigurační sada dat zaváděcího programu vyžaduje IBM MQ jako mechanismus pro odkazy na datové sady protokolu a záznamy protokolu. Tyto informace jsou povinné během normálního zpracování a restartování zotavení.

## Co je datová sada zaváděcího programu pro

Datová sada *bootstrap data set* (BSDS) je datová sada KSDS (VSAM Key-Sequenced data set), která uchovává informace potřebné pro IBM MQ. Obsahuje následující informace:

- Soupis všech aktivních a archivovaných datových sad protokolů známých produktu IBM MQ. Produkt IBM MQ používá tento soupis k:
  - Sledování aktivních a archivovaných datových sad protokolu
  - Vyhledejte záznamy protokolu tak, aby mohly uspokojit požadavky na čtení protokolu během normálního zpracování
  - Vyhledejte záznamy protokolu tak, aby bylo možné zpracovat zpracování restartu

IBM MQ ukládá informace v soupisu pokaždé, když je definována datová sada protokolu archivace, nebo se znovu použije aktivní datová sada protokolu. V případě aktivních protokolů se zobrazí soupis, který je plný a který je k dispozici pro opětovné použití. Inventář uchovává relativní adresu bytu (RBA) každé části protokolu uchovávané v této datové sadě.

- Inventář *wrap around* poslední aktivity produktu IBM MQ . Tento stav je nutný, pokud je třeba správce front restartovat.

BSDS je povinná, pokud má správce front chybu a vy ji musíte restartovat. IBM MQ **musí** mít BSDS. Chcete-li minimalizovat pravděpodobnost výskytu problémů během restartu, můžete nakonfigurovat produkt IBM MQ s duálními BSDS, přičemž každý záznam bude zaznamenávat stejné informace. Použití duálních BSDS je známo jako spuštění v *duálním režimu*. Je-li to možné, umístěte kopie na samostatné svazky. Tím se snižuje riziko ztráty nebo zničení svazku, pokud je tento nosič poškozen nebo zničen. Namísto duálního zápisu do DASD používejte duální BSDS (BSDS).

BSDS je nastaven, když je produkt IBM MQ přizpůsobený a můžete spravovat zásoby pomocí obslužného programu změn protokolu změn ( CSQJU003 ). Další informace o tomto obslužném programu viz [Administrace IBM MQ for z/OS](#). Odkazuje na něj příkaz DD v rámci procedury spuštění správce front.

Za normálních okolností příkaz IBM MQ uchovává duplicitní kopie BSDS. Dojde-li k chybě I/O, dealokuje selhávající kopii a bude pokračovat s jednou BSDS. Operace dual-mode můžete obnovit, což je popsáno v publikaci [Administrace IBM MQ for z/OS](#).

Aktivní protokoly jsou poprvé registrovány v BSDS, když je nainstalován produkt IBM MQ . Aktivní protokoly nemůžete nahradit bez ukončení a restartování správce front.

Archivní datové sady protokolů jsou alokovány dynamicky. Je-li alokována jedna, je název datové sady registrován v BSDS. Seznam datových sad protokolu archivace se rozbálí jako archivy a zabalí se, když bylo dosaženo uživatelem určeného počtu záznamů. Maximální počet položek je 1000 pro protokolování jednoho archivu a 2000 pro duální protokolování.

K odstranění datových sad protokolu archivace můžete použít systém správy pásek ( IBM MQ nemá automatizovanou metodu). Proto mohou být informace o datové sadě protokolu archivu v BSDS po odstranění archivní datové sady systémem administrátorem systému.

Naopak, maximální počet datových sad protokolu archivace mohl být překročen a data z BSDS budou zahozena dlouho předtím, než bude datová sada dosažena své datum vypršení platnosti.

Můžete použít následující příkaz MQSC k určení rozsahu protokolu a názvu aktivní nebo archivní datové sady, která uchovává nejdřívější protokol RBA protokolu, požadovanou pro různé typy médií nebo zotavení správce front:

```
DISPLAY USAGE TYPE(DATASET)
```

Pokud modul parametrů systému uvádí, že datové sady protokolu archivu jsou katalogizovány, jsou BSDS odkazovány na katalog ICF (Integrated Catalog Facility) pro informace potřebné pro pozdější alokace. Jinak záznamy BSDS pro každý svazek zaregistrují sériové číslo svazku a informace o jednotce, které jsou potřebné pro pozdější alokace.

## Verze BSDS

Formát BSDS se liší v závislosti na jeho verzi. Zvýšení verze sady BSDS umožňuje použití nových funkcí. Produkt IBM MQ podporuje následující verze BSDS:

### Verze 1

Podporováno všemi vydáními produktu IBM MQ. BSDS verze 1 podporuje 6bajtové hodnoty protokolu RBA protokolu 6 bajtů.

### verze 2

Podporováno systémem IBM MQ 8.0 a vyšším. BSDS verze 2 povoluje 8bajtové hodnoty protokolu RBA protokolu a až 310 datových sad v každé aktivní kopii protokolu.

**V 9.2.5** Ve výchozím nastavení je pro správce front vytvořeném v produktu IBM MQ 9.2.4 a vyšším povoleno výchozí nastavení.

### verze 3

Podporováno systémem IBM MQ 8.0 a vyšším. BSDS se automaticky konvertuje na verzi 3, z verze 2, když je přidáno více než 31 datových sad do jedné aktivní kopie protokolu.

Verzi sady BSDS můžete určit spuštěním obslužného programu pro mapování tiskových protokolů (CSQJU004). Chcete-li převést sadu BSDS z verze 1 na verzi 2, spusťte obslužný program převodu BSDS (CSQJUCNV).

Další informace o 6bajtovém a 8bajtovém protokolu RBA najdete v příručce [“Větší relativní adresa bajtů protokolu”](#) na stránce 228.

## Archivní datové sady protokolu a kopie BSDS

Při každém vytvoření nové datové sady protokolu archivu bude vytvořena také kopie BSDS. Je-li archivní protokol na pásce, BSDS je první datovou sadou na prvním výstupním svazku. Je-li protokol archivace na DASD, BSDS je oddělená datová sada.

Názvy datových sad archivního protokolu a BSDS jsou stejné, kromě toho, že kvalifikátor nejnižší úrovně názvu protokolu archivace začíná na A a BSDS kopie začíná s B, například:

### Název protokolu archivace

CSQ.ARCHLOG1.E00186.T2336229. A 0000001

### BSDS-název kopie

CSQ.ARCHLOG1.E00186.T2336229. B 0000001

Pokud při kopírování BSDS došlo k chybě čtení, kopie nebude vytvořena, bude vydána zpráva [CSQJ125E](#) a odkládání do nové datové sady protokolu archivu bude pokračovat bez kopie BSDS.

## Definování systému v systému z/OS

Produkt IBM MQ for z/OS používá mnoho výchozích definic objektů a poskytuje ukázkou JCL pro vytvoření těchto výchozích objektů. Toto téma slouží k pochopení těchto výchozích objektů a ukázky JCL.

## Nastavení systémových parametrů

V produktu IBM MQ for z/OS modul parametrů systému řídí prostředí protokolování, archivace, trasování a připojení, které produkt IBM MQ používá ve své operaci. Parametry systému jsou určeny třemi makry assembleru následujícím způsobem:

### CSQ6SYSP

Systémové parametry, včetně nastavení prostředí pro připojení a trasování.

### CSQ6LOGP

Parametry protokolování.

### CSQ6ARVP

Parametry archivu protokolu.

Výchozí moduly parametrů jsou dodávány s produktem IBM MQ for z/OS. Pokud tyto hodnoty neobsahují hodnoty, které chcete použít, můžete vytvořit své vlastní moduly parametrů s použitím ukázky dodané s produktem IBM MQ. Ukázka je `th1qua1.SCSQPROC (CSQ4ZPRM)`.

Při spuštění správce front můžete některé systémové parametry změnit. 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 232
- [“Vyladění správce front v systému IBM MQ for z/OS”](#) na stránce 236
- [“Ukázkové definice dodávané s produktem IBM MQ for z/OS”](#) na stránce 237

## Související pojmy

[Zdroje, ze kterých můžete zadávat příkazy MQSC v systému z/OS](#)

## Související úlohy

[Upravit vstupní datové sady inicializace ukázek](#)

[Správa serveru z/OS](#)

## Definování systémových objektů pro IBM MQ for z/OS

Produkt IBM MQ for z/OS vyžaduje další předdefinované objekty pro aplikace typu publikování/odběr, klastru a řízení kanálů a další funkce správy systému.

Systémové objekty vyžadované produktem IBM MQ for z/OS mohou být rozděleny do následujících kategorií:

- [Publikování/odběr objektů](#)
- [Systémové výchozí objekty](#)
- [Systémové příkazové objekty](#)
- [Objekty administrace systému](#)
- [Fronty kanálů](#)
- [Fronty klastru](#)
- [Fronty skupiny sdílení front](#)
- [Paměťové třídy](#)
- [Definování fronty zablokovaných objektů systému](#)
- [Výchozí přenosová fronta](#)
- [Interní fronty](#)
- [“Fronta ověření kanálu” na stránce 236](#)

### Objekty publikování/odběru

Existuje několik objektů systému, které je třeba definovat, než budete moci používat aplikace publikování/odběru s produktem IBM MQ for z/OS. K definování těchto objektů jsou dodávány vzorové definice s IBM MQ. Tyto ukázky jsou popsány v [CSQ4INSG](#).

Chcete-li použít publikování/odběr, je třeba definovat následující objekty:

- Lokální fronta s názvem SYSTEM.RETAINED.PUB.QUEUE, která se používá k uložení kopie každé zachované publikace ve správci front. Každé úplné jméno tématu může mít až jednu zachovanou publikaci uloženou v této frontě. Pokud vaše aplikace budou používat zachovaná publikování v mnoha různých tématech nebo jsou-li vaše zachované publikační zprávy velké zprávy, požadavky na úložiště pro tuto frontu by měly být pečlivě naplánovány, včetně přiřazení k vlastní sadě stránek, pokud jsou požadavky na úložiště velké. Chcete-li zlepšit výkon, měli byste definovat tuto frontu s typem indexu MSGID (jak je zobrazeno v zadané definici vzorové 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 definovat tuto frontu s typem indexu CORRELID (jak je zobrazeno v zadané definici vzorové 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 netrvalé 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, aby se shodovaly s objekty tématu k odběru bodů odběru.
- Téma s názvem SYSTEM.BASE.TOPIC, které se používá jako základní téma pro interpretaci atributů.



- Téma s názvem SYSTEM.BROKER.DEFAULT.STREAM, což je výchozí proud používaný rozhraním pro publikování/odběr ve frontě.
- Téma s názvem SYSTEM.BROKER.DEFAULT.SUBPOINT, což je výchozí bod odběru RFH2 použitý rozhraním pro publikování/odběr ve frontě.
- Téma s názvem SYSTEM.BROKER.ADMIN.STREAM, což je administrativní proud používaný rozhraním pro publikování/odběr ve frontě.
- Předplatné SYSTEM.DEFAULT.SUB, což je výchozí objekt odběru používaný k poskytnutí výchozích hodnot v příkazech DEFINE SUB.

## Systémové výchozí objekty

Systémové výchozí objekty se používají k poskytnutí výchozích atributů, když definujete objekt a nespecifikujete název jiného objektu, na kterém má být definice založena.

Názvy výchozích definic systémových objektů začínají znaky "SYSTEM.DEFAULT"nebo"SYSTEM.DEF. " Například výchozí lokální fronta systému je pojmenována SYSTEM.DEFAULT.LOCAL.QUEUE.

Tyto objekty definují výchozí nastavení 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, definice je založena na SYSTEM.DEFAULT.LOCAL.QUEUE. Musíte si zapamatovat, že chcete dodat hodnotu pro název struktury prostředku Coupling Facility, protože není uvedena ve výchozí definici. Alternativně můžete definovat svou vlastní výchozí definici sdílené fronty, která má být použita jako základ pro sdílené fronty tak, aby všechny dělily požadované atributy. Nezapomeňte, že je třeba definovat sdílenou frontu v jednom správci front pouze ve skupině sdílení front.

## Systémové příkazové objekty

Názvy systémových příkazů začínají znaky SYSTEM.COMMAND. Tyto objekty musíte definovat dříve, než budete moci používat operace IBM MQ a ovládací panely pro zadávání příkazů do subsystému IBM MQ .

Jsou zde dva objekty systémových příkazů:

1. Vstupní fronta systémového příkazu je lokální fronta, na které jsou příkazy vloženy před tím, než jsou zpracovány příkazovým procesorem IBM MQ . Musí se jmenovat SYSTEM.COMMAND.INPUT. Alias nazvaný SYSTEM.ADMIN.COMMAND.QUEUE by měla být také definována, kvůli kompatibilitě s IBM MQ for Multiplatformsa pro použití v MQ Console a administrative REST API.
2. SYSTEM.COMMAND.REPLY.MODEL je modelová fronta, která definuje frontu systémových odpovědí-na frontu.

K dispozici jsou dva objekty navíc pro použití produktem IBM MQ Explorer:

- SYSTEM.MQEXPLORER.REPLY.MODEL fronta
- SYSTEM.ADMIN.SVRCONN kanál

SYSTEM.REST.REPLY.QUEUE je fronta odpovědí používaná serverem IBM MQ administrative REST API.

Příkazy se normálně odesílají pomocí přechodných zpráv, takže objekty systémových příkazů by měly mít atribut DEFPSIST (NO), takže aplikace, které je používají (včetně dodaných aplikací, jako je obslužný program a operace a řídicí panely), mají standardně přechodné zprávy. Máte-li aplikaci, která používá trvalé zprávy pro příkazy, nastavte atribut DEFTYPE (PERMDYN) pro frontu pro odpověď na frontu, protože zprávy odpovědi na tyto příkazy jsou trvalé.

## Objekty administrace systému

Jména objektů administrace systému začínají znaky SYSTEM.ADMIN.

K dispozici je sedm objektů administrace systému:

- SYSTEM.ADMIN.CHANNEL.EVENT
- SYSTEM.ADMIN.COMMAND.EVENT
- SYSTEM.ADMIN.CONFIG.EVENT
- SYSTEM.ADMIN.PERFM.EVENT
- SYSTEM.ADMIN.QMGR.EVENT
- SYSTEM.ADMIN.TRACE.ROUTE.QUEUE
- SYSTEM.ADMIN.ACTIVITY.QUEUE

## Fronty kanálů

Chcete-li používat distribuované ukládání do fronty, je třeba definovat následující objekty:

- Lokální fronta s názvem SYSTEM.CHANNEL.SYNCQ, která se používá k udržování pořadových čísel a logických jednotek identifikátorů práce (LUWID) kanálů. Chcete-li zlepšit výkon kanálu, měli byste definovat tuto frontu s typem indexu MSGID (jak je zobrazeno v zadané definici vzorové 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 použít klastry produktu IBM MQ, je třeba 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 všechna místa určená v klastru. Z výkonnostních důvodů byste měli tuto frontu definovat s typem indexu CORRELID (jak je zobrazeno v definici vzorové fronty).

Tyto fronty obvykle obsahují velké počty zpráv.

Tyto fronty nelze definovat jako sdílené fronty.

## Fronty skupiny 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, která 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, je třeba tuto frontu definovat 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. Další definice paměťové třídy jsou doporučeny, protože jsou použity v ukázkových definicích front.

### VÝCHOZÍ (požadováno)

Tato paměťová třída se používá pro všechny fronty zpráv, které nejsou kritické pro výkon a které se nevejdou do žádné z ostatních paměťových tříd. Pokud při definování fronty neurčíte žádnou výchozí paměťovou třídu, je to také dodaná výchozí třída úložiště.

### NOSDEFINE (povinné)

Tato třída úložiště se používá, pokud není definována třída ukládání, když definujete frontu.

### REMOTE (požadováno)

Tato paměťová třída se používá primárně pro přenosové fronty, tj. fronty související se systémem s krátkou životností-kritické zprávy.

### SYSLNGLV

Tato třída ukládání se používá pro dlouhotrvající zprávy o výkonu.

### SYSTEM (povinné)

Tato paměťová třída se používá pro kritické kritické situace, fronty zpráv související se systémem, například SYSTEM.CHANNEL.SYNQ a SYSTEM.CLUSTER fronty.

### SYSVOLLAT

Tato paměťová třída se používá pro krátkodobé, výkonové kritické zprávy.

Podle potřeby můžete upravit jejich atributy a přidat další definice třídy úložiště.

## Definování fronty zablokovaných objektů systému

Fronta nedoručených zpráv se používá v případě, že místo určení zprávy není platné. Příkaz IBM MQ vkládá tyto zprávy do lokální fronty s názvem fronty nedoručených zpráv. Ačkoli není fronta nedoručených zpráv povinná, je třeba ji považovat za nezbytnou, zvláště pokud používáte distribuované řazení do fronty nebo jeden z mostů IBM MQ .

**Ne** definujte frontu nedoručených zpráv jako sdílenou frontu. Do fronty zablokovaných dopisů může být vložena do lokální fronty v jednom správci front. Pokud byla fronta nedoručených zpráv sdílená fronta, může 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 měl jiný profil zabezpečení. Pokud fronta neexistuje, nepodaří se ji znovu zpracovat.

Rozhodnete-li se definovat frontu nedoručených zpráv, je třeba jeho název také sdělit správci front. Chcete-li to provést, použijte příkaz ALTER QMGR DEADQ (*název-fronty*). Další informace viz [Zobrazení a změna atributů správce front](#).

## Výchozí přenosová fronta

Výchozí přenosová fronta se použije, když není k dispozici žádná jiná vhodná přenosová fronta pro odesílání zpráv do jiného správce front. 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 jeho název. Chcete-li to provést, použijte příkaz ALTER QMGR.

## Vnitřní fronty

### • Nevyřízená datová fronta

- Fronta definovaná pro vnitřní použití, SYSTEM.PENDING.DATA.QUEUE podporuje použití trvalých odběrů v prostředí typu publikování/odběru JMS .

### • JMS 2.0 výstupní fronta odložení doručení

- Pokud je použita funkce zpoždění doručení poskytnutá serverem JMS 2.0 , pak bude interní fázová fronta SYSTEM.DDELAY.LOCAL.QUEUE, musí být definována. Tato fronta je používána správcem front k dočasnému ukládání zpráv odeslaných s nenulovým doručovacím zpožděním do dokončení prodlevy doručení a zpráva je vložena do cílového místa určení. K dispozici je definice ukázky pro tuto frontu, komentář, v CSQ4INSG.
- Definujete-li SYSTEM.DDELAY.LOCAL.QUEUE , musíte nastavit atributy STGCLASS, MAXMSGL a MAXDEPTH pro očekávaný počet zpráv, které budou odeslány s prodlevou doručení. Navíc při definování SYSTEM.DDELAY.LOCAL.QUEUE Fronta se ujistěte, že do této fronty může vkládat zprávy pouze správce front. Měli byste dbát na to, aby žádný identifikátor uživatele neměl oprávnění vkládat zprávy do této fronty.

## Fronta ověření kanálu

Pro interní použití kanálu ověření kanálu je SYSTEM.CHLAUTH.DATA.QUEUE je povinná. K definování těchto objektů jsou dodávány vzorové definice s IBM MQ . Tato ukázka je popsána v souboru CSQ4INSA, který 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 správce front vyladěn, abyste se vyhnuli základním problémům s výkonem.

Existuje celá řada způsobů, jak zlepšit výkon správce front, který je řízen atributy správce front nastavenými příkazem ALTER QMGR. Tato sekce obsahuje informace o tom, jak to lze provést nastavením maximálního počtu zpráv povolených ve správci front nebo provedením 'ucheping' ve správci front. Produkt IBM MQ SupportPac MP16 - WebSphere MQ pro z/OS Plánování kapacity & ladění poskytuje více informací o výkonu a ladění.

## Synchronizační body

Jedna z rolí správce front je řízením synchronizačního bodu v rámci aplikace. Aplikace vytvoří jednotku práce obsahující libovolný počet volání MQPUT nebo MQGET s volaným voláním MQCMIT.

Vzhledem k tomu, že počet volání MQPUT nebo MQGET v rámci jednoho zvýšení MQCMIT se zvyšuje, výrazně se zvýší náklady na výkon pro provedení operace commit. Aplikace, obecně, by měly být navrženy tak, aby neMQPUT/MQPUT nevyužívají velký počet zpráv v jednom synchronizačních bodech.

Administrativně můžete omezit počet zpráv v rámci jednoho synchronizačního bodu pomocí atributu správce front MAXUMSGS. Pokud aplikace překročí tento limit, obdrží volání MQRC\_SYNCPOINT\_LIMIT\_REACHED na volání MQPUT, MQPUT1 nebo MQGET, které přesahuje mezní hodnotu. Aplikace by pak měla vydat příkaz MQCMIT nebo MQBACK podle potřeby.

Výchozí hodnota MAXUMSGS je 10000. Tato hodnota může být snížena, chcete-li vynutit nižší limit, což může také pomoci při ochraně proti cyklování aplikací. Před snížením hodnoty MAXUMSGS se ujistěte, že rozumíte vašim existujícím aplikacím, aby se zajistilo, že nepřekročily limit, nebo mohou tolerovat návratový kód MQRC\_SYNCPOINT\_LIMIT\_REACHED

## Vypršelé zprávy

Platnost zpráv, jejichž platnost vypršela, budou zahozeny při dalším příslušném volání MQGET. Pokud však takové volání neexistuje, zprávy s vypršenou platností se nevyřazují a v případě některých front, zejména těch, kde je načítání zpráv provádí MessageId, CorrelId nebo GroupId a fronta je indexována pro výkon, se může hromadit řada zpráv s vypršenou platností. Správce front může pravidelně skenovat všechny fronty pro zprávy s ukončenou platností, které jsou poté odstraněny. Můžete zvolit, jak často se bude toto skenování provádět, pokud vůbec. Existují dva způsoby, jak to provést:

#### Explicitní požadavek

Můžete určit, které fronty budou skenovány a kdy. Zadejte příkaz REFRESH QMGR TYPE (EXPIRY) a určete frontu nebo fronty, které chcete skenovat.

#### Pravidelné skenování

Interval vypršení platnosti můžete zadat v objektu správce front pomocí atributu EXPRYINT. Správce front udržuje informace o zprávách s vypršenou platností na každé frontě a ví o tom, jakou dobu má skenování vypršelych zpráv za to, že se vyplatí. Pokaždé, když je dosažen interval EXPRYINT, vyhledá správce front kandidátské fronty, které stojí za skenování zpráv s vypršenou platností, a prohledá pouze ty fronty, které považuje za užitečné. Neskenuje všechny fronty. Tím se vyvarujete jakéhokoli času procesoru, který je zbytečný na zbytečném skenování

Sdílené fronty jsou skenovány pouze jedním správcem front v rámci skupiny sdílení front. Obecně platí, že první správce front, který se má restartovat, nebo první, který má sadu EXPRYINT provést, 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 slouží jako reference pro ukázkový kód JCL a kód dodávaný 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 úpravě vašich vlastních objektů. Některé z nich můžete zahrnout do vstupních datových sad inicializace (popsáno v tématu [Inicializační příkazy](#) ).

| <i>Tabulka 21. Vzorové definice IBM MQ pro systémové objekty</i> |                                                                                                                                                                       |
|------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>vstupní datová sada inicializace</b>                          | <b>Vzorový název</b>                                                                                                                                                  |
| <u>CSQINP1</u>                                                   | CSQ4INP1<br>CSQ4INPR                                                                                                                                                  |
| <u>CSQINP2</u>                                                   | CSQ4INSA<br>CSQ4INYS <sup>1</sup><br>CSQ4INSX<br>CSQ4INSG<br>CSQ4INSR<br>CSQ4INSS<br>CSQ4INSJ<br>CSQ4INSM<br>CSQ4INYG<br>CSQ4INYR<br>CSQ4INYC<br>CSQ4INYD<br>CSQ4INSC |
| <u>CSQINPT</u>                                                   | CSQ4INST<br>CSQ4INYT                                                                                                                                                  |

Tabulka 21. Vzorové definice IBM MQ pro systémové objekty (pokračování)

| vstupní datová sada inicializace | Vzorový název                                                                            |
|----------------------------------|------------------------------------------------------------------------------------------|
| Jiná                             | CSQ4DISP<br>CSQ4INPX<br>CSQ4IVPQ<br>CSQ4IVPG<br>CSQ4MSTR<br>CSQ4MSRR<br>V 9.2.0 CSQ4QMIN |

**Poznámka:**

1. Pořadí těchto definic vzorku je důležité: dojde k chybě, pokud jsou INYS, INSX a INSG nesprávně objednány.

**Ukázky CSQINP1**

Použijte vzorovou datovou sadu CSQINP1 thlqual.SCSQPROC(CSQ4INP1), když používáte jednu sadu stránek pro každou třídu zprávy, nebo thlqual.SCSQPROC(CSQ4INPR) při použití více sad stránek pro hlavní třídy zprávy. Obsahuje definice fondů vyrovnávacích pamětí, sady stránek do asociací fondů vyrovnávacích pamětí a příkaz ALTER SECURITY. Začleňte ukázkou do zřetězení CSQINP1 spuštěné procedury spuštěné úlohy správce front.

**Ukázky CSQINP2**

**Ukázkový systémový objekt CSQ4INSG**

Ukázka datové sady CSQINP2 thlqual.SCSQPROC(CSQ4INSG) obsahuje definice pro následující systémové objekty pro obecné použití:

- Systémové výchozí objekty
- Systémové příkazové objekty
- Objekty administrace systému
- Jiné objekty pro použití systémem

Musíte definovat objekty v této ukázce, ale musíte ji provést pouze jednou, když je podsystém poprvé spuštěn. Nejlepším způsobem, jak toho dosáhnout, je zahrnutí definic do datové sady CSQINP2 . Jsou udržovány po ukončení a restartování správce front. Nesmíte měnit názvy objektů, ale můžete změnit jejich atributy, je-li to nutné.

Když jsou splněny následující podmínky, jedna zpráva se umístí do SYSTEM.DURABLE.SUBSCRIBER.QUEUE fronta (i v případě, že publikování odběru není aktivní):

- Atribut QMGR PSMODE je nastaven na hodnotu DISABLED
- Ukázkový objekt CSQ4INST ukázkového objektu DEFINE SUB( ' SYSTEM . DEFAULT . SUB ' ) je přítomen.

Chcete-li se tomuto příkazu vyhnout, odstraňte nebo označte jako komentář příkaz DEFINE SUB( ' SYSTEM . DEFAULT . SUB ' ) .

Pracovní fronta zpoždění doručení JMS 2.0 , SYSTEM.DDELAY.LOCAL.QUEUE je třeba definovat pouze tehdy, je-li použita prodleva doručení JMS 2.0 . Při výchozím nastavení je definice fronty označena jako komentář, což můžete zrušit, je-li to nutné.

## **Systémový objekt CSQ4INSA a ukázka ověření**

Ukázka datové sady CSQINP2 thlqual.SCSQPROC(CSQ4INSA) obsahuje definici systémové fronty pro ověření kanálu. Tato fronta zadržuje záznamy ověření kanálu. Obsahuje také výchozí pravidla ověření kanálu.

Objekty v této ukázce musíte definovat, je-li parametr CHLAUTH na správci front POVOLENO a chcete spustit kanály, nebo chcete použít záznam SET nebo DISPLAY CHLAUTH. Je třeba je definovat pouze jednou při prvním spuštění subsystému. Nejlepším způsobem, jak toho dosáhnout, je zahrnutí definic do datové sady CSQINP2. Jsou udržovány v rámci ukončení a restartování správce front, nesmíte změnit název fronty.

## **Ukázka systémového objektu CSQ4INSS**

Pokud používáte skupiny sdílení front, můžete definovat další systémové objekty.

Ukázková datová sada thlqual.SCSQPROC(CSQ4INSS) obsahuje ukázkové příkazy pro použití se strukturami CF a sadou definic pro systémové objekty vyžadované pro sdílené kanály a řazení do front v rámci skupiny.

Tuto ukázku nemůžete použít jako takovou; musíte ji upravit před použitím. Poté můžete tento člen zahrnout do zřetězení definic dat CSQINP2 z procedury spuštění správce front, nebo jej můžete použít jako vstup pro funkci COMMAND obslužného programu CSQUTIL, chcete-li vydat požadované příkazy.

Když definujete skupinu nebo sdílené objekty, musíte je zahrnout do zřetězení definic dat CSQINP2 pouze pro jednoho správce front ve skupině sdílení front.

## **Ukázka systémového objektu CSQ4INSX**

Pokud používáte distribuované fronty a klastrování, musíte definovat další systémové objekty.

Vzorová datová sada thlqual.SCSQPROC(CSQ4INSX) obsahuje požadované definice fronty. Tento člen můžete zahrnout do zřetězení definic dat CSQINP2 z procedury spuštění správce front, nebo jej můžete použít jako vstup pro funkci COMMAND v obslužném programu CSQUTIL, chcete-li vydat požadované příkazy DEFINE.

Existují dva typy definic objektů:

- SYSTEM.CHANNEL.xx, potřebný pro všechny distribuované fronty.
- SYSTEM.CLUSTER.xx, potřebný pro klastrování

## **Ukázkový objekt systému CSQ4INSJ JMS**

Definuje fronty použité v doméně publikování/odběru JMS .

## **Ukázkový systémový objekt CSQ4INSM**

Používáte-li rozšířenou zabezpečení zpráv, je třeba definovat další systémové objekty. Vzorová datová sada thlqual.SCSQPROC(CSQ4INSM) obsahuje požadované definice fronty.

## **Ukázkový objekt CSQ4INSR**

Definuje fronty použité produktem WebSphere Application Server a zprostředkovateli.

## **Ukázkový objekt CSQ4INYD**

Pokud používáte distribuované fronty a potřebujete nastavit vlastní fronty, procesy a kanály.

Ukázková datová sada thlqual.SCSQPROC(CSQ4INYD) obsahuje ukázkové definice, které lze použít pro přizpůsobení distribuovaných objektů řazení do fronty. Zahrnuje:

- Sada definic pro odesílající konec
- Sada definic pro přijímající konec
- Sada definic pro použití klientů

Tento vzorek nemůžete použít jako je-musíte ji před použitím upravit. Pak můžete tento člen zahrnout do zřetězení definic dat CSQINP2 z procedury spuštění správce front, nebo jej můžete použít jako vstup pro funkci COMMAND obslužného programu CSQUTIL, abyste mohli vydat požadované příkazy DEFINE. (Toto je vhodnější, protože to znamená, že nemusíte předefinovat tyto objekty pokaždé, když restartujete správce front).

### **Ukázkový objekt CSQ4INYC**

Pokud používáte klastrování, jsou definice ekvivalentní definicím kanálu a definice vzdálených front distribuovaných front vytvářeny automaticky, je-li to potřeba. Je však třeba definovat některé ruční definice kanálů-kanál příjemce klastru pro klastr a definice odesílatele klastru alespoň k jednomu správci front úložiště klastru.

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INYC) obsahuje následující definice ukázek, které můžete použít k úpravě vašich klastrovaných objektů:

- Definice pro správce front
- Definice přijímajícího kanálu
- Definice pro odesílající kanál
- Definice pro fronty klastru
- Definice pro seznamy klastrů

Tento vzorek nemůžete použít jako je-musíte ji před použitím upravit. Pak můžete tento člen zahrnout do zřetězení definic dat CSQINP2 z procedury spuštění správce front, nebo jej můžete použít jako vstup pro funkci COMMAND obslužného programu CSQUTIL, abyste mohli vydat požadované příkazy DEFINE. To je vhodnější, protože to znamená, že nemusíte znovu definovat tyto objekty při každém spuštění produktu IBM MQ.

### **Ukázkový objekt CSQ4INYG**

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INYG) obsahuje následující ukázkové definice, které můžete použít k přizpůsobení vlastních objektů pro obecné použití:

- Fronta nedoručených zpráv
- Výchozí přenosová fronta
- Objekty adaptéru CICS

Tento vzorek nemůžete použít jako je-musíte ji před použitím upravit. Pak můžete tento člen zahrnout do zřetězení definic dat CSQINP2 z procedury spuštění správce front, nebo jej můžete použít jako vstup pro funkci COMMAND obslužného programu CSQUTIL, abyste mohli vydat požadované příkazy DEFINE. To je vhodnější, protože to znamená, že nemusíte znovu definovat tyto objekty při každém spuštění produktu IBM MQ.

Kromě zde uvedených definic ukázek můžete použít definice systémových objektů jako základ pro definice vašich vlastních prostředků. Například, můžete vytvořit pracovní kopii SYSTEM.DEFAULT.LOCAL.QUEUE a pojmenujte ji MY.DEFAULT.LOCAL.QUEUE. V této kopii můžete podle potřeby změnit kterýkoli z parametrů v této kopii. Poté můžete zadat příkaz DEFINE podle toho, jakou metodu zvolíte, za předpokladu, že máte oprávnění k vytvoření prostředků daného typu.

### **Výchozí přenosová fronta**

Před rozhodnutím, zda chcete definovat výchozí přenosovou frontu, si přečtěte popis [Výchozí přenosová fronta](#) .

- Rozhodnete-li se definovat výchozí přenosovou frontu, pamatujte na to, že je třeba definovat také kanál, který má sloužit.
- Pokud se rozhodnete, že ji nechcete definovat, nezapomeňte odebrat příkaz DEFXMLTQ ze zpracování příkazu ALTER QMGR ve vzorku.



## Objekty adaptéru CICS

Ukázka definuje inicializační frontu s názvem CICS01.INITQ. Tato fronta je používána transakcí CKTI dodávaným produktem IBM MQ . Můžete změnit název této fronty, ale musí odpovídat názvu uvedenému v tabulce inicializace systému CICS (SIT) nebo SYSEIN v příkazu INITPARM.

## Ukázky objektů CSQ4INYS/CSQ4INYR

Definice tříd úložiště pro použití:

- jedna stránka nastavena pro každou třídu zprávy
- více sad stránek pro hlavní třídy zprávy

Například SYSTEM.COMMAND.INPUT používá STGCLASS ('SYSVOLAT') a SYSTEM.CLUSTER.TRANSMIT.QUEUE používá STGCLASS ('REMOTE'). V CSQ4INYSobě paměťové třídy používají stejnou sadu stránek. V souboru CSQ4INYRtyto třídy ukládání používají různé sady stránek, aby se zmenšil dopad zaplnění přenosové fronty.

## Ukázky CSQINPT

### CSQ4INST

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INST) obsahuje definici pro výchozí odběr systému.

Musíte definovat objekt v této ukázce, ale musíte jej provést pouze jednou, když je poprvé spuštěn stroj publikování/odběru. Zahrnutí definice do datové sady CSQINPT je nejlepší způsob, jak toho dosáhnout. Je udržován mezi vypnutím a restartováním správce front. Nesmíte měnit název objektu, ale můžete změnit jejich atributy, je-li to nutné.

### CSQ4INYT

Ukázková datová sada: thlqual.SCSQPROC(CSQ4INYT) obsahuje sadu příkazů, které byste mohli chtít spustit při spuštění stroje pro publikování/odběr. 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 a definice IBM MQ , jako jsou paměťové třídy a trasování. Tyto příkazy mohou generovat velké množství výstupů. Tuto ukázkou 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 byste mohli chtít provést při každém spuštění inicializátoru kanálu. Tuto ukázkou si musíte před použitím upravit; můžete ji pak zahrnout do datové sady CSQINPX pro iniciátor kanálu.

### Ukázky CSQ4IVPQ a CSQ4IVPG

Ukázkové datové sady: thlqual.SCSQPROC(CSQ4IVPQ) a thlqual.SCSQPROC(CSQ4IVPG) obsahují sady příkazů DEFINE, které jsou vyžadovány ke spuštění programů pro verifikaci instalace (IVPs).

Tyto ukázky můžete zahrnout do datové sady CSQINP2 . Pokud jste IVP spustili úspěšně, nemusíte je znovu spouštět pokaždé, když se správce front restartuje. Proto nemusíte tyto ukázky trvale uchovávat v zřetězení CSQINP2 .

### Ukázky CSQ4MSTR a CSQ4MSRR

Jedná se o ukázkové procedury spuštěné úlohy pro správce front: thlqual.SCSQPROC(CSQ4MSTR) a thlqual.SCSQPROC(CSQ4MSRR).

CSQ4MSRR používá CSQ4INZR ve zřetězení CSQINP2 tak, aby se důležité fronty rozšířily napříč různými sadami stránek.

Můžete odebrat komentáře, abyste mohli v případě potřeby použít kartu CSQMINI pro nově vytvořené správce front.

## **V 9.2.0** Ukázka CSQ4QMIN

Ukázková datová sada QMINI, thlqual.SCSQPROC(CSQ4QMIN).

Podrobnosti o datové sadě QMINI a stanze **TransportSecurity** najdete v tématu [Datová sada QMINI](#).

## **z/OS** Zotavení a restartování v systému z/OS

Použijte odkazy v tomto tématu k vyhledání informací o funkcích produktu IBM MQ for z/OS pro restart a zotavení.

IBM MQ for z/OS má robustní funkce pro restart a obnovu. Informace o tom, jak se správce front po zastavení zotaví a co se stane po jeho restartu, naleznete v následujících dílčích tématech:

- [“Jak jsou prováděny změny dat v produktu IBM MQ for z/OS” na stránce 242](#)
- [“Jak konzistence se udržuje v produktu IBM MQ for z/OS” na stránce 244](#)
- [“Co se děje během ukončení v produktu IBM MQ for z/OS” na stránce 246](#)
- [“Co se děje během restartování a zotavení v produktu IBM MQ for z/OS” na stránce 247](#)
- [“Jak jsou vyřešeny neověřené jednotky zotavení” na stránce 249](#)
- [“Obnova sdílené fronty” na stránce 252](#)

### Související pojmy

**z/OS** Akce zotavení produktu IBM MQ for z/OS

[Zdroje, ze kterých můžete zadávat příkazy MQSC v systému z/OS](#)

### Související úlohy

[Plánování zálohování a obnovy](#)

**z/OS** Správa serveru z/OS

### Související odkazy

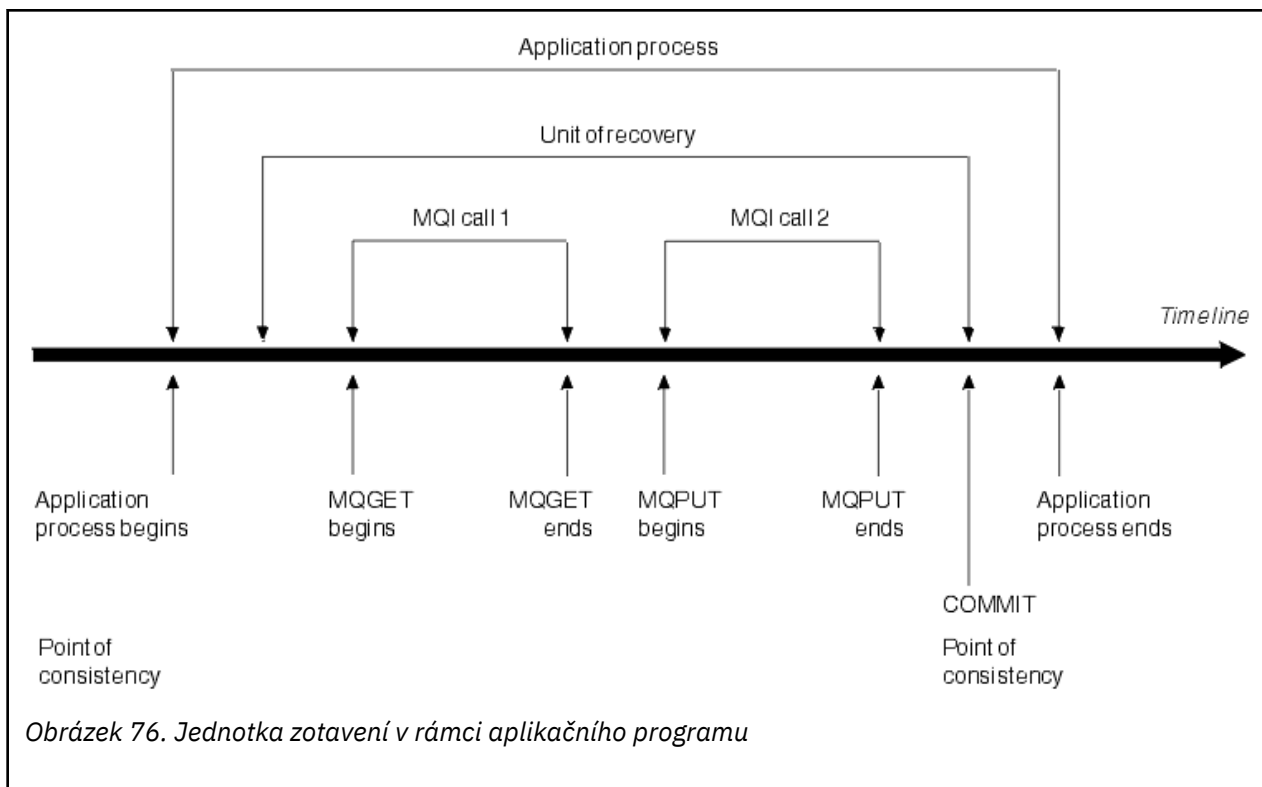
**z/OS** [Zprávy pro IBM MQ for z/OS](#)

## **z/OS** Jak jsou prováděny změny dat v produktu IBM MQ for z/OS

Produkt IBM MQ for z/OS musí spolupracovat s ostatními subsystémy, aby byla zachována konzistence všech dat. Toto téma obsahuje informace o *jednotkách zotavení*, o tom, jak jsou a jak se používají ve *zpětné platnosti*.

### Jednotky zotavení

*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* - také nazývaný *synchronizační bod* nebo *bod potvrzení* - je bod v čase, kdy jsou konzistentní všechna obnovitelná data, ke kterým přistupuje aplikační program.



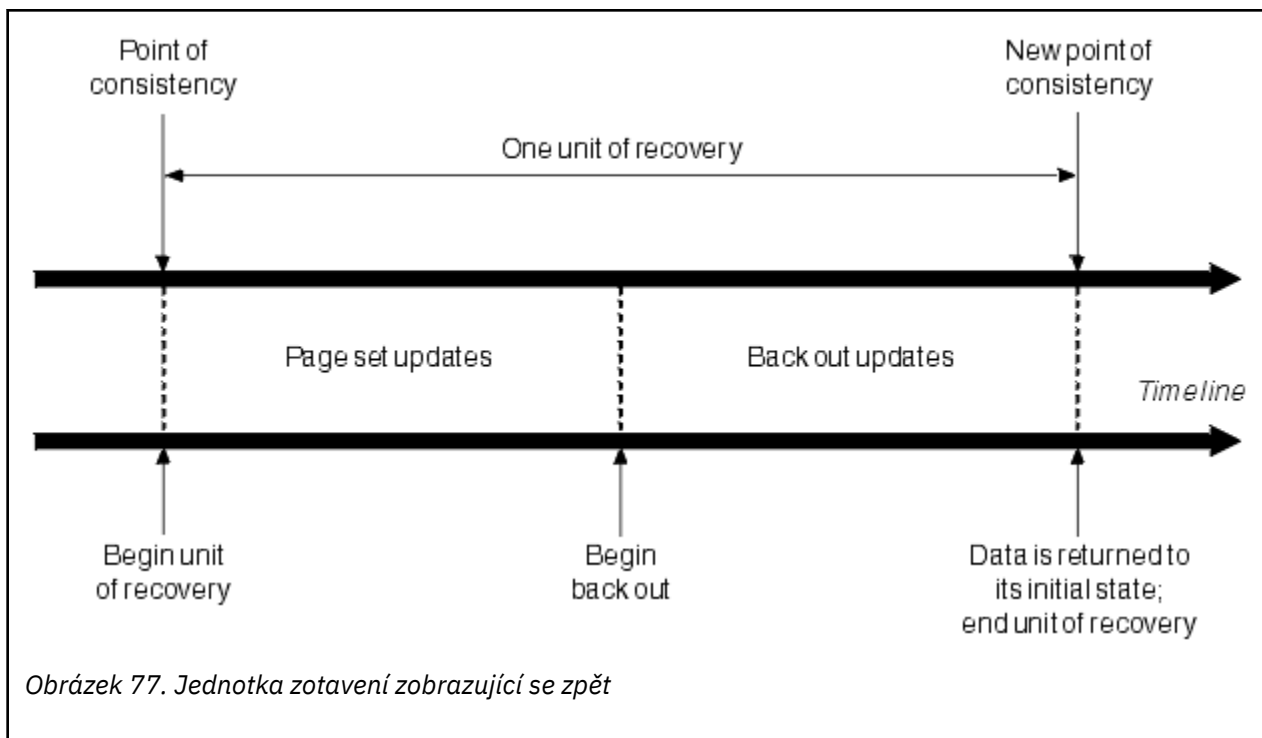
Jednotka zotavení začíná první změnou dat po začátku programu nebo po předchozím bodu konzistence; končí s pozdějším bodem konzistence. Obrázek 76 na stránce 243 ukazuje vztah mezi jednotkami obnovy, bodem konzistence a aplikačním programem. V tomto příkladu provádí aplikační program změny ve frontách prostřednictvím volání MQI 1 a 2. Aplikační program může obsahovat více než jednu jednotku zotavení nebo pouze jednu jednotku. Avšak jakákoli úplná jednotka zotavení se ukončí v bodu potvrzení.

Např. bankovní transakce převádí finanční prostředky z jednoho účtu do druhého. Za prvé, program odečte částku od prvního účtu, účet A. Pak přidá částku na druhý účet, B. Po odečtení částky od A, jsou tyto dva účty nekonzistentní a IBM MQ nelze potvrdit. Jakmile je částka přidána k účtu B, stanou se konzistentní. Když jsou oba kroky dokončeny, program může oznámit bod konzistence pomocí operace commit, čímž se změny zviditelní pro ostatní aplikační programy.

Normální ukončení aplikačního programu automaticky způsobí bod konzistence. Některé programové požadavky v programech CICS a IMS také způsobují konzistenci, například EXEC CICS SYNCPOINT.

## Zálohování práce

Dojde-li k chybě v rámci jednotky zotavení, produkt IBM MQ odstraní veškeré změny dat a vrátí data do stavu na začátku jednotky zotavení; to znamená, že produkt IBM MQ zálohuje tuto práci. Události jsou zobrazeny v [Obrázek 77 na stránce 244](#).



## z/OS Jak konzistence se udržuje v produktu IBM MQ for z/OS

Data v produktu IBM MQ for z/OS musí být konzistentní s dávkou, CICS, IMS nebo TSO. Jakákoli data změněná v jednom musí být porovnávána změnou v druhé.

Dříve než jeden systém potvrdí změněná data, musí vědět, že jiný systém může provést odpovídající změnu. Takže systémy musí komunikovat.

Během *Dvoufázového potvrzování* (například pod CICS) jeden subsystém koordinuje proces. Tento subsystém se nazývá *koordinátor*; druhý je *účastník*. CICS nebo IMS je vždy koordinátorem v interakci s produktem IBM MQ a IBM MQ je vždy účastníkem. V dávkovém prostředí nebo v prostředí TSO se produkt IBM MQ může podílet na protokolech dvoufázového potvrzování koordinovaných produktem z/OS RRS.

Během *jednofázového potvrzování* (například v rámci TSO nebo dávkového zpracování) je produkt IBM MQ vždy koordinátorem v rámci interakcí a zcela řídí proces potvrzování.

V prostředí produktu WebSphere Application Server určuje sémantika objektu relace JMS, zda se používá koordinace jednofázového nebo dvoufázového vázaného zpracování.

### Soulad s CICS nebo IMS

Připojení mezi IBM MQ a 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 distribuovaný protokol synchronizačního bodu. Zahrnuje více protokolování a toků zpráv než jednofázové potvrzování.

- Jednorázové operace commit-pro transakce, které aktualizují prostředky vlastněné jedním správcem prostředků (IBM MQ).

Tento protokol je optimalizován pro protokolování a toky zpráv.

- Vynechat synchronizační bod-pro transakce, které zahrnují produkt IBM MQ, ale které ve správci front nedělejte nic, co vyžaduje synchronizační bod (například procházení fronty).

V každém případě se produkt CICS nebo IMS chová jako správce synchronizačního bodu.

Fáze dvoufázového potvrzování, které produkt IBM MQ používá ke komunikaci s CICS nebo IMS, jsou následující:

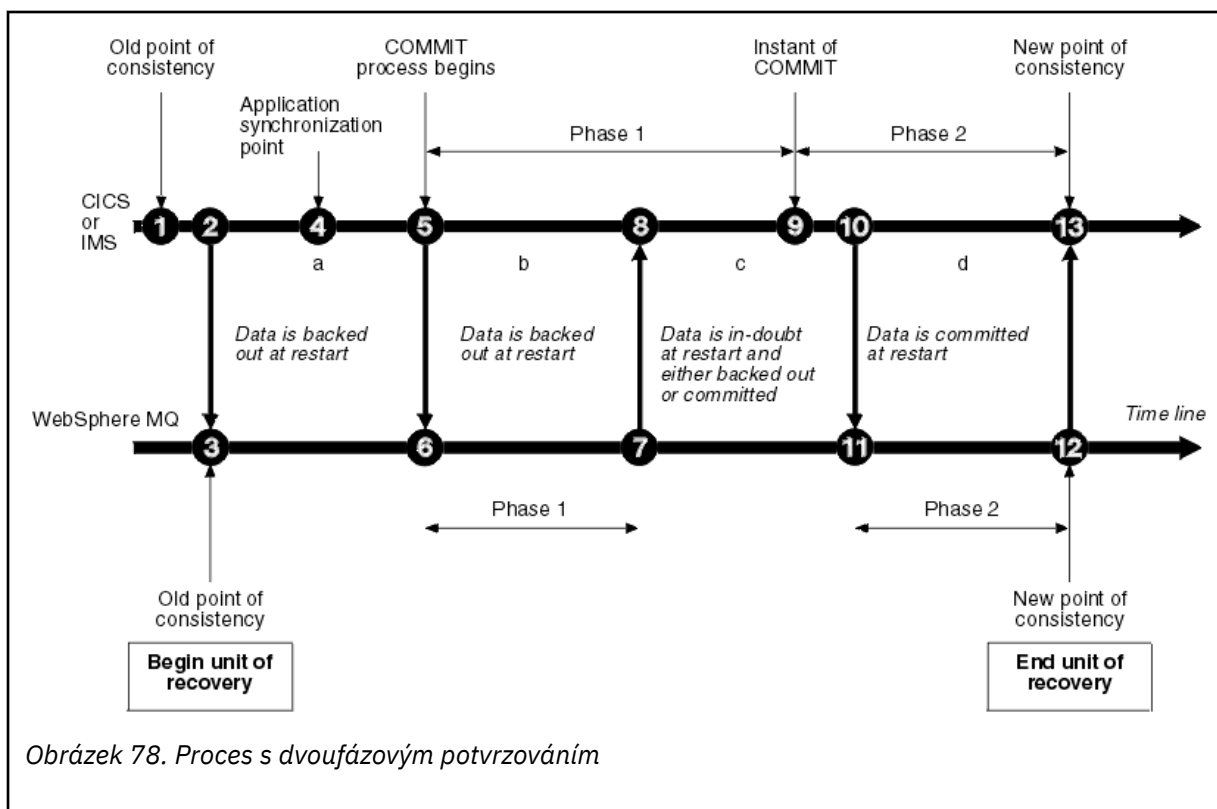
1. Ve fázi 1 určuje každý systém nezávisle na tom, zda zaznamenal ve svém protokolu dostatečné informace o obnově a může potvrdit svou práci.

Na konci fáze komunikují systémy. Pokud se shodují, každý z nich začíná další fázi.

2. Ve fázi 2 se změny stanou trvalými. Pokud se jeden ze systémů ukončí během fáze 2, operace se během restartu dokončí procesem obnovy.

### Ilustrace procesu dvoufázového potvrzování

Obrázek 78 na stránce 245 ilustruje proces dvoufázového potvrzování. Události v koordinátorovi CICS nebo IMS se zobrazují na horní čáře, události v IBM MQ na spodním řádku.



Obrázek 78. Proces s dvoufázovým potvrzováním

Čísla v následující sekci jsou propojena s čísly zobrazenými na obrázku.

1. Data ve koordinátorovi jsou v bodě konzistence.
2. Aplikační program v koordinátorovi volá produkt IBM MQ, aby aktualizoval frontu přidáním zprávy.
3. Tím se spustí jednotka zotavení v produktu IBM MQ.
4. Zpracování pokračuje v koordinátorovi až do dosažení synchronizačního bodu aplikace.
5. Koordinátor pak zahájí zpracování potvrzení. Programy produktu CICS používají příkaz SYNCPOINT nebo normální ukončení aplikace ke spuštění potvrzení. Programy typu IMS mohou spustit potvrzení pomocí volání CHKP, volání SYNC, volání GET UNIQUE u IOPCB nebo normálního ukončení aplikace. Fáze 1 zpracování potvrzení začíná.
6. Protože koordinátor zahájí zpracování fáze 1, takže IBM MQ.
7. Produkt IBM MQ úspěšně dokončí fázi 1, zapíše tuto skutečnost do svého protokolu a oznámí koordinátorovi.
8. Koordinátor obdrží oznámení.

9. Koordinátor úspěšně dokončí zpracování fáze 1. Nyní oba subsystemy souhlasí s potvrzením změn dat, protože obě strany dokončily fázi 1 a mohou se zotavit z jakýchkoli chyb. Koordinátor zaznamená ve svém protokolu okamžité potvrzení-neodvolatelné rozhodnutí dvou subsystemů, aby se změny projevíly.

Koordinátor nyní začíná fází 2 zpracování-skutečným závazkem.

10. Koordinátor upozorní produkt IBM MQ , aby zahájil fázi 2.
11. Produkt IBM MQ protokoluje začátek fáze 2.
12. Fáze 2 byla úspěšně dokončena a nyní se jedná o nový bod konzistence pro produkt IBM MQ. Produkt IBM MQ poté upozorní koordinátora, že dokončil zpracování fáze 2.
13. Koordinátor dokončí zpracování fáze 2. Data, která jsou řízena oběma subsystemy, jsou nyní konzistentní a dostupná pro jiné aplikace.

## Jak konzistence se udržuje po nestandardním ukončení

Je-li správce front restartován po nestandardním ukončení, musí určit, zda má být při ukončení činnosti aktivní nebo odvrácení jednotek zotavení aktivních. U některých jednotek zotavení má produkt IBM MQ dostatek informací, aby bylo možné rozhodnutí provést. Pro jiné nikoli, a při opětovném navázání spojení musí získat informace od koordinátora.

Obrázek 78 na stránce 245 zobrazuje čtyři období ve dvou fázích: a, b, c, a d. Stav jednotky zotavení závisí na období, ve kterém k ukončení došlo. Stav může mít jednu z následujících hodnot:

### V letu

Správce front byl ukončen před dokončením fáze 1 (období a nebo b); během restartu produkt IBM MQ zálohuje aktualizace.

### Nejisté

Správce front byl ukončen po dokončení fáze 1 a před začátkem fáze 2 (tečka c); pouze koordinátor ví, zda k chybě došlo před potvrzením nebo po něm (bod 9). Pokud k tomu došlo dříve, musí produkt IBM MQ zazálohovat své změny; pokud k tomu došlo po, musí produkt IBM MQ provést změny a potvrdit je. Při restartu IBM MQ čeká na informace od koordinátora před zpracováním této jednotky zotavení.

### Potvrzováním

Správce front byl ukončen poté, co zahájil vlastní zpracování fáze 2 (období d); provádí potvrzené změny.

### In backout

Správce front byl ukončen poté, co byla dokončena jednotka zotavení, ale před dokončením procesu (nezobrazeno na obrázku) během restartu, IBM MQ pokračuje v zálohování změn.

## Co se děje během ukončení v produktu IBM MQ for z/OS

Správce front je normálně ukončen v odezvě na příkaz STOP QMGR. Pokud se správce front zastaví z jakékoli jiné příčiny, ukončení je nestandardní.

Všimněte si, že během ukončení správce front IBM MQ interně vydává příkaz

```
DISPLAY CONN(*) TYPE(CONN) ALL WHERE (APPLTYPE NE SYSTEMAL)
```

takže jste si vědomi toho, jaká vlákna mohou správci front zabránit v dokončení ukončování práce systému.

SYSTEMAL odpovídá APPLTYPES buď SYSTEM nebo CHINIT, takže příkaz DISPLAY CONN filtrující typy aplikací neodpovídá systému SYSTEMAL, vrací do protokolu údaje informace o podprocesech, které mohou bránit normálnímu ukončení práce systému.

### Běžné ukončení

Při normálním ukončení IBM MQ zastaví všechny aktivity řádným způsobem. Produkt IBM MQ můžete zastavit pomocí režimu uvedení do klidového stavu, režimu vynucení nebo restartování. Efekty jsou uvedeny v části [Tabulka 22 na stránce 247](#).

| <i>Tabulka 22. Ukončení při použití QUIESCE, FORCE a RESTART</i> |                      |                |                |
|------------------------------------------------------------------|----------------------|----------------|----------------|
| <b>Typ podprocesu</b>                                            | <b>QUIESCE</b>       | <b>Vynutit</b> | <b>RESTART</b> |
| Aktivní podprocesy                                               | Spustit do dokončení | Vrátit zpět    | Vrátit zpět    |
| Nové podprocesy                                                  | Lze spustit          | Nepovoleno     | Nepovoleno     |
| Nová připojení                                                   | Nepovoleno           | Nepovoleno     | Nepovoleno     |

Dávkové aplikace jsou oznámeny, pokud dojde k ukončení, když je aplikace stále připojena.

V produktu CICS je aktuální podproces spuštěn pouze na konci jednotky zotavení. Při použití obslužného programu CICS zastaví správce front v režimu uvedení do klidového stavu adaptér CICS, a pokud aktivní úloha obsahuje více než jednu jednotku zotavení, nebude úloha nezbytně spuštěna až do dokončení.

Pokud zastavíte správce front v režimu vynucení nebo spuštění, nebudou alokovány žádné nové podprocesy a práce na připojených podprocesech se odvolají. Použití těchto režimů může vytvořit neověřené jednotky zotavení pro podprocesy, které jsou mezi fázemi zpracování potvrzení. Jsou vyřešeny, je-li IBM MQ znovu připojen k ovládacímu subsystému CICS, IMS nebo subsystému RRS.

Když zastavíte správce front, v jakémkoli režimu jsou tyto kroky:

1. Připojení jsou ukončena.
2. IBM MQ přestane přijímat příkazy.
3. Produkt IBM MQ zajišťuje, aby byly dokončeny všechny nevyřízené aktualizace sad stránek.
4. Příkaz DISPLAY USAGE je interně vydáván produktem IBM MQ tak, aby byla RBA restartovacího protokolu zaznamenána v protokolu konzoly produktu z/OS.
5. Je proveden kontrolní bod vypnutí a BSDS se aktualizuje.

Ukončení, které určují režim uvedení do klidového stavu, neovlivňují neověřené jednotky zotavení. Každá jednotka, která má pochybnosti, zůstává nejistá.

### **Nestandardní ukončení**

Nestandardní ukončení může zanechat data v nekonzistentním stavu, například:

- Jednotka zotavení byla přerušena dříve, než dosáhla 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 potvrzování, přičemž byla ponechána jednotka zotavení v nejistém stavu.

IBM MQ řeší veškeré nekonzistence dat vznikající při nestandardním ukončení během restartování a zotavení.

## **z/OS Co se děje během restartování a zotavení v produktu IBM MQ for z/OS**

Produkt IBM MQ používá svůj protokol pro zotavení a zaváděcí datovou sadu (BSDS) k určení, co má být po restartu zotavováno. BSDS identifikuje datové sady aktivního a archivního protokolu a umístění posledního kontrolního bodu IBM MQ v protokolu.

## Úvod do restartování a zotavení

Po inicializaci produktu IBM MQ se proces restartování správce front provede takto:

- Inicializace protokolu
- Opětovné sestavení aktuálního stavu
- Postoupit zotavení protokolu
- Zpětné zotavení protokolu
- Opětovné sestavení indexu fronty

Po dokončení obnovy:

- Potvrzené změny se odrazí v datech.
- Aktivita na pochybách se odráží v datech. Data jsou však uzamčena a nelze ji použít, dokud produkt IBM MQ nerozpozná a nebude jednat o rozhodnutí, které je na pochybách.
- Přerušené změny v letu a přerušené změny byly odebrány z front. Zprávy jsou konzistentní a lze je použít.
- Byl proveden nový kontrolní bod.
- Byly sestaveny nové indexy pro indexované fronty obsahující trvalé zprávy (popsané v části [“Přestavění indexů fronty”](#) na stránce 249).

Jsou-li používány duální sady BSDSP, produkt IBM MQ kontroluje konzistenci časových razítek v BSDS:

- Jsou-li obě kopie BSDS aktuální, produkt IBM MQ testuje, zda jsou dvě časová razítka shodná. Pokud tomu tak není, IBM MQ vydá zprávu CSQJ120E a skončí. To se může stát, když jsou dvě kopie BSDS udržovány na oddělených svazcích DASD a jeden z svazků byl obnoven, zatímco byl zastaven správce front. Produkt IBM MQ zjišťuje situaci při restartu.
- Je-li jedna kopie BSDS dealokovaná a protokolování pokračuje s jednou BSDS, může vzniknout problém. 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 obnovení detekovat. V takovém případě by záznamy žurnálu, které nejsou zaznamenány v BSDS, nebyly systému známy.

Dávkové aplikace se neoznamují, když se restart vyskytne *poté*, kdy aplikace požadovala připojení.

## Základní informace o rozsahu protokolu požadovaném pro zotavení

Během restartu je rozsah dat protokolu, který musí být přečten, závislý na mnoha faktorech:

- V době abnormálního ukončení je v systému obvykle mnoho neúplných jednotek práce. Jak již bylo popsáno dříve, opětovné spuštění zpracování přivede systém do stavu konzistence, který může zahrnovat zálohování informačních jednotek práce nebo obnovení zámků na neověřených jednotkách práce. Obnova jednotky práce vyžaduje, aby byly k dispozici všechny záznamy v protokolu pracovní jednotky pro podvrácení, odvrácení a pochybných jednotek práce. Produkt IBM MQ bude obsahovat staré pracovní jednotky, aby bylo možné provést zotavení 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 přečteny z protokolu a znovu použity na data uchovávané v sadách stránek. Objekty RBA obnovy stránky v kontrolním bodu popisují nejnižší protokol RBA, který je nezbytný pro aktualizaci sad stránek do konzistentního stavu.
- Pokud byly do systému zavedeny staré sady stránek, například byla zavedena záloha sady stránek pro zotavení ze selhání média, všechny změny musí být přečteny z protokolu od doby, kdy byla záloha provedena. Tyto změny se znovu aplikují na data, která se nacházejí v obnovované sadě stránek. Objekty RBA pro obnovení sady stránek na stránce 0 v sadě stránek popisují nejnižší hodnotu adresy RBA protokolu, která je vyžadována pro zotavení z médií sady stránek.



- Pokud se používají trvalé zprávy ve sdílených frontách, je požadován rozsah dat protokolu k obnovení CFSTRUCT, které zadržují trvalé zprávy. Nejdřívější data protokolu, která by byla požadována k provedení zotavení CFSTRUCT, jsou z přibližně času starého CFSTRUCT BACKUP.

Během normálního běhu lze použít příkaz DISPLAY USAGE TYPE (DATASET) k zobrazení rozsahu protokolu pro zotavení přidruženého k těmto faktorům (není schopen poskytovat informace kvůli opětovnému zavedení starých sad stránek). Chcete-li se vyhnout problémům, které by mohly prodloužit restartování správce front v případě nestandardního ukončení, pravidelně monitorujte výstup hodnot z parametru DISPLAY USAGE TYPE (DATASET).

Kromě toho správce front vydá informativní zprávy týkající se těchto faktorů:

- CSQJ160I a CSQJ161I varuje před dlouho provoznými jednotkami práce.
- CSQR026I a CSQR027I poskytují informace o tom, zda jsou tyto dlouho spuštěné jednotky práce úspěšně posuneny.
- CSQE040I a CSQE041E varují, že zálohování struktury jsou staré, a proto by operace RECOVER CFSTRUCT trvalo dlouhou dobu.

## Určení, která aplikace má přerušinou jednotku práce

Je možné určit aplikaci s dlouhotrvající jednotkou práce. Pro tento úkol použijte příkaz DISPLAY CONN.

Příkaz DISPLAY CONN vrací informace o připojení pro všechny aplikace připojené ke správci front společně s dalšími informacemi, které vám pomohou určit, které aplikace mají v současné době práci s dlouhou dobou zpracování. Informace vrácené příkazem DISPLAY CONN jsou podobné informacím vráceným příkazem DISPLAY QSTATUS, ale hlavní rozdíl je, že příkaz DISPLAY CONN zobrazuje informace o objektech a transakční informace pro konkrétní připojení, spíše než podrobnosti o tom, která připojení jsou přidružena k určitému objektu.

Pro každou připojenou aplikaci vrací příkaz DISPLAY CONN následující informace:

- Základní informace včetně ID připojení a PID.
- Transakční informace pro toto připojení, včetně času a data, kdy byla transakce vytvořena (tj. když byla první položka MQGET/PUT provedena pod synchronizačním bodem), a kdy byla transakce poprvé zapsána do protokolu.
- Informace o době protokolu označující, která aplikace má stále dlouhou spuštěnou jednotku práce.
- Seznam všech objektů, které má aktuálně otevřeno připojení. Podrobnosti pro každý objekt se vrátí jako samostatná zpráva, přičemž ID připojení se používá jako klíč. Vzhledem k tomu, že existují různé typy objektů, jako jsou například fronty a správci front, jsou informace zobrazované spolu s objektem specifické pro příslušný typ objektu.

## Přestavění indexů fronty

Chcete-li zvýšit rychlost operací MQGET ve frontě, kde se zprávy nenačítají sekvenčně, můžete určit, že chcete, aby produkt IBM MQ udržoval index zprávy nebo korelační identifikátory nebo ID skupiny pro všechny zprávy v této frontě.

Při restartování správce front jsou tyto indexy znovu sestaveny pro každou frontu. To platí pouze pro trvalé zprávy; přechodné zprávy se při restartu odstraní. Pokud vaše indexované fronty obsahují velký počet trvalých zpráv, zvýší se tím doba potřebná k restartování správce front.

Můžete zvolit automatické znovusestavení indexů do spuštění správce front pomocí parametru QINDXBLD makra CSQ6SYSP. Nastavíte-li QINDXBLD=NOWAIT, IBM MQ se restartuje bez čekání na nové sestavení indexů.

## Jak jsou vyřešeny neověřené jednotky zotavení

Pokud produkt IBM MQ ztratí připojení k jinému správci prostředků, obvykle se pokusí o obnovení všech nekonzistentních objektů při restartu.

Pokud produkt IBM MQ ztratí své připojení k produktu CICS, IMS nebo RRS, obvykle se pokusí o zotavení všech nekonzistentních objektů při restartu. Informace potřebné k vyřešení nejistých jednotek zotavení musí pocházet z koordinačního systému. Následující sekce popisují proces rozpoznání pro různá prostředí.

- [Jak neověřené jednotky zotavení se řeší z produktu CICS](#)
- [Jak neověřené jednotky zotavení se řeší z produktu IMS](#)
- [Jak neověřené jednotky zotavení jsou řešeny od služby RRS](#)
- [Jak jsou vyřešeny neověřené jednotky zotavení s dispozicí skupiny obnovení zotavení](#)

## **Jak jsou nejisté jednotky zotavení odstraněny z produktu CICS**

Za určitých okolností nemůže produkt CICS spustit proces IBM MQ k vyřešení nejistých jednotek zotavení. Pokud k tomu dojde, produkt IBM MQ odešle jednu z následujících zpráv:

- CSQC404E
- CSQC405E
- CSQC406E
- CSQC407E

následováno zprávou CSQC408I.

Podrobné informace o tom, co tyto zprávy znamenají, najdete v publikaci [IBM MQ for z/OS zprávy, dokončení, a kódy příčiny](#).

Řešení sporných jednotek neovlivňují prostředky produktu CICS. CICS je v řízení koordinace obnovy a při restartu automaticky potvrzuje nebo zálohuje každou jednotku, v závislosti na tom, zda existoval záznam protokolu označující začátek potvrzení. Existence objektů s nejistým stavem nezamyká CICS prostředky, zatímco IBM MQ je znovu připojen.

Jednou z funkcí adaptéru CICS je zachování synchronizace dat mezi CICS a IBM MQ. Pokud správce front dojde k nestandardům při připojení k produktu CICS, je možné, aby produkt CICS potvrdil nebo odvrátil práci, aniž by o tom produkt IBM MQ věděl. Po restartování správce front je tato práce označována jako *sporná*.

Produkt IBM MQ nemůže vyřešit tyto neověřené jednotky zotavení (to znamená potvrdit nebo vrátit změny provedené na prostředcích IBM MQ), dokud se připojení k produktu CICS nerestartuje nebo znovu nepřipojí.

Proces k vyřešení nejistých jednotek zotavení je zahájen během spuštění adaptéru CICS. Proces se spustí, když adaptér požádá o seznam nejistých jednotek zotavení. pak:

- Adaptér obdrží seznam nejistých jednotek zotavení pro toto ID připojení z produktu IBM MQ a předá je produktu CICS k vyřešení.
- CICS porovnává záznamy z tohoto seznamu s položkami ve vlastním protokolu. CICS určuje ze svého vlastního seznamu akcí, které akce podničila pro každou neověřenou jednotku zotavení.

U všech vyřešených jednotek IBM MQ aktualizuje fronty podle potřeby a uvolní odpovídající zámky. Nevyřešené jednotky mohou zůstat po restartu. Vyřešte je pomocí metod popsanych v příručce [Administrace IBM MQ for z/OS](#).

## **Jak jsou nejisté jednotky zotavení odstraněny z produktu IMS**

Řešení nejistých jednotek zotavení v produktu IMS neovlivňují prostředky DL/I. IMS má kontrolu nad koordinací obnovy a po restartu automaticky potvrdí nebo zazálohuje nekompletní DL/I práci. Rozhodnutí o potvrzení nebo vrácení pro online oblasti (ne-zkrácená cesta) je na přítomnost nebo nepřítomnost typů záznamů protokolu IMS X'3730 'a X'3801'. Existence neověřených jednotek zotavení neznamená, že záznamy DL/I jsou uzamknuty, dokud se IBM MQ nepřipojí.

Během restartu správce front produkt IBM MQ vytváří seznam nejistých jednotek zotavení. Příkaz IMS sestaví vlastní seznam zbytkových položek pro zotavení (RPE). RRE se protokolují na kontrolní body IMS , dokud se nevyřeší všechny záznamy.

Během opětovného připojení oblasti IMS k IBM MQ označuje IMS , zda má IBM MQ potvrdit nebo vrátit jednotky práce označené IBM MQ jako nejisté.

Jsou-li v nejistém stavu jednotky vyřešeny:

1. Pokud produkt IBM MQ rozpozná, že označil položku pro potvrzení a IMS ji označil, aby byla vrácena, IBM MQ vydá zprávu CSQQ010E. IBM MQ vydá tuto zprávu pro všechny nekonzistence tohoto typu mezi IBM MQ a IMS.
2. Má-li produkt IBM MQ nějaké zbývající nejisté jednotky, adaptér vydá zprávu CSQQ008I.

U všech vyřešených jednotek IBM MQ aktualizuje fronty podle potřeby a uvolní odpovídající zámky.

Produkt IBM MQ udržuje zámky na neověřené práci, která nebyla vyřešena. To může způsobit, že se v systému nahromadí nevyřízené zámky, pokud se budou zdržovat důležité zámky. Připojení zůstává aktivní, takže můžete přeložit IMS RRE. Proveďte zotavení neověřených podprocesů pomocí metod popsanych v příručce [Administrace IBM MQ for z/OS](#).

Všechny neověřené práce by měly být vyřešeny, pokud neexistují softwarové nebo provozní problémy, jako například se studeným startem IMS . Nejisté rozlišení řídicí oblasti IMS se provádí za dvou okolností:

1. Na začátku připojení k produktu IBM MQ, během kterého je rozpoznání prováděno synchronně.
2. Když se program neukončí, během kterého je rozlišení provedeno asynchronně.

## Jak jsou v nejistých jednotkách zotavení odstraněny z RRS

Jednou z funkcí adaptéru RRS je udržovat data synchronizovaná mezi produktem IBM MQ a dalšími správci prostředků RRS. Pokud dojde k selhání, když produkt IBM MQ dokončí fázi jedné operace commit a čeká na rozhodnutí RRS (koordinátor potvrzení), bude jednotka zotavení přejde do stavu nejistoty.

Je-li komunikace znovu zavedena mezi RRS a IBM MQ, služba RRS automaticky potvrdí nebo zazálohuje 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 produktu IBM MQ ), dokud nebude znovu ustanoveno připojení k RRS.

Za určitých okolností nemůže služba RRS vyřešit jednotky zotavení s nejistým stavem. Pokud k tomu dojde, produkt IBM MQ odešle na konzolu z/OS jednu z následujících zpráv:

- CSQ3011I
- CSQ3013I
- CSQ3014I
- CSQ3016I

Podrobné informace o tom, co tyto zprávy znamenají, najdete v publikaci [IBM MQ for z/OS zprávy, dokončení, a kódy příčiny](#) .

Pro všechny vyřešené jednotky zotavení produkt IBM MQ aktualizuje fronty podle potřeby a uvolní odpovídající zámky. Nevyřešené jednotky zotavení mohou zůstat po restartu. Vyřešte je pomocí metody popsané v příručce [Administrace IBM MQ for z/OS](#).

## Jak jsou vyřešeny neověřené jednotky zotavení s dispozicí skupiny obnovení zotavení

Neověřené transakce, které mají dispozici GROUP jednotky zotavení, může být vyřešeno koordinátorem transakce libovolného správce front v rámci skupiny sdílení front (QSG), kde je povolen atribut správce front GROUPPUR. Kdykoli se koordinátor transakcí znovu připojí, obvykle požádá o seznam všech neprovedených neověřených transakcí a poté je vyřeší pomocí informací z protokolů.

Pokud koordinátor transakcí, který je připojen k dispoziční jednotce skupiny obnovy, požádá o seznam sporných transakcí, vrácený seznam obsahuje všechny neověřené transakce se skupinou sdílení zotavení, které existují v rámci skupiny sdílení front. Tento seznam není závislý na tom, který správce front tyto neověřené transakce spustily. Správce front zpracovává tento požadavek zkompile seznam tím, že komunikuje se všemi ostatními aktivními správci front ve skupině sdílení front za použití systému SYSTEM.QSG.UR.RESOLUTION.QUEUE. Správce front poté načte protokoly všech neaktivních správců front z jejich posledního kontrolního bodu, aby identifikoval všechny další neověřené transakce, které by byly hlášeny, kdyby byly aktivní.

Pokud koordinátor transakcí požádá o vyřešení transakce s nejistým stavem, identifikuje správce front, k němuž je připojen, identifikuje, zda byla transakce vytvořena sama o sobě, a pokud ji tak řeší, stejně jako transakce s dispozíci QMGR zotavení. Pokud byla transakce vytvořena v jiném aktivním správci front v QSG, je požadavek na dokončení vyřešení přeměrován do tohoto správce front pomocí SYSTEM.QSG.UR.RESOLUTION.QUEUE. V případě, že transakce pocházela z neaktivního správce front v QSG, je veškerá práce ve sdílené frontě vyřešena okamžitě a požadavek na vyřešení jakékoli zbývající práce soukromé fronty je umístěn na SYSTEM.QSG.UR.RESOLUTION.QUEUE. Neaktivní správce front zpracovává tento požadavek po spuštění před přijetím nové práce. V tomto scénáři protokoly původního správce front stále odrážejí, že jednotka zotavení je nejistá, dokud nerestartuje a nezpracovává požadavek.

## **Obnova sdílené fronty**

V tomto tématu se rozumí obnova a odolnost různých komponent v prostředí skupiny sdílení front produktu IBM MQ .

- [“Transakční obnova” na stránce 252](#)
- [“Obnova partnerských serverů” na stránce 252](#)
- [“Definice sdílených front” na stránce 253](#)
- [“Protokolování” na stránce 253](#)
- [“Selhání prostředku Coupling Facility a struktury” na stránce 253](#)
- [“Scénáře poruchy struktury” na stránce 254](#)
- [“Odolnost proti selhání konektivity prostředku Coupling Facility” na stránce 255](#)
- [“Správa odolnosti vůči selháním konektivity prostředku CF” na stránce 256](#)
- [“Operační chování” na stránce 258](#)

## **Transakční obnova**

Když aplikace vydá volání MQBACK nebo je nestandardně ukončen (například kvůli informacím EXEC CICS ROLLBACK nebo IMS abend), které jsou uloženy ve správci front, zajistíte, aby byla odletová jednotka práce odvrácena. Operace MQPUT a MQGET v rámci synchronizačního bodu ve sdílených frontách jsou odvolány stejným způsobem jako aktualizace nesdílených front.

## **Obnova partnerských serverů**

Pokud správce front selže, odpojí se od struktur prostředku Coupling Facility, ke kterým je momentálně připojen. Dojde-li k selhání připojení mezi instancí produktu z/OS a prostředku Coupling Facility (například selhání fyzického spoje nebo vypnutí prostředku CF nebo oblast), je toto připojení zjištěno také jako nestandardní ukončení připojení mezi správcem front a zapojenými strukturami zařízení Coupling Facility. Ostatní správci front ve stejné skupině sdílení front, které zůstávají připojeny k této struktuře, detekují nestandardní odpojení a všechny pokusy o zahájení *partnerského zotavení* pro správce front, který selhal, v této struktuře. Pouze jeden z těchto správců front iniciuje zotavení typu peer úspěšně, ale všechny ostatní správce front spolupracují při obnově transakcí, které vlastní správce front, který selhal.

Pokud správce front selže, když nejsou k žádné struktuře připojeny žádné rovnocenné uzly, zotavení se provede, když se k této struktuře připojí jiný správce front nebo když se správce front, který selhal, restartuje.

Zotavení typu peer, často označované jako PLR (Peer Level Recovery), se provádí na základě struktury podle struktury struktury a je možné, aby se jeden správce front podílel na zotavení více než jedné struktury současně. Avšak sada rovnocenných uzlů spolupracujících při obnově různých struktur se může lišit v závislosti na tom, kteří správci front byli připojeni k různým strukturám v době selhání.

Po restartování správce front se selháním se znovu připojí ke strukturám, ke kterým byl připojen v době selhání, a obnoví všechny zbývající nevyřešené jednotky práce, které nebyly zotaveny pomocí partnerského zotavení.

Zotavení typu peer je vícefázový proces. Během první fáze se zotavují jednotky práce, které postupovaly mimo fázi letu; to může zahrnovat potvrzení zpráv pro jednotky práce, které jsou ve stavu odevzdání a zamykání pro jednotky práce, které mají pochybnosti. Během druhé fáze jsou zkontrolovány fronty, které měly aktivní podprocesy v selhávajícím správcí front, nepotvrzené zprávy související s odletem práce v rámci pracovní jednotky a informace o aktivních manipulátorech ve sdílených frontách ve správcí front, který selhal, jsou obnoveny. To znamená, že produkt IBM MQ resetuje všechny indikátory, které má selhávající správce front sdílenou frontu otevřenou pro vstup, což umožňuje jiným aktivním správcům front otevřít frontu pro vstup.

## Definice sdílených front

Objekty fronty, které reprezentují atributy sdílené fronty, se nacházejí ve sdíleném úložišti Db2 používaném skupinou sdílení front. Ujistěte se, že jsou zavedeny adekvátní procedury pro zálohování a obnovu tabulek Db2 používaných k zadržení objektů IBM MQ. Obslužný program IBM MQ CSQUTIL můžete také použít k vytvoření příkazů MQSC pro přehrání do správce front za účelem předefinování objektů IBM MQ, včetně definic sdílené fronty a skupin uložených v produktu Db2.

## Protokolování

Skupiny sdílení front mohou podporovat trvalé zprávy, protože zprávy ve sdílených frontách mohou být protokolovány v protokolech správce front.

## Selhání prostředku Coupling Facility a struktury

Existují dva typy selhání, které mohou být hlášeny pro strukturu prostředku CF (Coupling Facility): selhání struktury a ztráta konektivity. Služby prostředí sysplex pro sdílení dat (XES) informují IBM MQ o selhání struktury prostředku CF nebo o selhání prostředku CF s událostí selhání struktury. Pokud XES vytvoří ztrátu připojitelnosti události, nemusí to nutně znamenat, že se vyskytl problém se strukturou, může to být, že není k dispozici žádné připojení pro komunikaci se strukturou. Je možné, že ne všichni správci front obdrží ztrátu připojitelnosti události pro strukturu; závisí na konfiguraci připojení k prostředku CF. Událost ztráty konektivity může být také přijata kvůli příkazům operátora, například VARY PATH OFFLINE nebo CONFIG CHP OFFLINE.

Struktury prostředku mezipaměti, které používá produkt IBM MQ, lze konfigurovat tak, aby používaly oboustranný tisk spravovaného systémem. To znamená, že pokud dojde k jednomu selhání, zpracování překonání selhání spravované systémem skryje selhání struktury nebo ztrátu konektivity a správce front není informován o selhání. Dojde-li k selhání obou instancí duplexní struktury nebo připojení, obdrží správce front příslušnou událost a zpracuje ji stejným způsobem jako událost selhání pro strukturu simplex. Podrobnosti o tom, jak správce front zpracovává události, je popsán ve [Scénáře](#).

Dojde-li k selhání prostředku CF nebo selhání struktury, budou ztraceny všechny přechodné zprávy uložené v ovlivněných strukturách aplikace. Trvalé zprávy můžete obnovit pomocí příkazu RECOVER CFSTRUCT. Pokud se zotavitelná struktura aplikace nezdařila, je zabráněno jakékoli další aktivitě aplikace této struktuře, dokud se struktura neobnoví.

Chcete-li zajistit obnovení struktury prostředku CF v přiměřeném časovém intervalu, proveďte časté zálohy pomocí příkazu BACKUP CFSTRUCT. Můžete zvolit provedení záloh u všech správců front ve skupině sdílení front nebo vyhradit jednoho správce front, aby provedl všechny zálohy. Automatizujte proces vytváření záloh, abyste zajistili, že se budou provádět pravidelně.

Každá záloha je zapsána do aktivní datové sady protokolu správce front, který provádí zálohování. Úložiště sdílené fronty Db2 zaznamenává název zálohované struktury prostředku CF, název správce front, který provádí zálohu, rozsah RBA pro tuto zálohu v protokolu správce front a čas zálohování.

Struktura administrace obsahuje informace o nekompletních jednotkách práce ve sdílených frontách v době selhání struktury aplikace, takže struktura administrace musí být k dispozici během zpracování RECOVER CFSTRUCT. Pokud se struktura administrace nezdařila, všichni správci front ve skupině sdílení front musí před vydáním příkazu RECOVER CFSTRUCT znovu sestavit své položky struktury administrace.

Správci front automaticky a bez ukončení znovu sestaví své položky administrativní struktury. Není-li správce front v okamžiku selhání spuštěn, lze jeho položky struktury administrace znovu sestavit jiným správcem front v rámci skupiny sdílení front, která je spuštěna na stejné nebo vyšší úrovni.

Chcete-li obnovit strukturu aplikace, zadejte příkaz RECOVER CFSTRUCT na správce front, kterého chcete provést pro zotavení. Můžete obnovit jednu strukturu CF, nebo můžete obnovit několik struktur prostředku CF současně. Můžete obnovit použití libovolného správce front ve skupině sdílení front, nemusí to být ten, který provedl zálohu, nebo který byl dříve připojen ke struktuře, která se nezdařila.

Příkaz RECOVER CFSTRUCT používá zálohování umístěné prostřednictvím informací o úložišti produktu Db2 ( Db2 musí být proto k dispozici ve správci front, ve kterém probíhá obnova), a obnoví tuto zálohu do bodu selhání.

Příkaz RECOVER CFSTRUCT provede použití záznamů protokolu ze všech správců front ve skupině sdílení front, která provedla operaci MQPUT nebo MQGET mezi začátkem zálohování a časem selhání, do jakékoli sdílené fronty, která se mapuje na strukturu CF. Výsledná sloučení protokolů může vyžadovat přečtení značného objemu dat protokolu, protože všechna data protokolu zapsaná zúčastněnými správci front od doby, kdy je záloha čtena. Důrazně se doporučuje provádět časté (například hodinové) zálohy, zvláště pokud jsou v rámci zálohy velké zprávy.

## Scénáře poruchy struktury

### Scénáře

Je-li pro strukturu prostředku CF hlášena porucha, závisí akce provedená připojenými správci front na následujícím:

- Typ selhání nahlášený komponentou XES produktu z/OS na IBM MQ.
- Typ struktury (aplikace nebo administrace)
- Úroveň správce front
- CFLEVEL objektu IBM MQ CFSTRUCT (2, 3, 4 nebo 5). Toto není CFLEVEL mikrokódu CFCC)
- Atribut RECERAO objektu IBM MQ CFSTRUCT na úrovni CFLEVEL (5)

Následující scénáře popisují, co se stane, když se hlášení o struktuře administrace hlásí:

- Je-li pro strukturu administrace přijata událost selhání struktury, struktura se znovu přidělí a znovu sestaví automaticky, aniž by se správce front ukončil. Nová instance struktury je alokována XES, když se správce front pokusí o připojení k serveru.

Pokud se správce front připojí k nové instanci struktury, správce front запиše položky do struktury. Toto zpracování je prováděno správcem front a není součástí zpracování opětovného sestavení XES.

Pokud správce front nebyl v době selhání spuštěn nebo je ukončen před dokončením zotavení její části administrativní struktury, lze její položky struktury administrace znovu sestavit jiným správcem front ve skupině sdílení front.

Záznamy struktury administrace správce front lze znovu sestavit pouze jiným správcem front spuštěným na stejné úrovni nebo vyšším. Pokud správci front v rámci skupiny sdílení front nemůže

znovu vytvořit položky struktury správy správce front, restartujte správce front, aby mohl dokončit znovusestavení její části struktury.

Některé akce jsou pozastaveny, dokud nejsou znovu sestaveny položky struktury administrace pro všechny správce front. Pozastavené akce zahrnují následující:

- Otevření a zavření sdílených front.
- Potvrzení nebo vrácení jednotek zotavení.
- Serializované aplikace, které se připojují ke správci front nebo odpojují od správce front.
- Zálohování nebo obnova struktury aplikace.

Všechny serializované aplikace, které jsou již připojeny ke správci front, mohou pokračovat ve zpracování. Každá serializovaná aplikace pokoušející se o připojení k parametrům MQCNO\_SERIALIZE\_CONN\_TAG\_QSG nebo MQCNO\_RESTRICT\_CONN\_TAG\_QSG obdrží návratový kód MQRC\_CONN\_TAG\_NOT\_USABLE.

Po opětném sestavení položek administrativní struktury pro správce front dojde k obnovení pozastavených akcí.

Následující scénáře popisují, co se stane, když se hlášení o struktuře aplikace hlásí pro:

- Je-li pro strukturu aplikace přijata událost poruchy struktury a hodnota CFLEVEL je 1 nebo 2, správce front se ukončí. Restartujte správce front. První správce front, který se pokusí o připojení ke struktuře, znovu způsobí, že XES přidělí novou instanci struktury.
- Je-li pro strukturu aplikace přijata událost poruchy struktury a verze CFLEVEL je 3, 4 nebo 5, správci front připojené ke struktuře budou pokračovat ve zpracování. Aplikace, které nepoužívají fronty v neúspěšné struktuře, mohou pokračovat v normálním zpracování.

Aplikace, které se pokoušejí o operace ve frontách ve vadném stavu, však obdrží chybu MQRC\_CF\_STRUC\_FAILED, dokud nebude struktura úspěšně znovu sestavena, v tom okamžiku může aplikace znovu otevřít fronty.

Přestavění struktury se spustí automaticky pro aplikační struktury CFLEVEL (5) definované s parametrem RECASTO (YES). Jinak bude struktura znovu sestavena, až bude vydán příkaz RECOVER CFSTRUCT.

## **Odolnost proti selhání konektivity prostředku Coupling Facility**

### **Co je odolnost vůči selháním konektivity prostředku CF?**

Odolnost proti selhání konektivity prostředku odkazuje na schopnost správců front ve skupině sdílení front tolerovat ztrátu připojení ke struktuře prostředku Coupling Facility, aniž by došlo k ukončení. Tato funkce se také pokouší znovu sestavit strukturu v jiném propojovacím zařízení s lepší konektivitou, aby bylo možné co nejdříve znovu získat přístup ke sdíleným frontám.

### **Co je částečná ztráta konektivity?**

IBM MQ definuje částečnou ztrátu připojitelnosti jako situaci, kdy jeden nebo více systémů v prostředí sysplex ztratí připojení k prostředku CF, kde je přidělena struktura, ke které je systém přístupován, ale alespoň jeden systém v prostředí sysplex udržuje spojení se stejným prostředkem pro spojení.

### **Jaká je celková ztráta konektivity?**

IBM MQ definuje celkovou ztrátu konektivity jako situace, kdy žádné systémy v prostředí sysplex nemají připojitelnost k prostředku Coupling Facility a strukturování, které je v něm alokováno.

### **Proč jste povolili tuto funkci?**

Možnost odolnosti vůči selháním konektivity prostředku zvyšuje dostupnost produktu IBM MQa umožňuje, aby nesdílené fronty zůstaly dostupné poté, co správce front ztratil připojení k jedné nebo více strukturám prostředku Coupling Facility. Kromě toho správci front, kteří ztratili spojení se strukturou prostředku Coupling Facility, se automaticky pokusí znovu sestavit strukturu v jiném dostupném prostředku CF, čímž se zlepší dostupnost sdílených front v rámci skupiny sdílení front.

## Aspekty povolení této funkce

Správce front, který toleruje ztrátu připojení ke strukturám prostředku Coupling Facility bez ukončení, nemusí být schopen znovu navázat spojení se strukturou prostředku Coupling Facility, pokud již není k dispozici alternativní zařízení pro spojení. Sdílené fronty definované ve struktuře, která utrpěla ztrátu konektivity, zůstávají nedostupné, dokud nebude obnovena konektivita struktury. V této situaci mohou aplikace, které se připojují ke členům skupiny sdílení front za účelem provedení práce sdílené fronty, zjistit, že sdílené fronty, které potřebují k přístupu, nejsou k dispozici. Chcete-li se této situaci vyhnout, doporučujeme, aby byly správci front konfigurovány k ukončení při ztrátě připojení ke struktuře prostředku Coupling Facility. Toto ukončení aplikace vynutí připojení k jinému členu skupiny sdílení front, který má konektivitu ke strukturám prostředku Coupling Facility, v němž jsou definovány sdílené fronty, které aplikace vyžaduje.

## Správa odolnosti vůči selháním konektivity prostředku CF

### Jak lze tuto funkci povolit?

Následující kroky je třeba provést, chcete-li povolit odolnost vůči konektivě prostředku Coupling Facility

1. Ujistěte se, že datová sada CFRM byla formátována tak, aby podporovala znovusestavení spravované systémem. Díky tomu mohou správci front zahájit nové sestavení systému znovu pro opětovné vytvoření struktury v rámci dostupného prostředku CF. K určení formátu datové sady CFRM se používá příkaz **DISPLAY XCF, COUPLE, TYPE=CFRM**. Chcete-li podporovat opětovné sestavení spravované systémem, měla by být dvojice datových sad CFRM formátována zadáním následujícího příkazu:

```
"ITEM NAME(SMREBLD) NUMBER(1) "
```

Další informace o formátování datové sady CFRM najdete v dokumentaci k produktu [z/OS MVS Setting Up a Sysplex](#).

2. Ujistěte se, že je k dispozici alternativní prostředek Coupling Facility a že je v seznamu předvoleb CFRM pro všechny struktury prostředku Coupling Facility IBM MQ. Správci front se tak mohou pokusit o nové sestavení struktur do alternativního dostupného prostředku CF, aby byl obnoven přístup ke strukturám, jakmile to bude možné.

Struktury IBM MQ musí být definovány s ENFORCEORDER (NO) v zásadě CFRM, aby mohl být XCF schopen zvolit optimální prostředek CF v konfiguraci, pokud IBM MQ potřebuje znovu alokovat strukturu.

Další informace o seznamech předvoleb struktury naleznete v dokumentaci produktu [z/OS MVS Setting a Sysplex](#).

3. Pozměnit všechny struktury prostředku Coupling Facility, které potřebují tolerovat ztrátu připojení k kumulativní opravě CFLEVEL (5). To je minimální úroveň, která může tolerovat ztrátu konektivity.
4. Určete hodnoty požadované pro atributy **QMGR CFCONLOS** a **CFSTRUCT CFCONLOS** a změňte je odpovídajícím způsobem. Atribut **QMGR CFCONLOS** řídí, zda je tolerance konektivity ke struktuře administrace tolerována, a atribut **CFSTRUCT CFCONLOS** řídí, zda je ztráta konektivity tolerována každou strukturou prostředku Coupling Facility. Jsou-li zachované výchozí hodnoty těchto atributů, správce front bude po ztrátě připojení k libovolné struktuře prostředku Coupling Facility ukončen.
5. Určete hodnoty vyžadované pro atribut **CFSTRUCT RECAUTO** pro každou strukturu prostředku Coupling Facility, a odpovídajícím způsobem změňte tyto hodnoty. 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 použita výchozí hodnota, nebude provedena automatická obnova pro struktury aplikací po úplném výpadku konektivity.

### Scénář 1-Ztráta konektivity ke struktuře administrace

Správci front mohou tolerovat ztrátu připojení ke struktuře administrace bez ukončení, pokud jsou všichni správci front ve skupině sdílení front v produktu IBM WebSphere MQ 7.1 nebo novější. Pokud jsou správci front na úrovni nižší než IBM WebSphere MQ 7.1 ve skupině sdílení front, budou všichni



správci front v rámci skupiny sdílení front s kódem příčiny 00C510AB při ztrátě připojení ke struktuře administrace ztraceny.

Dojde-li ke ztrátě připojení ke struktuře administrace pomocí libovolného správce front, který byl nakonfigurován tak, aby toleroval ztrátu připojení ke struktuře administrace, dojde k odpojení všech členů skupiny sdílení front ze struktury administrace. Všichni aktivní správci front ve skupině sdílení front se poté pokusí znovu připojit ke struktuře administrace, což způsobí, že bude znovu přidělen ve spojovacím zařízení s nejlepší možností připojení ke všem systémům v prostředí sysplex, a znovu sestavit data struktury administrace.

**Poznámka:** To nemusí být nutně spojovací prostředek, který má nejlepší připojitelnost ke všem systémům, které mají aktivní správce front.

Pokud se správce front nemůže připojit ke struktuře administrace, například protože není k dispozici žádná ze spojovacích prostředků v seznamu předvoleb CFRM pro strukturu administrace, některé operace sdílené fronty zůstanou nedostupné, dokud se správce front nebude moci úspěšně znovu připojit ke struktuře administrace a znovu sestavit data struktury administrace. Opětovné připojení nastane automaticky, když je v systému k dispozici vhodná spojovací zařízení.

Selhání při pokusu o připojení ke struktuře administrace během spuštění správce front v důsledku nedostatku konektivity k prostředku CF nebo k dispozici není tolerováno žádné vhodné zařízení pro spojení, které by bylo možné přidělit této struktuře. Všichni aktivní správci front ve skupině sdílení front se poté pokusí znovu připojit ke struktuře administrace, což způsobí, že je znovu přidělen v jiném prostředku CF, je-li k dispozici, a znovu sestavit data struktury administrace.

## Scénář 2-Ztráta konektivity k aplikační struktuře

Ztráta konektivity k aplikačním strukturám na serveru **CFLEVEL (5)** nebo vyšší může být tolerována, aniž by se správce front ukončil. Správci front jsou připojeni ke strukturám aplikací na serveru **CFLEVEL (4)** nebo nižším nebo strukturám v produktu **CFLEVEL (5)**, které nebyly nakonfigurovány tak, aby tolerovaly ztrátu konektivity, přičemž došlo ke ztrátě spojení s kódem příčiny 00C510AB, když došlo ke ztrátě konektivity ke struktuře.

Dojde-li ke ztrátě připojení k aplikační struktuře, která byla konfigurována tak, aby tolerovala ztrátu konektivity, všichni správci front, kteří ztratili spojení se strukturou, se odpojí. Následné chování správce front závisí na tom, zda je ztráta konektivity částečná nebo úplná.

### Částečná ztráta konektivity k aplikační struktuře

Je-li ztráta konektivity určena jako částečná, správci front, kteří ztratili konektivitu ke struktuře, se pokusí o zahájení opětovného sestavení spravovaného systémem, aby bylo možné přesunout strukturu do jiného prostředku CF se zvýšenou konektivitou. Je-li toto znovusestavení úspěšné, budou jak trvalé tak netrvalé zprávy ve struktuře zkopírovány do jiného prostředku Coupling Facility, a obnoví se přístup k frontám ve struktuře. Správci front, kteří neztratili připojitelnost, zůstávají připojeni ke struktuře, avšak operace, které mají přístup ke struktuře, jsou zpožděny během procesu nového sestavení spravovaného systémem.

Pokud nelze strukturu aplikace znovu sestavit na jiné spojivé zařízení se zlepšenou konektivitou, nebo někteří správci front dosud nemají připojitelnost ke struktuře po její opětovné sestavení v jiném prostředku CF, fronty definované ve struktuře zůstanou nedostupné pro správce front, kteří nemají připojitelnost k této struktuře, dokud nebude připojení obnoveno do zařízení pro spojení. Správci front se znovu automaticky připojí ke struktuře, jakmile jsou 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 ztratili připojení k prostředku CF, který je alokován aplikační strukturou, produkt z/OS dealokuje strukturu z prostředku CF, kdykoli je proveden pokus o nové připojení ke struktuře. Je možné, že správce front se pokusí znovu připojit ke struktuře z několika důvodů, jako je například pokus aplikace o otevření sdílené fronty nebo upozornění ze systému, že se nové prostředky prostředku Coupling Facility mohou stát dostupnými. Je proto pravděpodobné, že všechny netrvalé zprávy v ovlivněné struktuře budou ztraceny po úplné ztrátě připojení ke struktuře aplikace.

Obnovitelné aplikační struktury jsou automaticky obnoveny po úplném výpadku konektivity, pokud byly definovány s **RECAUTO (YES)**. Zotavení se spustí téměř okamžitě, je-li k dispozici alternativní zařízení pro spojení, které má danou strukturu přidělit, nebo kdykoli je takové zařízení pro spojení s možností připojení k dispozici. Pokud nebyla struktura definována pomocí produktu **RECAUTO (YES)**, lze zotavení spustit zadáním příkazu **RECOVER CFSTRUCT**. Tím se obnoví všechny trvalé zprávy ve struktuře, ale všechny netrvalé zprávy se ztratí. Vzhledem k tomu, že tento proces zahrnuje čtení protokolu správce front, může dokončení některých času trvat, proto se doporučuje provádět pravidelné zálohování struktury, aby se zkrátí doba, dokud nebude obnoven přístup ke sdíleným frontám struktury.

Správci front se znovu pokusí znovu připojit k neobnovitelným strukturám aplikace, jakmile se aplikace pokusí otevřít sdílenou frontu, která je definována ve struktuře nebo obdrží oznámení ze systému, že se nové prostředky prostředku Coupling Facility staly dostupnými. Je-li k dispozici vhodné spojovací zařízení pro přidělení struktury, je přidělena nová struktura a přístup ke sdíleným frontám definovaným ve struktuře je obnoven. Vzhledem k tomu, že trvalé zprávy nelze vkládat do front definovaných v neobnovitelných strukturách, všechny zprávy ve sdílených frontách budou ztraceny.

## Operační chování

Je-li správce front IBM WebSphere MQ 7.1 nebo novější konfigurovaný tak, aby toleroval ztrátu připojení ke konkrétní struktuře prostředku Coupling Facility, ztratí připojení, členové skupiny sdílení front se pokusí o automatické zotavení ze selhání a znovu se připojit ke struktuře. Tato aktivita může zahrnovat přerozdělení struktury v jiném propojovacím zařízení s lepší konektivitou, je-li k dispozici. Avšak intervence operátora může být i nadále vyžadována pro zotavení ze ztráty konektivity.

Nezbytná akce operátora je obvykle následující:

1. Vyřešte příčinu selhání, které vedlo ke ztrátě konektivity.
2. Ujistěte se, že zařízení CF, kde lze přidělit struktury IBM MQ, je k dispozici na všech systémech v prostředí sysplex

Všechny struktury, které byly automaticky znovu přiděleny v jiném prostředku Coupling Facility po přerušení události připojení, lze přesunout do prostředku Coupling Facility s optimální možností připojení ke všem správcům front v rámci skupiny sdílení front. Je-li to nutné, lze to provést inicializací příkazu nového sestavení spravovaného systémem **SETXCF START, REBUILD**, jak je dokumentováno v publikaci [z/OS MVS Popis příkazů systému](#).

V případě částečné ztráty připojení ke struktuře aplikací se správci front, kteří ztratili konektivitu ke struktuře, pokusí o inicializaci znovusestavení spravovaného systémem. Tento proces alokuje strukturu pouze v jiném prostředku Coupling Facility, pokud má toto zařízení pro spojení možnost připojení ke všem aktivním správcům front, kteří jsou momentálně připojeni ke struktuře. Proto je možné, že pokud většina správců front ve skupině sdílení front ztratila připojitelnost ke struktuře aplikace, nemohou znovu sestavit strukturu do jiného prostředku CF vzhledem ke správcům front, kteří jsou stále připojeni k původní struktuře. V této situaci mohou být správci front, kteří jsou stále připojeni k původní struktuře, buď zticha, aby bylo možné strukturu znovu sestavit, nebo příkaz **RESET CFSTRUCT ACTION(FAIL)** může být vydán, aby selhal ve struktuře. Zotavení lze zahájit na příslušných strukturách zadáním příkazu **RECOVER CFSTRUCT**.

**Poznámka:** Když dojde k selhání a obnově struktury, všechny netrvalé zprávy ve struktuře se ztratí.

z/OS

## Koncepce zabezpečení v produktu IBM MQ for z/OS

Toto téma slouží k pochopení důležitosti zabezpečení pro produkt IBM MQ a o důsledcích neodpovídajícího nastavení zabezpečení v systému.

### Proč je třeba chránit prostředky produktu IBM MQ

Produkt IBM MQ zpracovává přenos informací, které jsou potenciálně cenné. Použití zabezpečení zajišťuje, že prostředky IBM MQ vlastní a spravované jsou chráněny před neoprávněným přístupem. Takový přístup by mohl vést ke ztrátě nebo zveřejnění informací.

Měli byste se ujistit, že žádný z následujících prostředků není přístupný nebo změněn neautorizovaným uživatelem nebo procesem:

- Připojení k produktu IBM MQ
- IBM MQ objektů, jako jsou fronty, procesy a seznamy názvů
- IBM MQ odkazy přenosu, tj. IBM MQ kanály
- IBM MQ řídicí příkazy systému
- Zprávy produktu IBM MQ
- kontextové informace přidružené ke zprávám

Chcete-li zajistit potřebné zabezpečení, produkt IBM MQ použije prostředek autorizace SAF (System Authorization Facility) systému z/OS pro směrování požadavků na autorizaci k externímu správci zabezpečení (ESM), například k produktu Security Server (dříve známý jako RACF). Produkt IBM MQ neprovádí žádné ověření zabezpečení svých vlastních. V případě použití distribuovaných front nebo klientů můžete vyžadovat další bezpečnostní opatření, pro která produkt IBM MQ poskytuje záznamy ověřování kanálu, uživatelské procedury kanálu, atribut kanálu MCAUSER a TLS.

Rozhodnutí o povolení přístupu k objektu se provádí pomocí ESM a IBM MQ následuje toto rozhodnutí. Pokud ESM nemůže učinit rozhodnutí, IBM MQ zabrání v přístupu k objektu.

## Co se stane, pokud neochráníte prostředky produktu IBM MQ

Pokud nic o zabezpečení neuděláte, je nejpravděpodobnější příčinou to, že *všichni* uživatelé mohou přistupovat a měnit *každý* prostředek. To zahrnuje nejen lokální uživatele, ale i ty na vzdálených systémech používajících distribuované fronty nebo klienty, kde kontroly zabezpečení přihlášení mohou být méně přísná než normálně v případě z/OS.


Chcete-li povolit kontrolu zabezpečení, musíte provést následující:

- Instalujte a aktivujte ESM (například Security Server).
- Pokud používáte jiný typ ESM než server zabezpečení, definujte třídu MQADMIN.
- Aktivace třídy MQADMIN.

Je třeba zvážit, zda použití názvů prostředků se smíšenými malými a velkými písmeny bude výhodné pro váš podnik. Pokud ve svých profilech ESM používáte názvy prostředků se smíšenými malými a velkými písmeny, musíte definovat a aktivovat třídu MXADMIN.

## z/OS Šifrování datové sady

Modul DSE (Data Set Encryption) poskytuje schopnost šifrovat datové sady produktu z/OS tak, aby data, která obsahují, mohla zobrazit nebo upravit pouze podle ID uživatelů, kterým bylo uděleno specifické oprávnění. To poskytuje šifrování dat v ostatních systémech souborů a zabraňuje neúmyslnému prozrazení citlivých informací uživatelům, kteří mají legitimní obchodní potřebu a oprávnění ke správě samotných datových sad.

 Před verzí IBM MQ for z/OS 9.1.4 nepodporuje produkt IBM MQ for z/OS použití modulu DSE s aktivními protokoly, sadami stránek a sdílenými datovými sadami zpráv (SMDS), které poskytují mechanismy primární perzistence pro zprávy produktu IBM MQ .

Místo toho produkt Advanced Message Security poskytuje řešení pro šifrování koncových uživatelů pro systém zpráv produktu IBM MQ , které zahrnuje celou síť produktu IBM MQ , šifrování dat za letu, v klidu a dokonce i uvnitř běhového prostředí produktu IBM MQ .

Ostatní datové sady VSAM a sekvenční datové sady použité v subsystému IBM MQ mohou být šifrovány pomocí DSE. Příklad:

- zaváděcí datová sada
- Sekvenční soubory obsahující příkazy konfigurace systému (MQSC) načtené při spuštění pomocí CSQINPx DDSNAMEs

- IBM MQ archivní protokoly, často používané pro dlouhodobé archivace dat protokolu produktu IBM MQ pro účely auditu.

Pomocí DSE můžete šifrovat pomocí přidělení datové třídy, která je definována s popisem klíče datové sady. Další informace najdete v tématu [Plánování archivního úložiště protokolu](#).

**V 9.2.0** Počínaje produktem IBM MQ for z/OS 9.1.4 podporuje produkt IBM MQ for z/OS kromě podpory poskytované v dřívějších vydáních také použití modulu DSE s aktivními protokoly a sadami stránek.

**V 9.2.0** Produkt IBM MQ for z/OS nepodporuje použití DSE pro sdílené datové sady zpráv (SMDS).

**V 9.2.0** Informace naleznete v části [Důvěrnost dat v produktu IBM MQ for z/OS s šifrováním datové sady](#). Další informace viz.

### **Související pojmy**

[Koncepce zabezpečení](#)

[Záznamy ověření kanálu](#)

[Oprávnění pro práci s objekty IBM MQ v systému z/OS](#)

[Kryptografické bezpečnostní protokoly: TLS](#)

[Zdroje, ze kterých můžete zadávat příkazy MQSC v systému z/OS](#)

### **Související úlohy**

[Nastavení zabezpečení v systému z/OS](#)

[Porovnání zabezpečení na úrovni odkazů 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 zabezpečení zapnuto 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 na úrovni rozhraní API.

### **Zabezpečení podsystému**

Zabezpečení podsystému je ovládací prvek, který uvádí, zda se provádí kontrola zabezpečení pro celého správce front. Pokud nevyžadujete kontrolu zabezpečení (například na testovacím systému) nebo pokud jste s úrovní zabezpečení na všech prostředcích, které se mohou připojit k produktu IBM MQ (včetně klientů a kanálů) spokojeni, můžete vypnout kontrolu zabezpečení u správce front nebo skupiny sdílení front, aby nedošlo k další kontrole zabezpečení.

Toto je jediná kontrola, která může zcela vypnout zabezpečení a určit, zda jsou provedeny jakékoli jiné kontroly zabezpečení ochrany dat. To znamená, že pokud vypnete kontrolu nad správcem front nebo skupinou sdílení front, nebude provedena žádná jiná kontrola IBM MQ; pokud ji necháte zapnutou, produkt IBM MQ zkontroluje vaše požadavky na zabezpečení pro další IBM MQ prostředky.

Můžete také zapnout nebo vypnout zabezpečení určitých sad prostředků, jako jsou například příkazy.

### **Kontrola úrovně správce front nebo skupiny sdílení front**

Zabezpečení je možné implementovat na úrovni správce front nebo na úrovni skupiny sdílení front. Pokud implementujete zabezpečení na úrovni skupiny sdílení front, všichni správci front v této skupině sdílejí stejné profily. To znamená, že je třeba méně profilů definovat a udržovat, což usnadňuje správu zabezpečení. Rovněž usnadňuje přidání nového správce front do skupiny sdílení front, protože je dědit existující profily zabezpečení.

Je také možné implementovat kombinaci obou, pokud to vaše instalace vyžaduje, například během migrace, nebo pokud máte jednoho správce front ve skupině sdílení 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 je prováděna pro celou skupinu sdílení front. Umožňuje zjednodušit administraci zabezpečení, protože vyžaduje definování menšího počtu profilů zabezpečení. Autorizace ID uživatele pro použití konkrétního prostředku je ošetřena na úrovni skupiny sdílení front a je nezávislá na tom, který správce front používá pro přístup k prostředku toto ID uživatele.

Řekněme například, že serverová aplikace běží pod ID uživatele SERVER a chce přístup do fronty s názvem SERVER.REQUEST a chcete spustit instanci SERVER na každém obrazu z/OS v prostředí sysplex. Spíše než povoluje SERVER otevírat SERVER.REQUEST v každém správci front (zabezpečení na úrovni správce front) můžete povolit přístup pouze na úrovni skupiny sdílení front.

Profily zabezpečení na úrovni skupiny sdílení front můžete 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 obou správců front a zabezpečení na úrovni skupiny sdílení front.

Nastavení zabezpečení na úrovni skupiny sdílení front pro konkrétního správce front, který je členem této skupiny, lze potlačit. To znamená, že můžete provádět různé kontroly zabezpečení jednotlivých správců front na úrovni, které jsou prováděny u ostatních správců front v dané skupině.

Další informace naleznete v tématu [Profily pro zabezpečení skupiny sdílení front nebo zabezpečení na úrovni správce front](#).

## **Řízení počtu zkontrolovaných ID uživatelů**

RESLEVEL je profil serveru zabezpečení, který řídí počet ID uživatelů kontrolovaných pro zabezpečení prostředků produktu IBM MQ . Když se uživatel obvykle pokusí o přístup k prostředku IBM MQ , zkontroluje příslušné ID uživatele nebo ID uživatele, aby zjistil, zda je k tomuto prostředku přístup povolen. Definováním profilu RESLEVEL můžete určit, zda má být provedena kontrola nula, jednoho nebo případně dvou ID uživatelů.

Tyto ovládací prvky jsou prováděny na základě připojení a poslední po dobu trvání připojení.

Pro každého správce front existuje pouze jeden profil RESLEVEL. Řízení je implementováno přístupem, který má ID uživatele k tomuto profilu.

## **Smíšené nebo velké třídy IBM MQ RACF**

Nyní můžete použít podporu smíšených profilů případu produktu RACF , která vám umožňuje použít názvy prostředků smíšených případů a definovat profily produktu IBM MQ RACF k jejich ochraně.

Můžete zvolit jednu z následujících možností:

- Pokračujte pouze s použitím velkých písmen produktu IBM MQ RACF , jako v předchozích verzích, nebo
- Použijte nové třídy IBM MQ RACF se smíšenými malými a velkými písmeny.

Bez použití smíšených profilů případu produktu RACF můžete v produktu IBM MQ for z/OS i nadále používat názvy prostředků se smíšenými malými a velkými písmeny; Tyto názvy prostředků však mohou být chráněny pouze generickými profily RACF ve třídách IBM MQ velkých písmen. Při použití podpory smíšených profilů případu produktu IBM MQ RACF můžete poskytnout přesnější úroveň ochrany definováním profilů produktu IBM MQ RACF ve smíšených třídách případu IBM MQ .

Když se správce front spustí nebo je-li to instruován příkazem operátora, IBM MQ for z/OS určuje, které prostředky chcete chránit.

Můžete řídit, které kontroly zabezpečení se budou provádět pro jednotlivé správce front. Můžete například implementovat určitý počet kontrol zabezpečení v produkčním správci front, ale žádný ve správci front testu.

## Zabezpečení připojení

Kontrola zabezpečení připojení je prováděna buď při pokusu aplikačního programu o připojení ke správci front. To se provádí zadáním požadavku MQCONN nebo MQCONNX, nebo když adaptér kanálu nebo adaptér CICS nebo IMS vydá požadavek na připojení.

Pokud používáte zabezpečení na úrovni správce front, můžete vypnout kontrolu zabezpečení připojení u konkrétního správce front. Pokud se však tento uživatel může k tomuto správci front připojit, může se k tomuto správci front připojit.

Pro adaptér CICS se používá pouze ID uživatele adresního prostoru CICS pro kontrolu zabezpečení připojení, nikoli pro ID uživatele terminálu CICS. Pokud se pro adaptér IMS připojí řídicí nebo závislé oblasti IMS k produktu IBM MQ, zkontroluje se ID uživatele adresního prostoru IMS. U inicializátoru kanálu je zaškrtnuto 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 příkazů se provádí, když uživatel vydá příkaz MQSC z libovolného ze zdrojů popsaných v tématu [Vydávání příkazů](#). Můžete provést samostatnou kontrolu prostředku zadaného pomocí příkazu, jak je popsáno v tématu [“Zabezpečení prostředků příkazů”](#) na stránce 262.

Pokud vypnete kontrolu příkazů, emitenti příkazů se nekontrolují, zda mají oprávnění k vydání příkazu.

Pokud jsou příkazy MQSC zadány z konzoly, konzola musí mít atribut oprávnění konzoly z/OS SYS. Příkazy, které jsou vydány z datových sad CSQINP1 nebo CSQINP2, nebo interně správcem front, jsou vyloučeny ze všech kontrol zabezpečení, zatímco ty pro CSQINPX používají ID uživatele adresního prostoru inicializátoru kanálu. Chcete-li tyto datové sady aktualizovat prostřednictvím běžné ochrany datové sady, musíte určit, kdo má povoleno aktualizovat tyto datové sady.

Kontrolu zabezpečení příkazů lze zapnout nebo vypnout buď u správce front, nebo na úrovni skupiny sdílení front.

## Zabezpečení prostředků příkazů

Některé příkazy MQSC, například definování lokální fronty, zahrnují manipulaci s prostředky produktu IBM MQ. Je-li zabezpečení prostředků příkazů aktivní, při každém vydání příkazu, který zahrnuje prostředek, produkt IBM MQ zkontroluje, zda je uživateli povoleno změnit definici tohoto prostředku.

Můžete použít zabezpečení prostředků příkazů, abyste pomohli vynutit standardy pojmenování. Administrátor mezd může například odstranit a definovat pouze fronty s názvy začínajícími "PAYROLL". Je-li zabezpečení prostředků příkazů neaktivní, nejsou prováděny žádné kontroly zabezpečení prostředku, se kterým manipuluje daný příkaz. Nezaměňujte zabezpečení s příkazovým prostředkem se zabezpečením příkazů; tyto dva jsou nezávislé.

Vypnutí kontroly zabezpečení prostředků příkazu neovlivní kontrolu prostředků, která je prováděna speciálně pro jiné typy zpracování, které nezahrnují příkazy.

Kontrolu zabezpečení prostředků příkazů lze zapnout nebo vypnout buď na úrovni správce front, nebo na úrovni skupiny sdílení front.

## Aspekty zabezpečení kanálu

### Zabezpečení kanálu

Když používáte kanály, jsou dostupné funkce zabezpečení závislé na tom, který komunikační protokol budete používat. Používáte-li protokol TCP, nejsou k dispozici žádné funkce zabezpečení s komunikačním protokolem, i když můžete použít TLS. Pokud používáte APPC, můžete informace o ID uživatele směřovat z odesílajícího agenta MCA prostřednictvím sítě do cílového agenta MCA pro ověření.

Pro oba protokoly můžete uvést, která ID uživatelů chcete zkontrolovat pro účely zabezpečení, a kolik jich bude. Opět platí, že volby, které máte k dispozici, závisí na protokolu, který používáte, toho, co určíte při definování kanálu, a nastavení RESLEVEL pro inicializátor kanálu.

Další informace o typech dostupných zabezpečení kanálu naleznete v tématu [Záznamy ověřování kanálu](#) a [Přehled uživatelských procedur zabezpečení](#).

### Související odkazy

“Zabezpečení na úrovni rozhraní API v produktu IBM MQ for z/OS” na stránce 263

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 na úrovni rozhraní API v produktu 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 na úrovni rozhraní API je rozděleno na tyto kontroly:

- [Fronta](#)
- [process, proces](#)
- [Namelist, Seznam názvů](#)
- [Alternativní uživatel](#)
- [Kontext](#)

Při otevírání objektu správce front nebo při přístupu k objektům třídy úložiště nejsou provedeny žádné kontroly zabezpečení.

### Fronta

Kontrola zabezpečení fronty určuje, kdo má povoleno otevřít frontu, a jaké možnosti je s nimi povoleno otevřít. Uživatel může mít například povoleno otevřít frontu s názvem PAYROLL.INCREASE.SALARY můžete procházet zprávy ve frontě (pomocí volby MQOO\_BROWSE), nikoli však odebírat zprávy z fronty (pomocí jedné z voleb MQOO\_INPUT\_\*). Pokud vypnete kontrolu nad frontami, každý uživatel může otevřít libovolnou frontu s libovolnou platnou volbou otevření (to znamená, že jakákoli platná volba MQOO\_\* na 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, každý uživatel může otevřít jakýkoli proces.

Kontrolu zabezpečení procesu můžete zapnout nebo vypnout buď na úrovni správce front, nebo na úrovni skupiny sdílení front.

## Seznam názvů

Kontrola zabezpečení seznamu názvů se provádí, když uživatel otevře seznam názvů. Pokud vypnete kontrolu názvů jmenovek, může libovolný uživatel otevřít libovolný seznam názvů.

Kontrolu zabezpečení seznamu názvů můžete zapnout nebo vypnout na úrovni správce front nebo skupiny sdílení front.

## Alternativní uživatel

Zabezpečení alternativního uživatele řídí, zda jedno ID uživatele může použít oprávnění jiného ID uživatele k otevření objektu IBM MQ .

Příklad:

- Serverový program spuštěný pod ID uživatele PAYSERV načte zprávu požadavku z fronty, která byla vložena do fronty, pomocí ID uživatele USER1.
- Když serverový program získá zprávu požadavku, zpracuje požadavek a vrátí odpověď zpět do fronty pro odpověď, která je uvedena spolu se zprávou požadavku.
- Místo použití vlastního ID uživatele (PAYSERV) k autorizaci otevření fronty pro odpověď může server v tomto případě zadat nějaké další ID uživatele, USER1. V tomto příkladu by zabezpečení alternativního uživatele kontroloval, zda má uživatel ID PAYSERV povoleno zadat ID uživatele USER1 jako alternativní ID uživatele při otevření fronty pro odpověď.

Alternativní ID uživatele je určeno v poli *AlternateUserId* v deskriptoru objektu (MQOD).

Na libovolném objektu typu IBM MQ můžete použít alternativní ID uživatelů, například procesy nebo seznamy názvů. Neovlivní ID uživatele použité jinými správci prostředků, například pro zabezpečení produktu CICS nebo pro zabezpečení datové sady produktu z/OS .

Není-li zabezpečení alternativního uživatele aktivní, může kterýkoli uživatel použít libovolné jiné ID uživatele jako alternativní ID uživatele.

Kontrolu zabezpečení alternativního uživatele můžete zapnout nebo vypnout ve správci front nebo na úrovni skupiny sdílení front.

## Kontext

Kontext je informace, která se vztahuje ke konkrétní zprávě a která je obsažena v deskriptoru zpráv (MQMD), který je součástí zprávy. Informace o kontextu se nacházejí ve dvou sekcích:

### Sekce Identita

Uživatel aplikace, který nejprve vložil zprávu do fronty. Skládá se z následujících polí:

- *UserIdentifier*
- *AccountingToken*
- *ApplIdentityData*

### Sekce Původ

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 funkce MQPUT nebo volání MQPUT1 . Aplikace může generovat data, data mohou být předána z jiné zprávy, nebo může správce front vygenerovat data standardně. Serverové programy mohou například použít data kontextu ke kontrole identity žadatele,



to znamená, že tato zpráva pochází ze správné aplikace? Typicky se pole *UserIdentifier* používá k určení ID uživatele alternativního uživatele.

Zabezpečení kontextu se používá k určení, zda uživatel může určit libovolnou kontextovou volbu pro volání MQOPEN nebo MQPUT . Informace o volbách kontextu naleznete v tématu [Volby MQOPEN týkající se kontextu zprávy](#). Popisy polí deskriptoru zpráv týkajících se kontextu viz [MQMD-Message descriptor](#)MQMD -Message descriptor.

Pokud vypnete kontrolu zabezpečení kontextu, může kterýkoli uživatel použít libovolnou z kontextových voleb, které povoluje zabezpečení fronty.

Kontrolu zabezpečení kontextu můžete zapnout nebo vypnout ve frontě, ve správci front nebo ve skupinové úrovni sdílení front.

z/OS

## Dostupnost v systému z/OS

Produkt IBM MQ for z/OS má mnoho funkcí pro vysokou dostupnost. Toto téma popisuje některé aspekty dostupnosti.

Pokud správce front nebo inicializátor kanálu selže, může zvýšit dostupnost systému pomocí více funkcí produktu IBM MQ . Další informace o těchto funkcích naleznete v následujících sekcích:

- [Aspekty prostředí sysplex](#)
- [Sdílené fronty](#)
- [Sdílené kanály](#)
- [Dostupnost sítě IBM MQ](#)
- [Použití produktu z/OS Automatic Restart Manager \(ARM\)](#)
- [Použití produktu 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* několik obrazů operačního systému z/OS spolupracuje v obrazu jednoho systému a komunikuje pomocí prostředku Coupling Facility. Produkt IBM MQ může využívat zařízení prostředí sysplex pro vyšší dostupnost.

Odebrání afinit mezi správcem front a určitým obrazem produktu z/OS umožňuje restartování správce front v jiném obrazu produktu z/OS v případě selhání obrázku. Mechanismus restartování může být ruční, použití ARM nebo automatizace systému, pokud se ujistíte o následujících:

- Na sdílených svazcích jsou definovány všechny sady stránek, protokoly, zaváděcí datové sady, knihovny kódu a konfigurační datové sady správce front.
- Definice subsystému má rozsah prostředí sysplex a jedinečný název v rámci prostředí sysplex.
- Úroveň kódu *early code* instalovaného na každém obrazu z/OS v době IPL je na stejné úrovni.
- TCP virtual IP addresses (VIPA) is available on each TCP stack in the sysplex, and you have configured IBM MQ TCP listeners and inbound connections to use VIPAs rather than default host names.

Další informace o používání protokolu TCP v prostředí sysplex najdete v tématu [TCP/IP v prostředí sysplex](#), SG24-5235, publikace IBM Redbooks .

Dále můžete nakonfigurovat více správců front spuštěných na různých obrazech operačního systému v prostředí sysplex, aby fungovaly jako skupina sdílení front, která může využívat sdílené fronty a sdílené kanály pro vyšší dostupnost a vyrovnávání pracovní zátěže.

### Sdílené fronty

Ve skupinovém 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; libovolný správce front v rámci skupiny sdílení front může sloužit jako služba pro všechny fronty. Tím získáte 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 viz [“Výhody používání sdílených front”](#) na stránce 178.

Chcete-li dále zvýšit dostupnost zpráv ve skupině sdílení front, produkt IBM MQ zjistí, zda je jiný správce front ve skupině odpojen od zařízení pro spojení nestandardně, a aby dokončil jednotky práce pro daného správce front, které jsou stále nevyřízené, je-li to možné. To se označuje jako *zotavení typu peer* a je popsáno v tématu [“Obnova partnerských serverů”](#) na stránce 252.

Zotavení typu peer nemůže obnovit jednotky práce, které byly nejisté v době selhání. Správce automatického restartování (ARM) můžete použít k restartování všech systémů zahrnutých do selhání (například CICS, Db2a IBM MQ) a aby se zajistilo, že všechny systémy budou restartovány na stejném novém procesoru. To znamená, že mohou resynchronizovat a poskytují rychlé zotavení v nejistých pracovních jednotkách. Tento popis je popsán v tématu [“Použití správce automatického restartu z/OS \(ARM\)”](#) na stránce 266.

## Sdílené kanály

Ve skupinovém prostředí skupiny sdílení front produkt IBM MQ poskytuje funkce, které poskytují vysokou dostupnost sítě. Inicializátor kanálu umožňuje používat síťové produkty, které vyrovnávají síťové požadavky v rámci sady vhodných serverů a skrývají selhání serveru ze sítě (například generické prostředky VTAM). IBM MQ používá generický port pro příchozí požadavky, aby bylo možné požadavky na připojení směřovat na libovolný dostupný inicializátor kanálu ve skupině sdílení front. Tento popis je popsán v tématu [“Sdílené kanály”](#) na stránce 198.

Sdílené odchozí kanály přijímají zprávy, které odesílají ze sdílené přenosové fronty. Informace o stavu sdíleného kanálu jsou udržovány na jednom místě pro celou úroveň skupiny sdílení front. To znamená, že kanál může být automaticky restartován na jiném inicializátoru kanálu ve skupině sdílení front, pokud selže inicializátor kanálu, správce front nebo komunikační subsystém. To se nazývá *zotavení partnerského kanálu* a je popsáno v tématu [Sdílené odchozí kanály](#).

## Dostupnost sítě IBM MQ

Zprávy produktu IBM MQ jsou přenášeny ze správce front do správce front v síti produktu IBM MQ s použitím kanálů. Konfiguraci můžete změnit na několika úrovních, a zlepšit tak dostupnost sítě správce front a schopnost kanálu produktu IBM MQ detekovat síťový problém a znovu navázat spojení.

Protokol TCP *Keepalive* je k dispozici pro kanály TCP/IP. Způsobí, že TCP pravidelně odesílá pakety mezi relacemi za účelem zjištění selhání sítě. Atribut kanálu KAINTE určuje frekvenci těchto paketů pro kanál.

*AdoptMCA* umožňuje kanálu, blokováným v přijímacích zpracování v důsledku výpadku sítě, který má být ukončen a nahrazen novým požadavkem na připojení. Pomocí vlastnosti správce front ADOPTMCA pomocí obslužného programu MQSC nebo pomocí vlastnosti AdoptNewMCAType s rozhraním Programmable Command Formats použijte vlastnost AdoptMCA .

Volba *ReceiveTimeout* zabrání tomu, aby kanál byl trvale zablokován v rámci volání příjmu sítě. Parametry inicializátoru kanálu RCVTIME a RCVTMIN, určují charakteristiky časového limitu příjmu pro kanály, jako funkce intervalu prezenčního signálu. Další podrobnosti lze najít v tématu [Parametr správce front](#) .

## Použití správce automatického restartu z/OS (ARM)

Produkt IBM MQ for z/OS můžete použít ve spojení se správcem automatického restartu z/OS (ARM). Pokud došlo k selhání správce front nebo inicializátoru kanálu, ARM ji restartuje na stejném obrazu produktu z/OS . Pokud příkaz z/OS selže, dojde také k selhání celé skupiny souvisejících subsystémů

a aplikací. ARM může restartovat všechny nezdařené systémy automaticky, v předdefinovaném pořadí, na jiném obrazu produktu z/OS v prostředí sysplex. To se nazývá cross-system restart.

ARM umožňuje rychlé zotavení transakcí s nejistým stavem v prostředí sdílené fronty. Také poskytuje vyšší dostupnost, pokud nepoužíváte skupiny sdílení front.

Pomocí ARM můžete restartovat správce front na jiném obrazu produktu z/OS v rámci prostředí sysplex v případě selhání produktu z/OS.

Chcete-li povolit automatický restart, musíte provést následující:

1. Nastavení datové sady spojení ARM.
2. Definujte akce automatického restartu, které má z/OS provádět v *zásadě ARM*.
3. Spusťte zásadu ARM.

Chcete-li restartovat správce front v různých obrazech produktu z/OS automaticky, každý správce front v každém obrazu produktu z/OS, na kterém může být restartován správce front, musí být definován s jedinečným 4znakovým názvem subsystému prostředí sysplex.

Použití ARM s IBM MQ je popsáno v tématu [Použití ARM v síti IBM MQ](#).

## **Použití nástroje z/OS Extended Recovery Facility (XRF)**

Produkt IBM MQ můžete použít v prostředí funkce Extended Recovery Facility (XRF). Všechny datové sady vlastněné IBM MQ (spustitelný kód, BSDS, protokoly a sady stránek) musí být v DASD sdílené mezi aktivními a alternativními procesory XRF.

Používáte-li pro zotavení XRF, musíte zastavit správce front v aktivním procesoru a spustit jej na alternativním procesoru. Pro produkt CICS to můžete provést pomocí tabulky se seznamem příkazů (CLT) poskytnuté serverem CICS nebo ji může operátor systému provést ručně. Pro produkt IMS je to ruční operace a vy ji musíte provést poté, co koordinační systém IMS dokončí přepínač procesoru.

Aby bylo možné správce front přepnout na alternativní procesor, je třeba dokončit nebo ukončit obslužné programy produktu IBM MQ. Zvažte vliv tohoto potenciálního přerušení při plánování plánů obnovy XRF.

Dbejte na to, abyste zabránili spuštění správce front v alternativním procesoru před ukončením správce front v aktivním procesoru. Předčasný start může způsobit závažné problémy s integritou v datech, katalogu a v protokolu. Použití globální serializace prostředků (GRS) pomáhá předcházet problémům s integritou tím, že zabraňuje současnému použití IBM MQ na těchto dvou systémech. Musíte zahrnout sadu BSDS jako chráněný prostředek a musíte zahrnout aktivní a alternativní procesory XRF v kruhu GRS.

## **Použití atributu z/OS GROUPUR pro zotavení ve skupině sdílení front**

Skupiny sdílení front (QSG) umožňují další transakční funkce, které jsou popsány v tomto tématu. Atribut GROUPUR umožňuje klientským aplikacím XA mít jakékoli případné neověřené zotavení transakce, které lze provést na libovolném členu skupiny QSG.

Pokud se aplikace klienta XA připojí ke skupině sdílení front (QSG) prostřednictvím prostředí sysplex, nelze zaručit, ke kterému konkrétnímu správci front se připojí. Použití atributu GROUPUR od správců front v rámci skupiny sdílení front může umožnit jakékoli neověřené zotavení transakce, které může být nezbytné k provedení na libovolném členu 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 aplikaci klienta XA ze všech závislostí na specifických členech skupiny sdílení front, a tím rozšiřuje dostupnost správce front. Skupina sdílení front se zobrazí v transakční aplikaci jako jediná entita, která poskytuje všechny funkce produktu IBM MQ a bez selhání jednoho správce front.

Tato funkčnost není viditelná pro transakční aplikaci.

## Kde najít další informace o dostupnosti

Další informace o těchto tématech naleznete z následujících zdrojů:

| Téma                                                        | Kde hledat                                                              |
|-------------------------------------------------------------|-------------------------------------------------------------------------|
| Skupiny sdílení front                                       | <a href="#">“Sdílené fronty a skupiny sdílení front” na stránce 158</a> |
| Parametry systému                                           | <a href="#">Konfigurace systémových parametrů</a>                       |
| Použití správce automatického restartu<br>Obslužné programy | <a href="#">Použití ARM v síti IBM MQ</a>                               |
| Příkazy MQSC                                                | <a href="#">Příkazy MQSC</a>                                            |

z/OS

## Monitorování a statistika v systému IBM MQ for z/OS

IBM MQ for z/OS má sadu zařízení pro monitorování správce front a shromažďování statistik.

IBM MQ poskytuje zařízení pro monitorování systému a shromažďování statistiky. Další informace o těchto zařízeních naleznete v následujících sekcích:

- [“Monitorování online” na stránce 268](#)
- [“IBM MQ trasování” na stránce 268](#)
- [“Události” na stránce 269](#)

### Monitorování online

Produkt IBM MQ obsahuje následující příkazy pro monitorování stavu objektů IBM MQ :

- Funkce DISPLAY CHSTATUS zobrazuje stav určeného kanálu.
- ZOBRAZIT QSTATUS zobrazuje stav zadané fronty.
- Příkaz DISPLAY CONN zobrazí stav určeného připojení.

Další informace o těchto příkazech najdete v tématu [Příkazy MQSC](#).

### IBM MQ trasování

Produkt IBM MQ poskytuje prostředek trasování, který můžete použít ke shromáždění následujících informací za běhu správce front:

#### Statistika výkonu

Trasování statistiky shromažďuje následující informace, které vám pomohou monitorovat výkon a vyladit systém:

- Počty různých požadavků MQI (statistika správce zpráv)
- Počty různých požadavků objektů (statistika správce dat)
- Informace o využití Db2 (statistika správce Db2)
- Informace o využití prostředku Coupling Facility (statistiky správce prostředku Coupling Facility)
- Informace o využití SMDS (sdílené statistiky datové sady zpráv)
- Informace o využití fondu vyrovnávacích pamětí (statistika správce vyrovnávací paměti)
- Informace o protokolování (statistika správce protokolů)

- Informace o využití úložiště (statistika správce datových úložišť)
- Informace o požadavcích zámku (statistika správce zámků)

## Účetní údaje

- Záznam sledování evidence shromažďuje informace o času procesoru stráveném zpracováním volání MQI a o počtu požadavků MQPUT a MQGET provedených konkrétním uživatelem.
- Produkt IBM MQ může také shromažďovat informace o každé úloze pomocí produktu IBM MQ. Tato data se shromažďují jako účtovací záznam na úrovni podprocesu. Pro každé vlákno produkt IBM MQ také shromažďuje informace o každé frontě použité tímto podprocesem.

Data generovaná trasováním se odesílají do prostředku SMF (System Management Facility) nebo GTF (generalized Trace Facility).

## Události

Události produktu IBM MQ poskytují informace o chybách, varováních a dalších významných událostech ve správci front. Začleněním těchto událostí do své vlastní aplikace pro správu systému můžete monitorovat aktivity v mnoha správcích front, a to pro více aplikací produktu IBM MQ. Zejména můžete monitorovat všechny správce front ve vašem systému z jednoho správce front.

Události mohou být hlášeny prostřednictvím uživatelsky napsaného mechanismu hlášení do aplikace administrace, která podporuje prezentaci událostí operátorovi. Události také umožňují aplikacím působícím jako agenti pro jiné administrativní sítě, např. NetView, monitorovat sestavy a vytvářet odpovídající výstrahy.

### Související úlohy

[Použití trasování produktu IBM MQ](#)

[Použití událostí produktu IBM MQ](#)

**z/OS**

## Dispozice jednotky zotavení v systému z/OS

Určité transakční aplikace mohou při připojení ke správci front ve skupině sdílení front (QSG) při připojení ke správci front v rámci skupiny sdílení front (QSG) při připojení ke správci front používat místo názvu správce front místo názvu skupiny sdílení front jako název skupiny sdílení front. To umožňuje, aby zotavení transakce bylo pružnější a robustnější tím, že odeberete požadavek na opětovné připojení ke stejnému správci front v QSG.

Transakce spuštěné aplikacemi, které se připojily s použitím názvu skupiny sdílení front, mají také dispozici skupina GROUP obnovy.

Když se transakční aplikace připojí k dispozici GROUP procesu zotavení, je logicky připojeno ke skupině sdílení front a nemá afinitu k žádnému určitému správci front. Všechny transakce s dvoufázovým potvrzováním, které spustily dokončení procesu phase-1 procesu odevzdání, tj. pochybně, lze vyšetřit a vyřešit, jsou-li připojeny k libovolnému správci front v rámci skupiny QSG. Ve scénáři zotavení to znamená, že koordinátor transakcí se nemusí znovu připojit ke stejnému správci front, což může být v daném okamžiku nedostupné.

Aplikace, které se připojují k dispozici správce front QMGR, mají přímou afinitu ke správci front, ke kterému jsou připojeny. V případě scénáře zotavení se koordinátor transakcí musí znovu připojit ke stejnému správci front, aby vyřešil všechny neověřené transakce bez ohledu na to, zda správce front patří do skupiny sdílení front.

Pokud aplikace určují název skupiny sdílení front a připojují se tak ke správci front v rámci skupiny QSG s dispozicí skupiny sdílení zotavení, bude skupina sdílení front logickým správcem prostředků logicky samostatného správce prostředků. To znamená, že neověřené transakce jsou viditelné pouze pro aplikaci, pokud se znovu připojí se stejnou jednotkou dispozice zotavení. Neověřené transakce s dispozicí QMGR pro dispozice zotavení nejsou viditelné pro aplikace, které se připojila k dispozici SKUPINY o dispozice zotavení a naopak.

## Související pojmy

“Povolení jednotek zotavení GROUP” na stránce 270

Skupina sdílení front může konfigurovat a povolit podporu pro jednotky SKUPINY typu "recovery".

“Podpora aplikací” na stránce 270

Na této stránce můžete určit, které aplikace se mohou připojit k dispoziční skupiny GROUP.

## Povolení jednotek zotavení GROUP

Skupina sdílení front může konfigurovat a povolit podporu pro jednotky SKUPINY typu "recovery".

Chcete-li pro správce front v rámci skupiny QSG použít skupiny objektů GROUP, povolte atribut správce front GROUPUR. Další informace o tomto konceptu naleznete v tématu [“Dispoziční jednotky zotavení v systému z/OS”](#) na stránce 269 před čtením zbytku tohoto tématu.

Je-li povolen atribut správce front GROUPUR, správce front přijímá nová připojení s dispoziční skupiny GROUP obnovy. Pokud zakážete tento atribut, nebudou přijata nová připojení s touto dispoziční, ačkoli aplikace, které jsou již připojeny, nejsou ovlivněny, dokud se neodpojují.

Když se aplikace připojí k dispoziční SKUPINY a buď rozloží transakce, které mají pochybnosti, nebo pokusy o vyřešení transakce spuštěné jinde ve skupině sdílení front (QSG), musí být správce front, k němuž je nyní připojen, schopen komunikovat s ostatními členy skupiny sdílení front, aby mohl zpracovat požadavek. K provedení tohoto úkolu používá sdílenou frontu s názvem SYSTEM.QSG.UR.RESOLUTION.QUEUE. Tato fronta musí být na zotavitelné struktuře aplikace s názvem CSQSYSAPPL. Struktura musí být obnovitelná, protože trvalé zprávy jsou v této frontě uloženy při zpracování požadavků na rozlišení.

Než budete moci povolit skupinové jednotky zotavení, je třeba zajistit, aby byla definována struktura prostředku Coupling Facility a sdílená fronta. Definice můžete použít v ukázce CSQ4INSS. Je-li fronta definována nebo zjištěna při spuštění, každý správce front v rámci skupiny sdílení front otevře frontu, aby mohla přijímat příchozí požadavky. Chcete-li frontu odstranit nebo ji přesunout, protože byla nesprávně definována, můžete požádat správce front o zavření jejich otevřených manipulátorů tím, že aktualizuje objekt fronty tak, aby blokoval požadavky MQGET. Když jste provedli nezbytné opravy, umožněte aplikacím získat zprávy z fronty ještě jednou přesměrovává každého správce front, aby jej znovu otevřel. Pomocí příkazu DISPLAY QSTATUS určete, které obslužné rutiny jsou ve frontě otevřené.

Po dokončení tohoto nastavení můžete povolit skupině zotavení GROUP v každém správci front, že chcete, aby byly transakční aplikace schopny připojit se k dispoziční jednotce SKUPINY zotavení. To nemusí být všechny správce front v rámci skupiny sdílení front, ale pokud se rozhodnete povolit pouze tuto funkci v podmnožině skupiny sdílení front, je třeba zajistit, aby se aplikace pokusily připojit pouze ke správcům front, v nichž jste je povolili. Další informace viz [“Podpora aplikací”](#) na stránce 270.

Když se pokusíte povolit atribut správce front GROUPUR, provede se několik kontrol konfigurace. Správce front kontroluje, zda:

- Patří do skupiny sdílení front.
- Sdílená fronta s názvem SYSTEM.QSG.UR.RESOLUTION.QUEUE byla definována v souladu s definicí v CSQ4INSS.
- SYSTEM.QSG.UR.RESOLUTION.QUEUE je na obnovitelné struktuře CF s názvem CSQSYSAPPL.

Pokud některá z výše uvedených kontrol selže, atribut GROUPUR zůstane vypnutý a vrátí se kód zprávy.

Tyto kontroly konfigurace se také provádějí při spuštění správce front, je-li povolen atribut správce front. Pokud se některá z kontrol nezdaří během spuštění jednotek zotavení GROUP, je funkce zotavení zakázána a správce front vydá zprávu s identifikací, která kontrola selhala. Když jste provedli nezbytnou nápravnou akci, musíte znovu povolit atribut správce front.

## Podpora aplikací

Na této stránce můžete určit, které aplikace se mohou připojit k dispoziční skupiny GROUP.

Podpora pro skupinu zotavení GROUP je omezena na určité typy transakčních aplikací, pro které správce IBM MQ for z/OS je správce prostředků, nikoli však koordinátor transakcí. Aktuálně podporované transakční aplikace jsou:

- Rozšířené transakční klientské aplikace IBM MQ
- Aplikace produktu IBM MQ classes for JMS spuštěné na aplikačním serveru, jako je například produkt WebSphere Application Server.
- Aplikace produktu CICS spuštěné v produktu CICS Transaction Server 4.2 nebo novější, je-li definice prostředku MQCONN produktu CICS konfigurována s parametrem RESYNCMEMBER (GROUPRESYNC).

### **Související pojmy**

[“Rozšířené transakční klientské aplikace IBM MQ” na stránce 271](#)

Na této stránce můžete určit, jak produkt IBM MQ rozšířený transakční klient může používat dispozice GROUP (GROUP) zotavení.

[“Aplikace rozhraní CICS” na stránce 271](#)

Na této stránce můžete určit, jak může produkt CICS využívat dispozice GROUP (GROUP) odebrání zotavení.

### **z/OS Rozšířené transakční klientské aplikace IBM MQ**

Na této stránce můžete určit, jak produkt IBM MQ rozšířený transakční klient může používat dispozice GROUP (GROUP) zotavení.

Příklad rozšířené transakční klientské aplikace klienta IBM MQ je takový, který používá produkt JMS a spouští se v produktu WebSphere Application Servera připojuje se k produktu IBM MQ přes TCP/IP, spíše než k lokálním vazbám. Tyto klientské aplikace se připojují k IBM MQ for z/OS přes síťová připojení, jako například přes TCP/IP. Pro tyto aplikace se jedná o hodnotu určenou pro parametr QMNAME řetězce xa\_info předaný v rámci volání xa\_open, který určuje, zda se použije objekt QMGR nebo GROUP zotavení pro zotavení. Další informace o parametru xa\_open naleznete v části [Formát řetězce xa\\_open](#) a [Další zpracování chyb pro xa\\_open](#). Pro aplikace produktu JMS se tato akce provádí zadáním názvu skupiny sdílení front (QSG) v rámci ConnectionFactory namísto názvu specifického správce front.

Aby klientské aplikace XA mohly využívat výhod dispozice GROUP, je třeba konfigurovat nastavení TCP/IP tak, aby bylo možné směřovat aplikace klienta do správců front v rámci skupiny sdílení front, která má povolený atribut GROUPUR, a nikoli pouze specifického správce front. Jedna z technologií dynamické virtuální adresy IP, kterou můžete použít k provedení této operace, je z/OS SysPlex Distributor. Další podrobnosti viz [z/OS Communications Server](#) a [z/OS Základní dovednosti: Dynamické virtuální adresování](#). Chcete-li skupině správců front ve skupině sdílení front povolit skupiny pro zotavení GROUP, ujistěte se, že vaše klientské aplikace nelze směřovat na ty, které nejsou povoleny. To zahrnuje všechny správce front ve verzi dřívější než IBM WebSphere MQ 7.0.1.

Chcete-li použít funkci dispozice skupiny zotavení v aplikaci klienta JMS, musíte použít knihovny klienta produktu IBM WebSphere MQ 7.0.1 verze JMS. Dřívější úrovně knihoven klienta JMS používají k dispozici jednotku zotavení správce front (QMGR).

Vaše klientské aplikace se nemusí připojovat ke skupině sdílení front prostřednictvím sdílených kanálů.

### **z/OS Aplikace rozhraní CICS**

Na této stránce můžete určit, jak může produkt CICS využívat dispozice GROUP (GROUP) odebrání zotavení.

Produkt CICS 4.2 a novější poskytuje volbu resynchronizace skupiny RESYNCMEMBER (GROUPRESYNC) v definici prostředku MQCONN. Produkt 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 oblast produktu CICS. Pro podporu volby CICS GROUPRESYNC musí být správce front spuštěn ve verzi MQ V7.1 nebo vyšší a musí být povolen pro podporu GROUPUR.

Transakce spuštěné v rámci oblasti CICS připojené k produktu MQ pomocí GROUPRESYNC vytvářejí jednotky práce se SKUPINOU jednotkou dispozice zotavení.

You can use RESYNCMEMBER(GROUPRESYNC) to enable faster recovery after a queue manager failure as it enables the CICS region to immediately connect to an alternative eligible queue manager running on the same LPAR, resolving any indoubt transactions as necessary, without waiting for queue manager restart.

RESYNCMEMBER (GROUPRESYNC) také umožňuje flexibilnější volby restartu pro CICS. Oblast CICS se svým připojením MQ konfigurovanou pro použití GROUPRESYNC a MQ sdílených front lze restartovat v libovolné oblasti LPAR, kde je správce front spuštěn jako člen stejné skupiny sdílení front.

## IBM MQ a další produkty z/OS

V tomto tématu se můžete seznámit s tím, jak produkt IBM MQ může pracovat s dalšími produkty z/OS .

### **Související pojmy**

[“IBM MQ a CICS” na stránce 272](#)

Všechny verze produktu CICS podporované produktem IBM MQ 9.0.0a pozdější používají zadanou verzi adaptéru a mostu produktu CICS .

[“IBM MQ for z/OS a WebSphere Application Server” na stránce 278](#)

Toto téma vám pomůže porozumět použití IBM MQ for z/OS u WebSphere Application Server.

### **Související odkazy**

[“IBM MQ a IMS” na stránce 273](#)

Toto téma vám pomůže pochopit, jak produkt IBM MQ pracuje s produktem IMS. Adaptér IMS umožňuje připojit správce front k produktu IMSa umožňuje aplikacím produktu IMS používat rozhraní MQI.

[“IBM MQ a adaptéry dávky, TSO a RRS produktu z/OS” na stránce 277](#)

Toto téma vám pomůže pochopit, jak produkt IBM MQ pracuje s adaptéry z/OS Batch, TSO a RRS.

## IBM MQ a CICS

Všechny verze produktu CICS podporované produktem IBM MQ 9.0.0a pozdější používají zadanou verzi adaptéru a mostu produktu CICS .

Další informace o konfiguraci adaptéru IBM MQ CICS a komponent produktu IBM MQ CICS bridge naleznete v části [Konfigurace připojení k produktu MQ](#) v dokumentaci produktu CICS .

### **Související úlohy**

[Použití IBM MQ s CICS](#)

## Připojení skupiny CICS

Připojení skupiny CICS poskytuje regionu CICS schopnost připojit se k libovolnému aktivnímu členu skupiny sdílení front produktu IBM MQ na stejné logické oblasti, než aby určoval jednotlivého 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 logické oblasti LPAR. Použití skupinového připojení poskytuje vyšší dostupnost, protože nepotřebujete, aby konkrétní správce front byl aktivní. Produkt CICS se připojuje k libovolnému správci front ve skupině sdílení front v oblasti LPAR.

Další informace naleznete v dokumentaci produktu CICS o prostředku MQCONN.

Produkt CICS se pokouší připojit k MQNAME předaným způsobem, jako kdyby 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 zjistil, které jsou aktivní ve stejné oblasti LPAR.
- Je-li aktivních více správců front, produkt CICS kontroluje RESYNCMEMBER (YES) a stav pracovní jednotky (UOW), aby určil, zda se produkt CICS potřebuje připojit ke konkrétnímu členu, nebo se má připojit ke konkrétnímu členu, nebo čekat, pokud není aktivní.
- Není-li třeba se připojit ke konkrétnímu členu, produkt CICS vybere správce front (pomocí algoritmu náhodného výběru).



- CICS se pokusí o připojení ke zvolenému správci front.
- Pokud se pokus nezdaří, pak v závislosti na návratovém kódu produkt CICS vybere dalšího člena a pak znovu přejde do smyčky výběru.
- Nejsou-li aktivní žádní správci front, produkt CICS odešle více připojení na seznam správců front a čeká na příkaz ECBLIST, dokud nebude k dispozici první správce front.

### Související pojmy

“Skupinové jednotky zotavení (GROUPUR) pro CICS” na stránce 273

Příkaz IBM MQ GROUPUR for CICS poskytuje rovnocenné zotavení pro nejisté jednotky práce ve skupině sdílení front (QSG). Jeden správce front produktu IBM MQ může v zastoupení jiného správce front v rámci skupiny sdílení front vyřešit neověřené jednotky práce. To znamená, že pokud se produkt CICS znovu připojí ke skupině připojené k jinému správci front v QSG, může vyřešit neověřené transakce z předchozího připojení produktu IBM MQ .

### Související informace

[Podpora pro skupiny sdílení front produktu IBM MQ](#)

## z/OS Skupinové jednotky zotavení (GROUPUR) pro CICS

Příkaz IBM MQ GROUPUR for CICS poskytuje rovnocenné zotavení pro nejisté jednotky práce ve skupině sdílení front (QSG). Jeden správce front produktu IBM MQ může v zastoupení jiného správce front v rámci skupiny sdílení front vyřešit neověřené jednotky práce. To znamená, že pokud se produkt CICS znovu připojí ke skupině připojené k jinému správci front v QSG, může vyřešit neověřené transakce z předchozího připojení produktu IBM MQ .

Pokud oblast produktu CICS pracuje se správcem front a správce front bude ukončen nestandardním způsobem, budou obnoveny všechny neověřené transakce. Tím vyloučíte nutnost, aby region CICS čekal na správce front, se kterým pracoval, a poté vyřešíte všechny neověřené transakce. To znamená, že v logické 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) v definici MQCONN produktu CICS :

- Používá funkci připojení skupiny IBM MQ a rovnocenné zotavení.
- Vyžaduje správce front s povoleným atributem GROUPUR.
- Stále podporuje existující nastavení MQCONN RESYNCMEMBER produktu CICS (ANO a NE):
  - Používá existující funkci připojení skupiny CICS a žádá partnerská obnova.
  - Změna nastavení RESYNCMEMBER se uplatní při příštím připojení CICS k prostoru IBM MQ.

### Související pojmy

“Povolení jednotek zotavení GROUP” na stránce 270

Skupina sdílení front může konfigurovat a povolit podporu pro jednotky SKUPINY typu "recovery".

## z/OS IBM MQ a IMS

Toto téma vám pomůže pochopit, jak produkt IBM MQ pracuje s produktem IMS. Adaptér IMS umožňuje připojit správce front k produktu IMSa umožňuje aplikacím produktu IMS používat rozhraní MQI.

Volitelný další most IBM MQ - IMS umožňuje aplikacím spouštět aplikaci produktu IMS , která nepoužívá rozhraní MQI. To znamená, že můžete používat své starší aplikace s produktem IBM MQ, aniž byste je museli přepsat.

Další informace o těchto komponentách najdete v následujících dílčích tématech:

### Související pojmy

[Aplikace mostu IMS a IMS v systému IBM MQ for z/OS](#)

### Související úlohy

[Nastavení adaptéru IMS](#)

[Nastavení mostu IMS](#)

## **Adaptér IMS**

Adaptér IMS je rozhraní mezi aplikačním programem IMS a subsystémem IBM MQ .

Adaptéry IBM MQ umožňují různým aplikačním prostředím odesílat a přijímat zprávy prostřednictvím sítě pro řazení zpráv do front zpráv. Adaptér IMS je rozhraní mezi aplikačním programem IMS a subsystémem IBM MQ . Umožňuje aplikačním programům produktu IMS používat rozhraní MQI.

Adaptér IMS přijímá a interpretuje požadavky na přístup k produktu IBM MQ pomocí modulu ESAF (External Subsystem Attach Facility) poskytovaného produktem IMS. Tato funkce je popsána v příručce *IMS Customization Guide*. Obvykle se IMS připojí k produktu IBM MQ automaticky bez zásahu operátora.

Adaptér IMS poskytuje přístup k prostředkům produktu IBM MQ pro programy spuštěné v následujících režimech nebo stavech:

- Režim úlohy (TCB)
- Stav problému
- Režim non-cross-memory
- Režim registru bez přístupu

Adaptér poskytuje podproces připojení z řídicího bloku úloh aplikace (TCB) do produktu IBM MQ.

Adaptér podporuje protokol s dvoufázovým potvrzováním pro změny provedené u prostředků vlastněných produktem IBM MQ a IMS chovat se jako koordinátor synchronizačního bodu. Konverzace, kde IMS není koordinátorem synchronizačního bodu, například konverzace chráněné APPC (SYNCLVL = SYNCPT), není podporováno adaptérem IMS .

Adaptér také poskytuje transakci monitoru spouštěčů (CSQQTRMN). Tento popis je popsán v tématu [“Monitor spouštěčů IMS”](#) na stránce 275.

Produkt IBM MQ můžete použít s produktem IMS Extended Recovery Facility (XRF) při podpoře zotavení z chyby IMS . Další informace o prostředí XRF naleznete v příručce *IMS Administration Guide: System* .

## **Použití adaptéru**

Aplikační programy a adaptér IMS se spouštějí ve stejném adresním prostoru. Správce front je oddělený ve svém vlastním adresním prostoru.

Každý program, který vydá jedno nebo více volání MQI do vhodného modulu rozhraní jazyka produktu IMS , je třeba propojit, a pokud nepoužívá dynamická volání MQI, program stub rozhraní API dodaný produktem IBM MQ, CSQQSTUB, pokud se používá dynamická volání MQI. Když aplikace vydá volání MQI, převede stub řízení na adaptér prostřednictvím rozhraní externího subsystému IMS , které spravuje zpracování požadavku správcem front zpráv.

## **Administrace systému a operace s produktem IMS**

Autorizovaný operátor terminálu IMS může vydávat příkazy IMS pro řízení a monitorování spojení s 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 produktu IBM MQ nejsou v rámci aplikace s použitím adaptéru IMS podporována:

- MQCB
- FUNKCE MQCB\_
- MQCTL

## Monitor spouštěčů IMS

The IMS trigger monitor ( **CSQQTRMN** ) is an IBM MQ-supplied IMS application that starts an IMS transaction when an IBM MQ event occurs, for example, when a message is put onto a specific queue.

### Způsob činnosti

Když je zpráva vložena do fronty zpráv aplikace, je spouštěč generován, jsou-li splněny podmínky spouštěče. Správce front poté zapíše zprávu (obsahující některá data definovaná uživatelem), známá jako *zpráva spouštěče*, do inicializační fronty, která byla uvedena pro tuto frontu zpráv. V prostředí IMS můžete spustit instanci CSQQTRMN pro monitorování inicializační fronty a pro načtení zpráv triggeru od ní, jakmile dorazí. Obvykle, CSQQTRMN naplánuje další transakci IMS příkazem INSERT (ISRT) do fronty zpráv IMS . Spuštěné aplikace IMS čte zprávu z fronty zpráv aplikace a pak ji zpracuje. CSQQTRMN musí být spuštěna jako nezpráva BMP.

Každá kopie služeb CSQQTRMN má jednu inicializační frontu. Když je spuštěna, monitor spouštěčů se spustí, dokud se nedokončí IBM MQ nebo IMS .

Makro APPLCTN pro CSQQTRMN musí určovat SCHDTYP=PARALLEL.

Protože monitor spouštěčů je dávkově orientovaný BMP, IMS transakce spuštěné monitorem spouštěčů obsahují následující údaje:

- Mezery v poli LTERM v IOPCB
- Název PSB monitoru spouštěčů ve formátu BMP v poli ID uživatele IOPCB

Pokud je cílová transakce IMS chráněná serverem Security Server (dříve známá jako RACF ), možná budete muset definovat CSQQTRMN jako ID uživatele pro server zabezpečení.

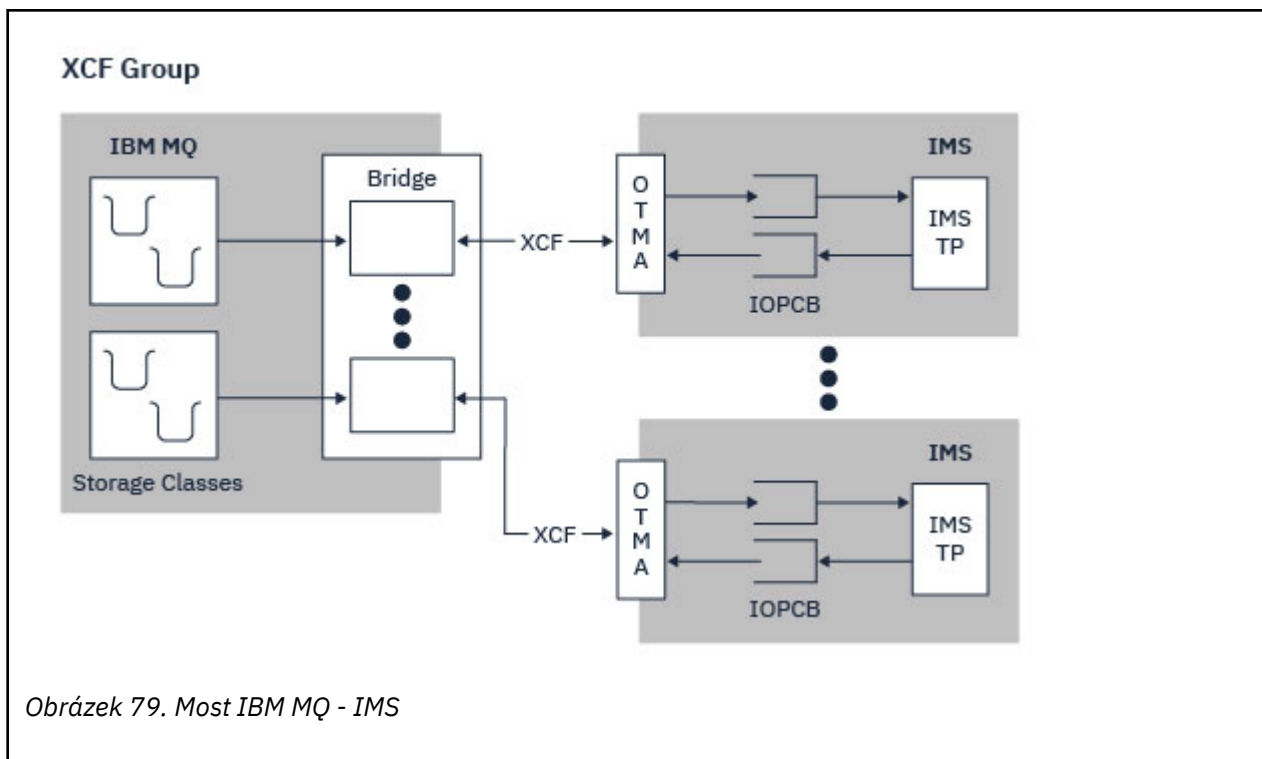
## Most IBM MQ - IMS

Most IBM MQ - IMS je komponenta produktu IBM MQ for z/OS , která umožňuje přímý přístup z aplikací produktu IBM MQ k aplikacím ve vašem systému IMS .

Most IBM MQ - IMS umožňuje *implicitní podporu MQI*. To znamená, že můžete přepracovat starší aplikace, které byly ovládány terminály připojenými pomocí 3270, které mají být řízeny zprávami IBM MQ , aniž by bylo nutné je přepisovat, znovu kompilovat nebo znovu propojit. Most je klientem produktu IMS *Open Transaction Manager Access* (OTMA).

V aplikacích mostu nejsou k dispozici žádná volání produktu IBM MQ v rámci aplikace IMS . Aplikace dostane svůj vstup pomocí příkazu GET UNIQUE (GU) do IOPCB a odešle svůj výstup pomocí ISRT do IOPCB. Aplikace produktu IBM MQ používají záhlaví IMS (strukturu MQIIH) v datech zprávy k zajištění toho, aby se aplikace mohly provádět tak, jak byly řízeny při neprogramovatelných terminálech. Pokud používáte aplikaci IMS , která zpracovává vícesegmentové zprávy, uvědomte si, že všechny segmenty by měly být obsaženy v jedné zprávě IBM MQ .

Most IMS je ilustrován v produktu [Obrázek 79](#) na stránce 276.



Obrázek 79. 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 náležet do stejné skupiny XCF a musí být všechny ve stejném prostředí sysplex.

Informace o nastavení mostu produktu IMS a přidání dalšího připojení produktu IMS ke stejnému správcí front najdete v tématu [Nastavení mostu IMS](#) .

## Co je OTMA?

Zařízení IMS OTMA je bezspojový protokol klienta/serveru, který běží na systému IMS 5.1 nebo novějším. Funkce slouží jako rozhraní pro komunikační servery založené na hostiteli, které přistupují k aplikacím IMS TM prostřednictvím z/OS prostředí CF (Cross Systems Coupling Facility) (XCF).

OTMA umožňuje klientům připojit se k produktu IMS za účelem zajištění vysokého výkonu pro interakce mezi klienty a produktem IMS pro velkou síť nebo velké množství relací. OTMA je implementováno v prostředí sysplex produktu z/OS . Doména OTMA je proto omezena na doménu XCF.

## Monitorování prostředků OTMA

Podpora zpráv protokolu OTMA pro '3C', která je k dispozici v produktu IMS v10 nebo vyšší, byla přidána do mostu IBM MQ - IMS v produktu IBM MQ for z/OS v7.1. Tyto zprávy jsou odesílány klientům OTMA produktem IMS za účelem ohlášení jeho stavu. Nemůže-li partner produktu IMS zpracovat odeslaný objem požadavků na transakce, oznámí to IBM MQ , že došlo k varovným varovným varovným signálem. V odezvě IBM MQ zpomalí rychlost, jakou jsou požadavky odeslány přes most. Pokud IMS stále není schopen zpracovat požadavky transakcí a nastane stav zahlcení, všechny TPIPEs do partnera IMS jsou pozastaveny. Po oznámení od partnera IMS , že se ulevilo od povodní nebo varování před povodněmi, produkt IBM MQ obnoví všechny pozastavené TPIPEs, je-li to vhodné, a postupně zvyšuje rychlost odesílání požadavků transakcí, dokud nebude dosaženo maximální rychlosti. Zprávy konzoly jsou vydávány produktem IBM MQ jako odezva na změnu stavu partnerů IMS .

Pokud se používají partneři IMS v10 , měli byste zajistit, aby byla použita oprava PTF UK45082 .

## Odeslání IMS transakcí z IBM MQ

Chcete-li odeslat transakci IMS , která používá most, aplikace umístí zprávy do fronty produktu IBM MQ jako obvykle. Zprávy obsahují transakční data produktu IMS , mohou mít záhlaví IMS (strukturu MQIIH) nebo umožňují mostu IBM MQ - IMS vytvářet hypotézy o datech v této zprávě.

IBM MQ potom vloží zprávu do fronty IMS (je nejprve zařazena do fronty v produktu IBM MQ , aby bylo možné použít synchronizační body k zajištění integrity dat). Třída ukládání fronty IBM MQ určuje, zda je fronta *frontou OTMA* (tj. fronta používaná k přenosu zpráv do mostu IBM MQ - IMS ) a příslušného partnera IMS , do kterého jsou data zprávy odeslána.

Vzdálení správci front mohou také spustit transakce IMS zápisem do těchto front OTMA na serveru IBM MQ for z/OS.

Data vrácená ze systému IMS se zapisují přímo do fronty pro odpověď IBM MQ na frontu zadanou ve struktuře deskriptoru zpráv (MQMD). (Může se jednat o přenosovou frontu na správce front určeného v poli *ReplyToQMgr* deskriptoru MQMD.)

### Související pojmy

[Aplikace mostu IMS a IMS v systému IBM MQ for z/OS](#)

### Související úlohy

[Přizpůsobení mostu IMS](#)

### Související odkazy

[“IBM MQ a IMS” na stránce 273](#)

Toto téma vám pomůže pochopit, jak produkt IBM MQ pracuje s produktem IMS. Adaptér IMS umožňuje připojit správce front k produktu IMSa umožňuje aplikacím produktu IMS používat rozhraní MQI.

z/OS

## IBM MQ a adaptéry dávky, TSO a RRS produktu z/OS

Toto téma vám pomůže pochopit, jak produkt IBM MQ pracuje s adaptéry z/OS Batch, TSO a RRS.

### Úvod do dávkových adaptérů

Adaptéry Batch/TSO jsou rozhraní mezi aplikačními programy IBM MQ a z/OS spuštěnými pod JES, TSO nebo z/OS UNIX System Services. Tyto adaptéry umožňují aplikačním programům produktu z/OS používat rozhraní MQI.

Adaptéry poskytují přístup k prostředkům produktu IBM MQ pro programy spuštěné v následujících režimech nebo stavech:

- Režim úlohy (TCB)
- Stav problému nebo supervizora
- Režim non-cross-memory
- Režim registru bez přístupu

Spojení mezi aplikačními programy a 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 provedené v prostředcích vlastněných produktem IBM MQ. Nepodporuje protokoly vícefázového potvrzování. Adaptér RRS umožňuje aplikacím produktu IBM MQ podílet se na protokolech dvoufázového potvrzování s ostatními produkty s podporou RRS, které jsou koordinovány službami z/OS Resource Recovery Services (RRS).

Adaptéry používají službu z/OS STIMERM k naplánování asynchronní události každou sekundu. Tato událost spustí blok požadavků přerušení (IRB), který neobsahuje žádné čekání na úlohu dávkové aplikace. This IRB checks to see if the IBM MQ termination ECB has been posted. V případě, že byla ECB vyslána ECB, vysílají IRB všechny aplikační podniky ECB, které čekají na událost v produktu IBM MQ (například signál nebo čekání).

## Adaptér Batch/TSO

Adaptér IBM MQ Batch/TSO poskytuje podporu produktu IBM MQ pro aplikace z/OS Batch a TSO. Všechny aplikační programy, které jsou spuštěny v produktu z/OS Batch nebo TSO, musí mít s sebou linkový program CSQBSTUB pro rozhraní API CSQBSTUB. Stub poskytuje aplikaci přístup ke všem voláním MQI. Jednofázové potvrzení a odvolání pro aplikace se používá při zadání volání MQI **MQCMIT** a **MQBACK**.

## Adaptér RRS

Služby RRS (*Resource Recovery Services*) je dílčí komponenta produktu z/OS, která poskytuje celosystémové služby pro koordinaci dvoufázového potvrzování mezi produkty z/OS. Adaptér IBM MQ Batch/TSO RRS (adaptér RRS) poskytuje podporu produktu IBM MQ 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 RRS (například Db2).

Adaptér RRS poskytuje dva stuby. Aplikační programy pro úpravu odkazů, které chtějí používat RRS s jedním z těchto stubů, musíte propojit.

### CSQBRSTB

Tento stub vám umožňuje používat dvoufázové potvrzování a odvolání pro aplikace pomocí služeb zotavení prostředku RRS volatelných místo volání MQI **MQCMIT** a **MQBACK**.

Musíte také propojit-upravit modul ATRSCSS z knihovny SYS1.CSSLIB s vaší aplikací. Použijete-li volání MQI **MQCMIT** a **MQBACK**, obdržíte návratový kód MQRC\_ENVIRONMENT\_ERROR.

### CSQBRRSI

Tento stub umožňuje použít volání MQI **MQCMIT** a **MQBACK**; Produkt IBM MQ ve skutečnosti implementuje tato volání jako volání služby **SRRCMIT** a **SRRBACK** RRS.

Informace o sestavování aplikačních programů, které používají adaptér RRS, najdete v tématu [Adaptér dávky RRS](#).

## Kde najít další informace o adaptérech z/OS Batch, TSO a RRS

Další informace o tématech v této sekci naleznete v následujících zdrojích:

| Tabulka 24. Kde najít další informace o použití dávky z/OS s IBM MQ |                                                                     |
|---------------------------------------------------------------------|---------------------------------------------------------------------|
| Téma                                                                | Kde hledat                                                          |
| Nastavení dávkových adaptérů                                        | <a href="#">Úloha 19: Nastavení adaptérů dávek, TSO a RRS</a>       |
| Služby zotavení RRS volatelných prostředků                          | <i>Programování MVS: Callable Services for High Level Languages</i> |

z/OS

## IBM MQ for z/OS a WebSphere Application Server

Toto téma vám pomůže porozumět použití IBM MQ for z/OS u WebSphere Application Server.

Aplikace napsané v Java, které jsou spuštěny pod WebSphere Application Server, mohou použít specifikaci Java Messaging Service (JMS) k provádění systému zpráv. Systém zpráv typu point-to-point v tomto prostředí může být poskytován správcem front IBM MQ for z/OS.

Výhodou použití správce front produktu IBM MQ for z/OS pro poskytnutí systému zpráv je to, že připojení aplikací produktu JMS se může plně podílet na funkčnosti sítě IBM MQ. Mohou například používat most IMS nebo si vyměňovat zprávy se správcem front spuštěnými na jiných platformách.

## Spojení mezi produktem WebSphere Application Server a správcem front

Pro objekt továrny připojení fronty můžete vybrat buď volbu *přenos klienta* , nebo *přenos vazeb* . Zvolíte-li přenos vazeb, musí oba existovat ve stejném obrazu produktu z/OS a musí existovat správce front WebSphere Application Server a správce front.

Všimněte si, že pro *přenos vazeb* potřebujete nativní knihovny.

Oba typy podpory připojení podporují transakční aplikace: přenos klienta pomocí protokolů XA, přenos vazeb pomocí stubu WebSphere Application Server , CSQBWSTB, který používá služby RRS.

Další informace o konfiguraci továren připojení fronty viz téma *IBM MQ Použití produktu Java* .

## Použití funkcí produktu IBM MQ z aplikací produktu JMS

Zprávy produktu JMS ve frontách produktu IBM MQ standardně používají záhlaví MQRFH2 k uchování některých informací záhlaví zprávy produktu JMS . Mnoho starších aplikací produktu IBM MQ nemůže zpracovávat zprávy s těmito záhlavími a vyžadovat jejich vlastní záhlaví charakteristik, například MQCIH pro most CICS Bridge nebo MQWIH pro aplikace IBM MQ Workflow. Další podrobnosti o těchto speciálních aspektech viz [Mapování zpráv produktu JMS na zprávy produktu IBM MQ](#) .

## Managed File Transfer

---

Managed File Transfer přenáší soubory mezi systémy ve spravovaném a kontrolovatelném způsobu, 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. Managed File Transfer eliminuje nákladné redundance, snižuje náklady na údržbu a maximalizuje vaše stávající investice do IT.

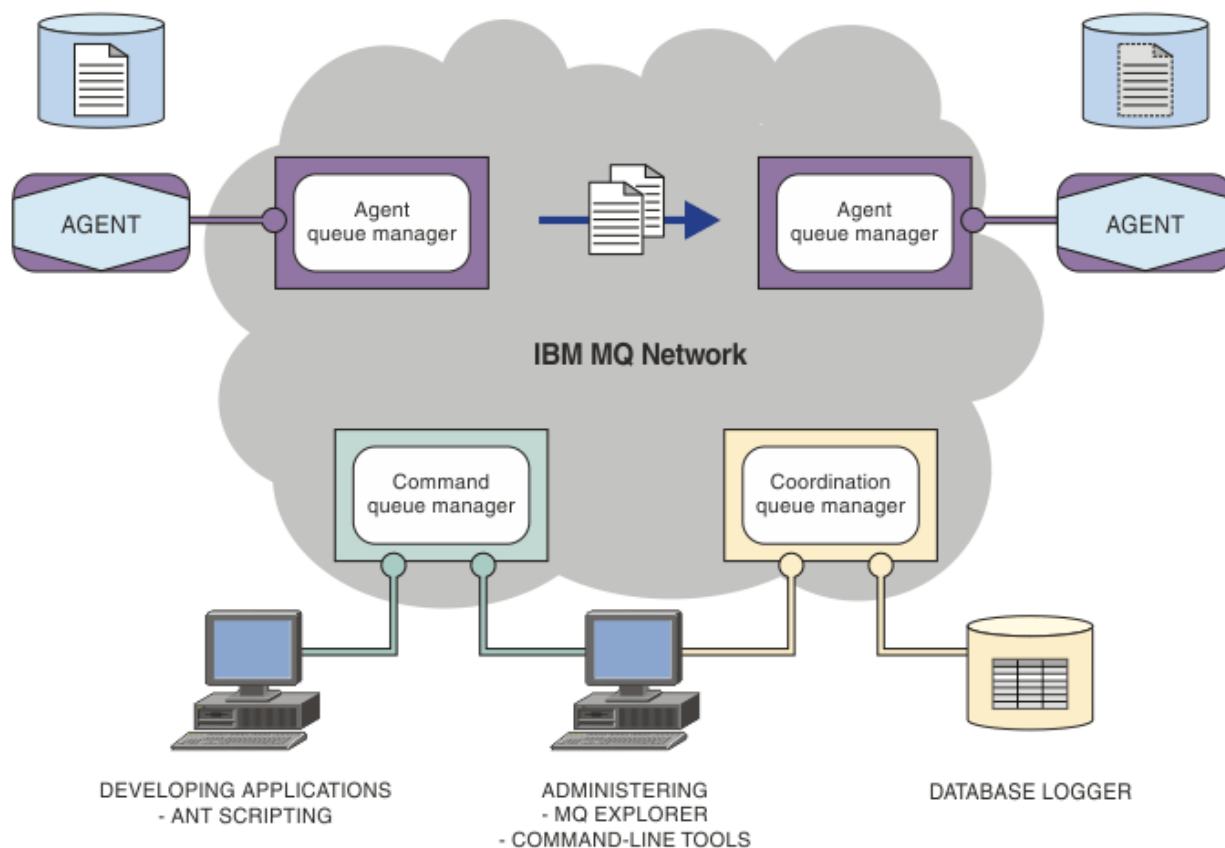



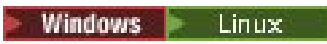
Diagram zobrazuje jednoduchou topologii produktu Managed File Transfer . K dispozici jsou dva agenti, každý se připojí ke svému vlastnímu správci front agenta v síti IBM MQ . Soubor se přenáší z agenta na jedné straně diagramu, přes síť IBM MQ , na agenta na druhé straně diagramu. Také v síti IBM MQ jsou koordinačním správcem front a správcem front příkazů. Aplikace a nástroje se připojují k těmto správcům front při konfiguraci, správě, provozu a protokolování aktivit Managed File Transfer v síti IBM MQ .

Produkt Managed File Transfer může být instalován 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 naleznete v tématu [Volby produktu Managed File Transfer](#) .

Produkt Managed File Transfer můžete použít k provádění následujících úloh:

- Vytvořit spravované přenosy souborů
  - Windows Linux Vytvořte nové přenosy souborů z produktu 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 .
  - Zapisovat aplikace, které řídí Managed File Transfer , tím, že vkládají zprávy do front příkazů agenta.
  - Naplánovat přenosy souborů, které se mají provést později. Naplánované přenosy souborů můžete také spouštět na základě rozsahu událostí systému souborů, například nového vytvářeného souboru.
  - Nepřetržité monitorování prostředku, například adresář, a když obsah tohoto prostředku splňuje některé předdefinované podmínky, spusťte úlohu. Tato úloha může být přenos souborů, skript Ant nebo úloha JCL.
  - Přenos souborů do a z front produktu IBM MQ .
  - Přeneste soubory na servery FTP, FTPS nebo SFTP a z nich.



- Přenést soubory do a z uzlů Connect:Direct .
- Přenos jak textových, tak binárních souborů. Textové soubory jsou automaticky převáděny mezi kódovými stránkami a konvencemi na konci řádku zdrojového a cílového systému.
- Přenosy mohou být zabezpečeny pomocí oborových standardů pro připojení založená na SSL (Secure Socket Layer).
- Zobrazení přenosů v průběhu a protokolování informací o všech převodech ve vaší síti
  -  Zobrazujte stav přenosů v průběhu IBM MQ Explorer na platformách Linux nebo Windows .
  -  Zkontrolujte stav dokončených přenosů pomocí produktu IBM MQ Explorer na platformách Linux nebo Windows .
  - Pomocí funkce modulu protokolování databáze produktu Managed File Transfer můžete uložit zprávy protokolu do databáze produktu Db2 nebo Oracle .

Produkt Managed File Transfer je postaven na produktu IBM MQ, který zajišťuje zajištěný přenos zpráv mezi aplikacemi pouze pro doručování zpráv. Můžete využít výhody různých funkcí produktu IBM MQ. Kompresi kanálu můžete například použít ke komprimování dat posílané mezi agenty pomocí kanálů produktu IBM MQ a k zabezpečení dat, která odesíláte mezi agenty, pomocí kanálů SSL. Soubory se přenášejí spolehlivě a mohou tolerovat selhání infrastruktury, přes kterou se přenos souboru provádí. Pokud dojde k výpadku sítě, přenos souborů se po obnovení konektivity znovu spustí, jakmile se obnoví.

Díky konsolidaci přenosu souborů s vaší stávající sítí IBM MQ se můžete vyhnout výdajům nutným pro údržbu dvou samostatných infrastruktur. Pokud již nejste zákazníkem produktu IBM MQ vytvořením sítě produktu IBM MQ pro podporu produktu Managed File Transfer , sestavujete páteřní síť pro budoucí implementaci architektury SOA. Jste-li již zákazníkem produktu IBM MQ , produkt Managed File Transfer může využít existující infrastruktury produktu IBM MQ včetně produktů IBM MQ Internet Pass-Thru a IBM Integration Bus.

Řešení vysoké dostupnosti systému IBM MQ můžete využít ke zlepšení odolnosti konfigurace produktu Managed File Transfer . Pokud agenti používají replikované správce datových front (RQMs), musíte je nakonfigurovat, aby mohli používat funkci plovoucí adresy IP. To znamená, že agenti používají stejnou adresu IP ke komunikaci s každou ze tří instancí RDR, které momentálně běží a automaticky se znovu připojí k překonání selhání (viz [Vysoká dostupnost RDQM](#) a [Vytvoření a odstranění plovoucí adresy IP](#)). Pokud použijete řešení správce front s více instancemi, pak aplikace používají jinou adresu IP ke komunikaci s každou instancí, kterou obsluhuje opětovně navázaní spojení s klientem při překonání selhání (viz [Správci front s více instancemi](#) a [Připojení kanálu a klienta](#)).

Produkt Managed File Transfer integruje s řadou dalších produktů IBM :

### **IBM Integration Bus**

Zpracovat soubory, které byly přeneseny systémem Managed File Transfer jako součást toku IBM Integration Bus . Další informace naleznete v tématu [Práce s MFT z IBM Integration Bus](#).

### **IBM Sterling Connect:Direct**

Přenos souborů do existující sítě Connect:Direct a z ní pomocí mostu Managed File Transfer Connect:Direct . Další informace naleznete v tématu [Most produktu 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 282](#)

[Přehled, jak jsou agenti Managed File Transfer připojeni ke koordinačnímu správci front v síti IBM MQ .](#)

[“Jak MFT pracuje s IBM MQ?” na stránce 282](#)

Produkt Managed File Transfer interaktivně spolupracuje s produktem IBM MQ.

## Jak MFT pracuje s IBM MQ?

Produkt Managed File Transfer interaktivně spolupracuje s produktem IBM MQ.

- Managed File Transfer přenáší soubory mezi procesy agentů tak, že každý soubor rozdělí do jedné nebo více zpráv a přenáší zprávy prostřednictvím sítě IBM MQ .
- Procesy agentů přesouvají data souboru pomocí přechodných zpráv a minimalizují tak dopad na protokoly produktu IBM MQ . Tím, že komunikuje s jedním jiným procesem agenta, reguluje tok zpráv obsahujících data souboru. Tím zabráníte zprávám obsahujícím data souboru v přenosových frontách produktu IBM MQ a zajistíte, že pokud nebude doručena žádná z přechodných zpráv, budou data souboru odeslána znovu.
- Agenti produktu Managed File Transfer používají počet front produktu IBM MQ . Další informace naleznete v tématu [Systémové fronty MFT a téma systému](#).
- Přestože některé z těchto front jsou striktně pro vnitřní použití, agent může přijímat požadavky ve formě speciálně formátovaných příkazových zpráv odeslaných do určité fronty, ze které agent čte. Oba příkazy příkazového řádku a modul plug-in IBM MQ Explorer odesílají do agenta zprávy IBM MQ , aby dali agentovi pokyn k provedení požadované akce. S tímto způsobem můžete psát aplikace produktu IBM MQ , které s tímto agentem spolupracují. Další informace naleznete v tématu [Řízení MFT pomocí vkládání zpráv do fronty příkazů agenta](#).
- Agenti produktu Managed File Transfer odesílají informace o svém stavu a průběhu a výsledku přenosu do správce front MQ , který byl označen jako koordinační správce front. Tyto informace jsou publikovány koordinačním správcem front a mohou být odebírány aplikacemi, které chtějí monitorovat průběh přenosu nebo uchovávat záznamy o provedených přenosech. Jak příkazy příkazového řádku, tak modul plug-in produktu IBM MQ Explorer mohou použít informace, které jsou publikovány. Můžete psát aplikace IBM MQ , které používají tyto informace. Další informace o tématu, na které jsou publikovány informace, najdete v tématu [SYSTEM.FTE téma](#).
- Klíčové komponenty produktu Managed File Transfer využívají možnosti správců front produktu IBM MQ ukládat a předávat zprávy. To znamená, že pokud dojde k výpadku, nebude mít žádný dopad na části vaší infrastruktury i nadále přenášet soubory. To je rozšířeno na koordinačního správce front, kde kombinace uložení a předání a trvalých odběrů umožňuje koordinačnímu správci front tolerovat, že přestane být k dispozici, aniž by došlo ke ztrátě klíčových informací o provedených přenosech souborů.

## MFT přehled topologie

Přehled, jak jsou agenti Managed File Transfer připojeni ke koordinačnímu správci front v síti IBM MQ .

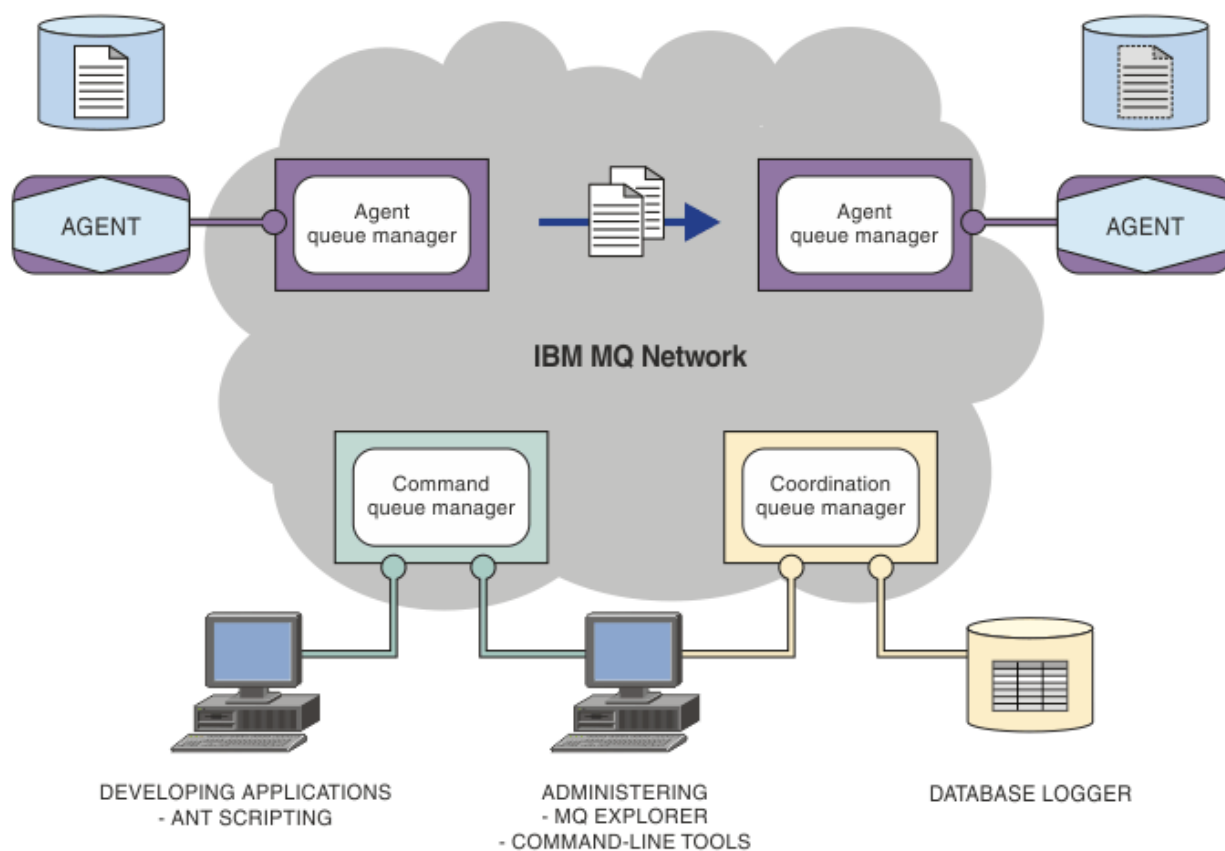
Agenti produktu Managed File Transfer odesílají a přijímají soubory, které jsou přeneseny. Každý agent má svou vlastní sadu front v přidruženém správci front a agent je připojen ke svému správci front buď v rámci vazeb, nebo v režimu klienta. Agent může také používat koordinačního správce front jako svého správce front.

Koordinační správce front vysílá audit a informace o přenosu souborů. Koordinační správce front představuje jeden bod pro shromažďování agentů, stav přenosu a informace o auditu přenosu. Koordinační správce front není nutný k tomu, aby byl k dispozici, aby mohly být přenosy umístěny. Pokud se koordinační správce front dočasně stane nedostupným, budou přenosy pokračovat jako normální. Zprávy auditu a stavu jsou uloženy ve správcích front agentů, dokud nebude dostupný koordinační správce front, a poté jej lze zpracovat jako normální.

Agenti se registrují s koordinačním správcem front a publikují jejich podrobnosti do tohoto správce front. Tyto informace o agentovi používá modul plug-in produktu Managed File Transfer k povolení zahájení přenosů z produktu IBM MQ Explorer. Informace o agentovi shromážděné v koordinačním správci front se také používají příkazy k zobrazení informací o agentovi a stavu agenta.

Stav přenosu a informace o přenosu auditu jsou publikovány v koordinačním správci front. Informace o stavu přenosu a informace o auditu přenosu používá modul plug-in produktu Managed File Transfer k monitorování postupu přenosů ze serveru IBM MQ Explorer. Informace o auditu přenosu uložené v koordinačním správci front mohou být zachovány, aby poskytovaly auditovatelnost.

Správce front příkazů se používá pro připojení k síti produktu IBM MQ a ke kterému je správce front připojen, když zadáte příkazy Managed File Transfer .



### Související pojmy

[“Managed File Transfer” na stránce 279](#)

Managed File Transfer přenáší soubory mezi systémy ve spravovaném a kontrolovatelném způsobu, bez ohledu na velikost souboru nebo použité operační systémy.

[“Jak MFT pracuje s IBM MQ?” na stránce 282](#)

Produkt Managed File Transfer interaktivně spolupracuje s produktem IBM MQ.

### Související odkazy

[Managed File Transfer scénář](#)

## Přehled produktu MFTREST API

Produkt REST API podporuje určité příkazy Managed File Transfer , včetně výpisů převodů a podrobností o agentech přenosu souborů.

V produktu IBM MQ 9.1.0 produkt REST API obsahuje volby pro výpis všech aktuálních Managed File Transfer přenosů a dotazování na stav agentů Managed File Transfer . Další informace najdete v tématu [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 přes internet.

**V 9.2.0** Z produktu IBM MQ 9.2.0 je MQIPT volitelná komponenta produktu IBM MQ. Chcete-li získat instalační soubory produktu MQIPT pro produkt IBM MQ 9.2.x, přejděte na adresu [IBM Fix Central for IBM MQ](#). Před IBM MQ 9.2.0 byl produkt MQIPT dostupný jako balík podpory.

Nemusíte spouštět produkt IBM MQ 9.2.0 pro použití produktu MQIPT v produktu IBM MQ 9.2.0 nebo novějším. Produkt MQIPT můžete použít k připojení libovolné podporované verze produktu IBM MQa nemusíte instalovat žádné jiné komponenty produktu IBM MQ na stejnou verzi jako produkt MQIPT.

Pokud jste zakoupili oprávnění k produktu IBM MQ , můžete instalovat tolik kopií, kolik je potřeba pro produkt MQIPT. Instalace produktu MQIPT se nezapočítávají do zakoupeného nároku produktu IBM MQ . Další informace o licencování IBM MQ naleznete v [IBM MQ licenčních informacích](#).

**Poznámka:** Tato dokumentace se vztahuje k produktu MQIPT v produktu IBM MQ 9.2.0 a novějším. Dokumentaci podpory produktu MQIPT (verze 2.1) v produktu IBM Documentation naleznete v dokumentu [MQIPT \(SupportPac MS81\)](#) v dokumentaci produktu IBM MQ 9.0 .

**Poznámka:** Pokud používáte produkt MQIPT 2.1 nebo starší, doporučujeme upgradovat na produkt MQIPT for IBM MQ 9.2.x, protože datum ukončení podpory pro balík podpory produktu MQIPT je 30th .

IBM MQ Internet Pass-Thru se spouští jako samostatná služba, která může přijímat a předávat toky zpráv produktu IBM MQ mezi dvěma správci front IBM MQ nebo mezi klientem IBM MQ a správcem front IBM MQ .

Produkt MQIPT umožňuje toto připojení, pokud klient a server nejsou ve stejné fyzické síti.

Jednu nebo více instancí produktu MQIPT lze umístit do cesty komunikace mezi dvěma správci front produktu IBM MQ nebo mezi klientem IBM MQ a správcem front IBM MQ . Instance produktu 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.

MQIPT naslouchá na jednom nebo více portech TCP/IP pro přichozí připojení, která mohou přenášet buď normální zprávy IBM MQ , IBM MQ zpráv tunelovaných uvnitř HTTP nebo zprávy zašifrované pomocí TLS (Transport Layer Security) nebo SSL (Secure Sockets Layer). Produkt MQIPT může zpracovávat více souběžných připojení.

Kanál IBM MQ , který činí počáteční požadavek na připojení TCP/IP, se nazývá *volající*, kanál, k němuž se pokouší připojit jako *odpovídací modul*, a správce front, kterého se nakonec pokouší kontaktovat jako *správce cílové fronty*.

Produkt MQIPT uchovává data v paměti, protože ji předává ze zdroje do místa určení. Na disku se neukládají žádná data (kromě paměti stránkované na disk operačním systémem). Jediný okamžik, kdy produkt MQIPT přistupuje k disku explicitně, je číst jeho konfigurační soubor a zapisovat protokol připojení a trasovací záznamy.

Úplný rozsah typů kanálů produktu IBM MQ lze provést prostřednictvím jedné nebo více instancí produktu MQIPT. Přítomnost MQIPT v komunikační cestě nemá žádný vliv na funkční charakteristiky připojených komponent IBM MQ , ale mohlo by to mít nějaký vliv na výkon přenosu zpráv.

MQIPT lze použít ve spojení s IBM MQ a IBM Integration Bus, jak je popsáno v [“Možné konfigurace produktu MQIPT”](#) na stránce 287.

Informace o instalaci produktu MQIPT naleznete v tématu [Instalace produktu MQIPT](#).

### **Související úlohy**

[Konfigurace produktu IBM MQ Internet Pass-Thru](#)

[Administrace a konfigurace produktu IBM MQ Internet Pass-Thru](#)

### **Související odkazy**

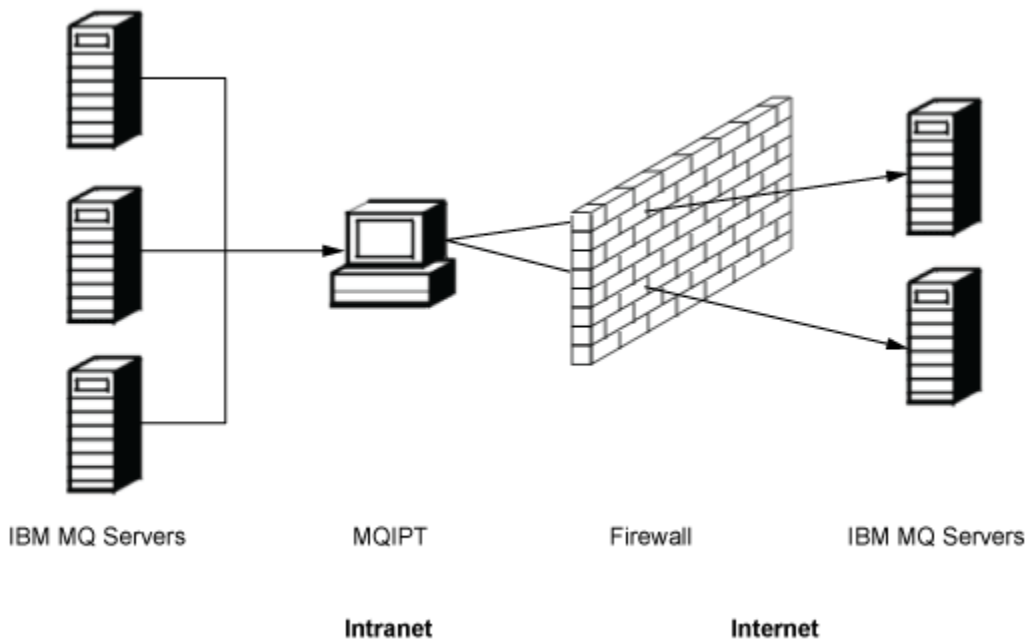
[Odkaz konfigurace produktu IBM MQ Internet Pass-Thru](#)

## **Použití MQIPT**

Pro IBM MQ Internet Pass-Thru (MQIPT) existuje celá řada možných použití.

### **MQIPT lze použít jako koncentrátor kanálů**

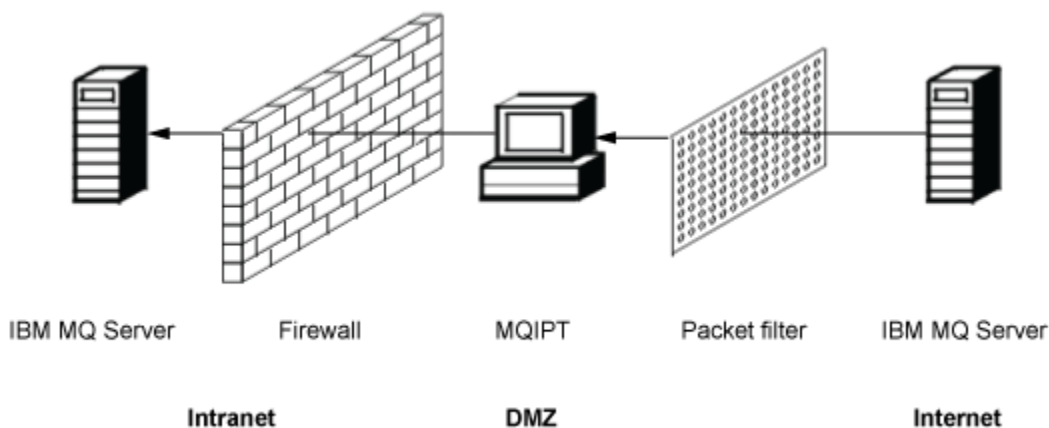
Díky použití produktu MQIPT se mohou kanály pro více samostatných hostitelů nebo z více samostatných hostitelů zobrazit na bráně firewall, jako kdyby byly všechny na hostiteli MQIPT nebo naopak z ní. Díky tomu je snazší definovat a spravovat pravidla filtrování brány firewall.



Obrázek 80. Příklad MQIPT jako koncentrátoru kanálů

### Produkt MQIPT lze umístit do zóny DMZ za účelem poskytnutí jediného místa přístupu.

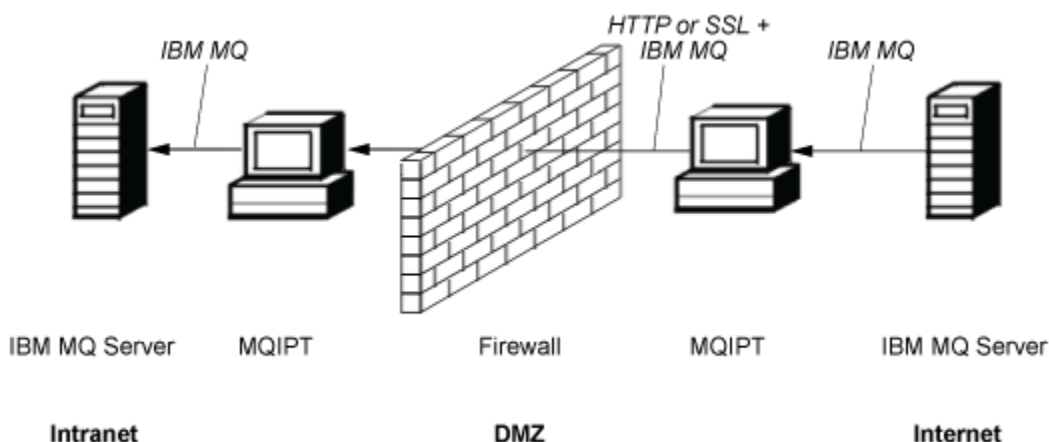
Je-li produkt MQIPT umístěn v rámci ochranné bariéry DMZ (konfigurace brány firewall pro zabezpečení lokálních sítí) na počítači se známým a důvěryhodným protokolem IP (IP), lze produkt MQIPT použít k naslouchání příchozím připojením kanálu produktu IBM MQ, které může poté předat důvěryhodnému intranetu; vnitřní brána firewall musí tomuto důvěryhodnému počítači povolit, aby mohla provádět příchozí připojení. V této konfiguraci produkt MQIPT zabrání externím požadavkům na přístup z příjmu skutečných adres IP počítačů v důvěryhodném intranetu. Tímto způsobem MQIPT poskytuje jednotný přístupový bod.



Obrázek 81. Příklad produktu MQIPT v bráně firewall DMZ

## MQIPT může komunikovat pomocí tunelového propojení HTTP

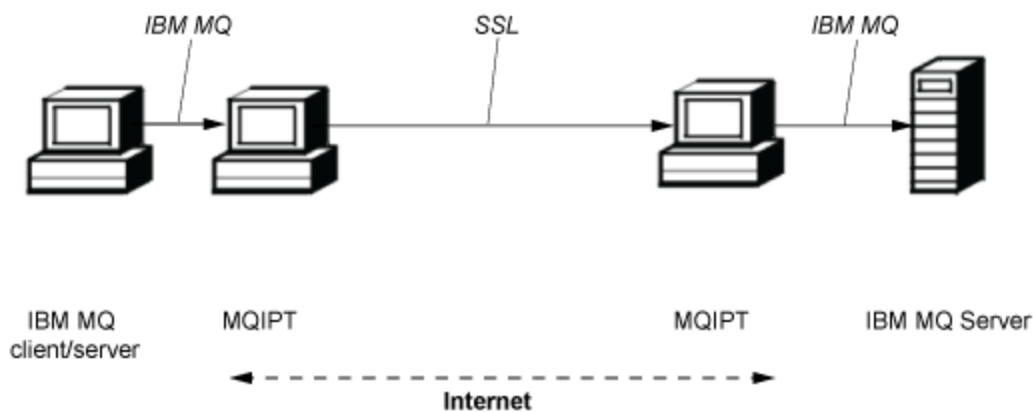
Pokud jsou ve frontě implementovány dvě instance produktu MQIPT, mohou komunikovat pomocí protokolu HTTP. Funkce tunelového propojení HTTP umožňuje předávat požadavky přenášené prostřednictvím bran firewall prostřednictvím existujících serverů proxy HTTP. První MQIPT vloží protokol IBM MQ do HTTP a druhý extrahuje protokol IBM MQ z jeho obálky HTTP a předá jej cílovému správci front.



Obrázek 82. Příklad tunelového propojení MQIPT a HTTP

## Produkt MQIPT může šifrovat zprávy

Je-li portál MQIPT nakonfigurován jako v předchozím příkladu, mohou být před přenosem prostřednictvím bran firewall šifrovány požadavky. První MQIPT zašifruje data a druhý dešifruje je pomocí SSL/TLS před odesláním do cílového správce front.



Obrázek 83. 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 produktu IBM MQ.

Je-li přijat správně vytvořený požadavek, produkt MQIPT vytvoří další připojení TCP/IP mezi sebou a cílovým správcem front IBM MQ. Poté předává všechny pakety protokolu, které přijímá od svého

příchozího připojení k cílovému správci front, a vrátí pakety protokolu z cílového správce front zpět do původního příchozího připojení.

Nejsou zahrnuty žádné změny protokolu IBM MQ (klient/server nebo správce front), protože ani jedna z nich není přímo informována o přítomnosti prostředníka. Nové verze kódu klienta nebo serveru IBM MQ se nepožadují.

Chcete-li použít produkt MQIPT, volající kanál musí být konfigurován tak, aby používal název hostitele a port produktu MQIPT, nikoli název hostitele a port cílového správce front. Tento parametr je definován vlastností **CONNNAME** kanálu IBM MQ. Produkt MQIPT načte příchozí data a jednoduše je předá do cílového správce front. Jiná pole konfigurace, jako je například ID uživatele a heslo v kanálu klienta/serveru, jsou podobně předána do cílového správce front.

## Více správců front

MQIPT lze použít k povolení přístupu do více než jednoho cílového správce front. Má-li to fungovat, musí existovat mechanismus, který říká MQIPT, ke kterému se má správce front připojit, takže MQIPT používá příchozí číslo portu TCP/IP, aby určil, 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 správce cílové fronty pomocí MQIPT *route*. Můžete definovat až 100 takových přenosových cest, které přidruží naslouchající port TCP/IP s názvem hostitele a portem cílového správce front. To znamená, že název hostitele (adresa IP) cílového správce front nebude nikdy viditelný pro původní kanál. Každá cesta může obsluhovat více spojení mezi svým naslouchajícím portem a místem určení, přičemž každé z nich pracuje nezávisle.

## MQIPT konfigurační soubor

Produkt MQIPT používá konfigurační soubor s názvem `mqipt.conf`. Tento soubor obsahuje definice všech přenosových cest a jejich přidružených vlastností. Další informace o produktu `mqipt.conf` naleznete v tématu [Administrace a konfigurace produktu IBM MQ Internet Pass-Thru](#).

Když je příkaz MQIPT spuštěn, spustí každou přenosovou cestu, která je uvedena v konfiguračním souboru. Zprávy se zapisují na systémovou konzolu s informací o stavu každé přenosové cesty. Když se zobrazí zpráva MQCPI078 pro přenosovou cestu, je tato trasa připravena přijímat požadavky na připojení.

## Možné konfigurace produktu MQIPT

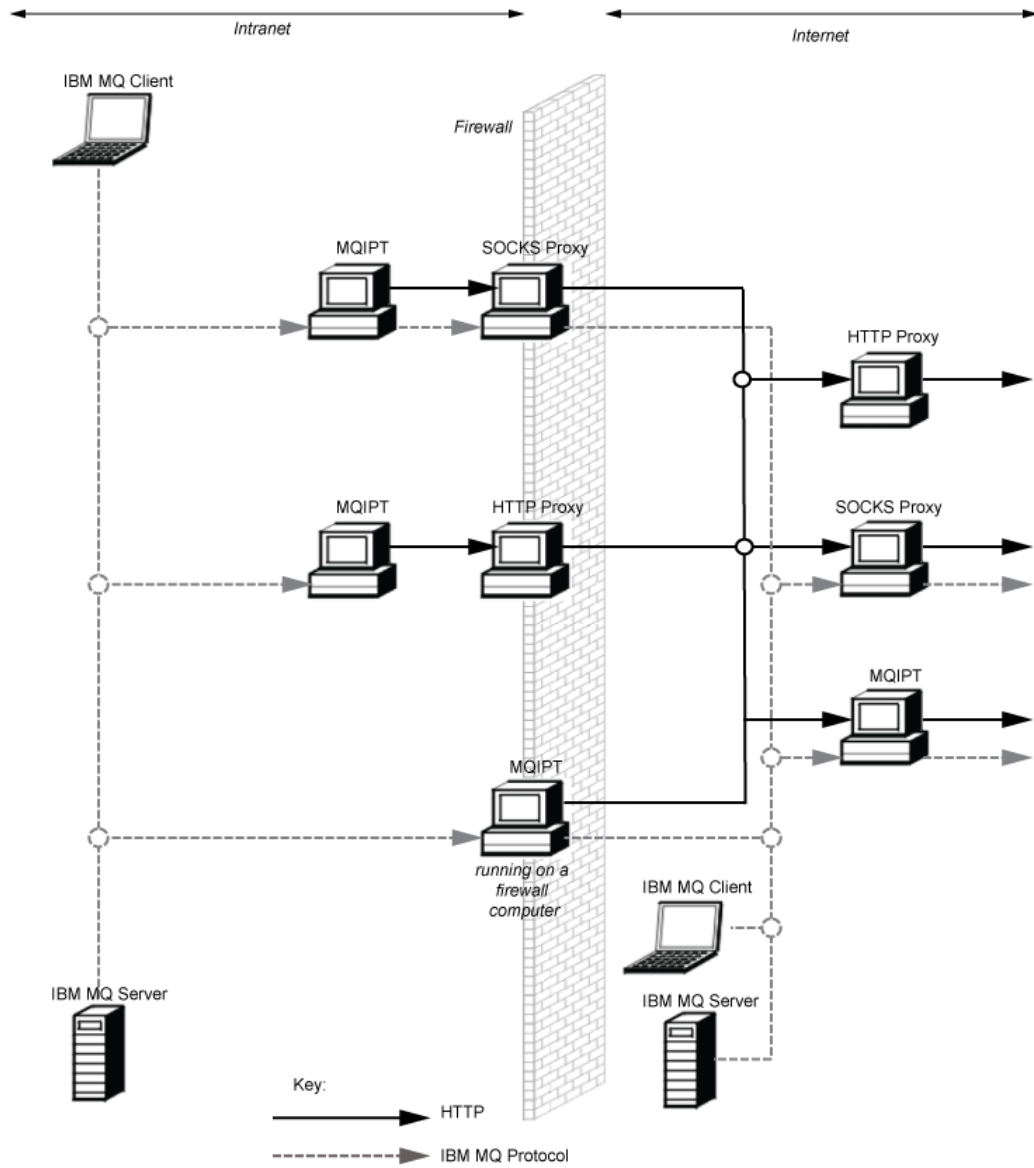
MQIPT lze použít ve spojení s IBM MQ a IBM Integration Bus.

Následující vícedílné číslo zobrazuje mnoho možných konfigurací pro MQIPT v topologii produktu IBM MQ. Tato zpráva ilustruje různé způsoby odeslání zpráv produktu MQIPT. Zobrazuje klienty a servery na intranetu, uvnitř firewallu a na Internetu mimo bránu firewall, předávat zprávy serveru MQIPT, HTTP proxy nebo SOCKS proxy, které je předává.

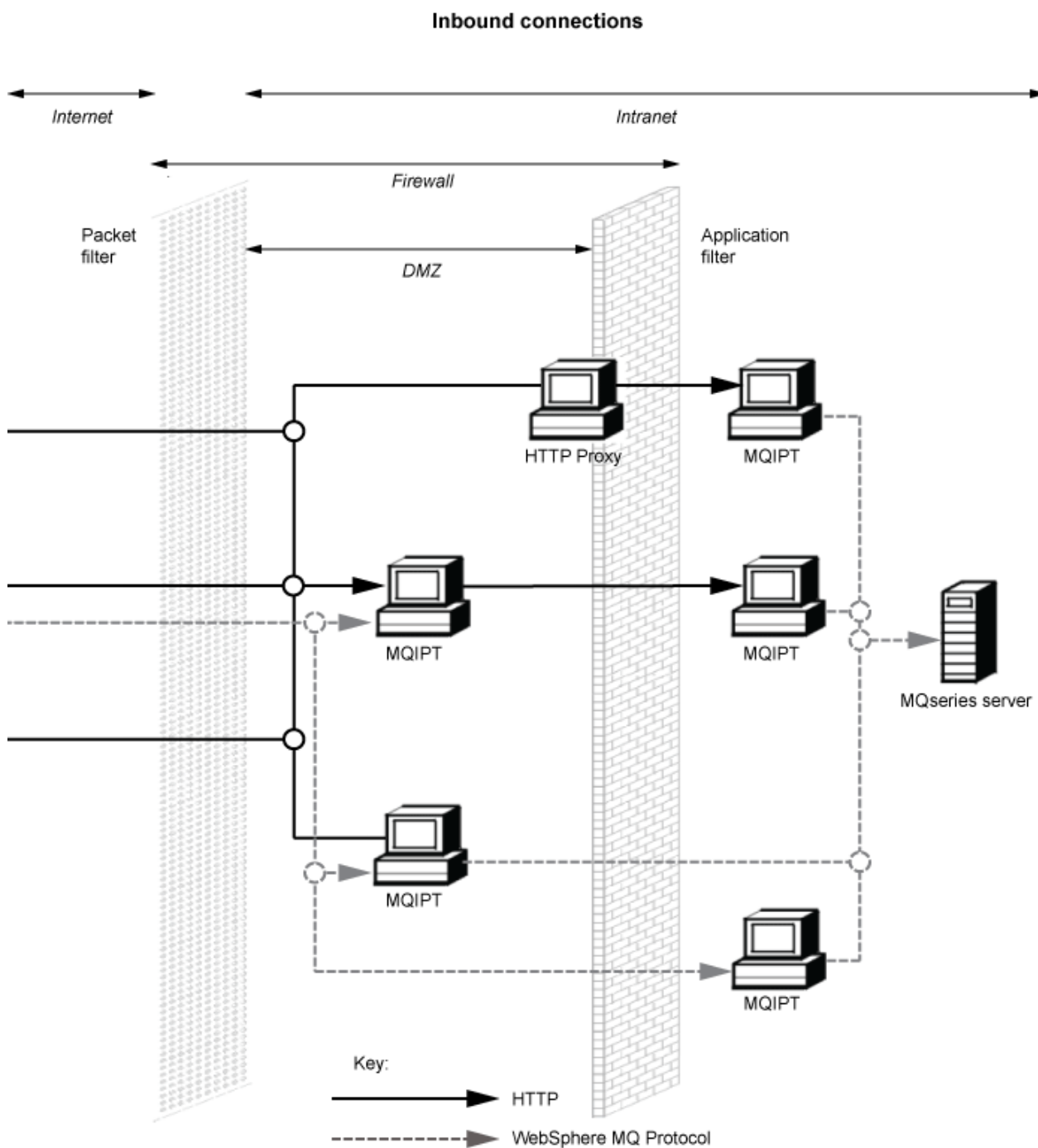
Zprávy jsou přijímány serverem proxy MQIPT nebo proxy HTTP v zóně DMZ před předáním zprávy prostřednictvím příchozí brány firewall na server.

Všimněte si, že počítače proxy HTTP, SOCKS a MQIPT na intranetové straně brány firewall představují možnost více počítačů zřetězených najednou na internetu. Například, počítač MQIPT může komunikovat přes jeden nebo více serverů proxy SOCKS nebo HTTP nebo dále MQIPT počítačů, než dosáhne svého cíle.

### Outbound connections







## Kompatibilní konfigurace

Kompatibilní scénáře připojení, ve kterých klient IBM MQ nebo správce front komunikuje s produktem MQIPT. Ke komunikaci s cílovým správcem front se používá stejná nebo druhá cesta MQIPT.

## Kompatibilní konfigurace s jednoduchou cestou MQIPT

Pro komunikaci s IBM MQ můžete použít jedinou přenosovou cestu MQIPT .

Sloupce v produktu [Tabulka 25](#) na stránce 290 obsahují následující informace:

1. Protokol používaný mezi IBM MQ a trasou MQIPT . Připojení může být vytvořeno buď klientem IBM MQ , nebo správcem front a může použít buď formát IBM MQ Formats and Protocols (FAP) nebo protokol SSL/TLS.
2. Režim, ve kterém trasa MQIPT funguje. Formát komunikace přes Internet mezi MQIPT a IBM MQ je určen konfigurací cesty MQIPT . Všimněte si, že v případě, že tabulka zmiňuje zabezpečení SSL, můžete také použít TLS.
3. Protokol použitý mezi trasou MQIPT a cílovým správcem front.

| <i>Tabulka 25. Platné konfigurace s jednou instancí produktu MQIPT</i> |                             |                                  |
|------------------------------------------------------------------------|-----------------------------|----------------------------------|
| <b>1. Zdrojový protokol IBM MQ</b>                                     | <b>2. Režim cesty MQIPT</b> | <b>3. IBM MQ cílový protokol</b> |
| FAP                                                                    | FAP-server 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 klient SSL     | SSL/TLS                          |

## Kompatibilní konfigurace s více než jednou přenosovou cestou MQIPT

Pro komunikaci s produktem IBM MQ se můžete rozhodnout použít více než jednu trasu, na jedné nebo více instancích produktu MQIPT.

Sloupce v produktu [Tabulka 26](#) na stránce 291 obsahují následující informace:

1. Protokol použitý mezi IBM MQ a první trasou MQIPT . Připojení může být vytvořeno buď klientem IBM MQ , nebo správcem front a může použít buď formát IBM MQ Formats and Protocols (FAP) nebo protokol SSL/TLS.
2. Režim, ve kterém funguje první trasa MQIPT . Formát komunikace přes Internet mezi MQIPT a IBM MQ je určen konfigurací cesty MQIPT . Všimněte si, že v případě, že tabulka zmiňuje zabezpečení SSL, můžete také použít TLS.
3. Režim, ve kterém druhá trasa MQIPT funguje.
4. Protokol použitý mezi druhou přenosovou cestou MQIPT a správcem cílové fronty.

Tabulka 26. Platné konfigurace s více instancemi produktu MQIPT

| 1. Zdrojový protokol IBM MQ | 2. Režim první trasy MQIPT | 3. Režim druhé přenosové cesty MQIPT | 4. IBM MQ cílový protokol |
|-----------------------------|----------------------------|--------------------------------------|---------------------------|
| FAP (výchozí)               | FAP-server proxy (výchozí) | FAP-server proxy (výchozí)           | FAP                       |
|                             | FAP-server a SSL-klient    | SSL-proxy                            | SSL/TLS                   |
|                             |                            | SSL-server a FAP-klient              | FAP                       |
|                             |                            | SSL-server a klient SSL              | SSL/TLS                   |
|                             | Klient HTTP                | HTTP-server a klient SSL             | SSL/TLS                   |
|                             | HTTPS-klient               | HTTPS-server a klient SSL            | SSL/TLS                   |
|                             | Klient HTTP                | Server HTTP                          | FAP                       |
| HTTPS-klient                | HTTPS-server               | FAP                                  |                           |
| SSL/TLS                     | SSL-proxy                  | SSL-proxy                            | SSL/TLS                   |
|                             |                            | SSL-server a FAP-klient              | FAP                       |
|                             |                            | SSL-server a klient SSL              | SSL/TLS                   |
|                             | Klient HTTP                | Server HTTP                          | FAP                       |
|                             | HTTPS-klient               | HTTPS-server                         | SSL/TLS                   |
|                             | Klient HTTP                | HTTP-server a klient SSL             | FAP                       |
|                             | HTTPS-klient               | HTTPS-server a klient SSL            | SSL/TLS                   |

## Podporované konfigurace kanálů

Všechny typy kanálů produktu IBM MQ jsou podporovány, ale konfigurace je omezena na připojení TCP/IP. Pro klienta nebo správce front produktu IBM MQ se produkt MQIPT jeví, jako by se jedná o správce cílové fronty. V případě, že konfigurace kanálu vyžaduje cílového hostitele a číslo portu, je určen název hostitele a číslo portu modulu listener produktu MQIPT .

### Kanály klienta/serveru

MQIPT naslouchá příchozím požadavkům na připojení klienta a poté je předává buď pomocí tunelového propojení HTTP, SSL/TLS, nebo jako standardní pakety protokolu IBM MQ . Pokud MQIPT používá tunelové připojení HTTP nebo SSL/TLS, předává je při připojení k druhému MQIPT. Pokud nepoužívá tunelové připojení HTTP, předá je v souvislosti s tím, co vidí jako cílového správce front (i když to může být dále MQIPT). Když cílový správce front přijal připojení klienta, jsou pakety přenášeny mezi klientem a serverem.

### Odesílatelové/přijímací kanály klastru

Pokud produkt MQIPT přijme příchozí požadavek od odesílacího kanálu klastru, předpokládá, že správce front byl povolen SOCKS a že skutečná cílová adresa bude získána během procesu navázání komunikace přes SOCKS. Předává požadavek dalšímu správci MQIPT nebo správci cílové fronty přesně stejným způsobem jako u kanálů připojení klienta. To zahrnuje také automaticky definované kanály odesílatele klastru.

### Odesílatel/příjemce

Pokud produkt MQIPT přijme příchozí požadavek od odesílacího kanálu, předá jej dalšímu správci MQIPT nebo správci cílové fronty přesně stejným způsobem jako pro kanály připojení klienta. Cílový správce front ověří příchozí požadavek a v případě potřeby spustí přijímací kanál. Veškerá komunikace mezi odesílatelem a přijímacím kanálem (včetně toků zabezpečení) jsou předávána.

### **Žadatel/server**

Tato kombinace je zpracována stejným způsobem jako předchozí konfigurace. Ověření požadavku na připojení je prováděno kanálem serveru v cílovém správci front.

### **Žadatel/odesílatel**

Konfigurace "callback" by mohla být použita, pokud dva správci front nemají povoleno vytvářet přímé připojení k sobě navzájem, ale jsou oba povoleny k připojení k produktu MQIPT a k přijetí připojení z této konfigurace.

### **Server/žadatel a server/příjemce**

Ty jsou zpracovány produktem MQIPT stejným způsobem, jakým zpracovává konfiguraci produktu Sender/Receiver.

## **Podmínky ukončení a selhání kanálu**

Když produkt MQIPT zjistí uzavření (normální nebo abnormální) kanálu produktu IBM MQ, šíří se uzavřením kanálu. Uzavřete-li cestu pomocí produktu MQIPT, všechny kanály procházející touto cestou se uzavřou.

MQIPT poskytuje volitelný prostředek idle-timeout. Pokud obslužný program MQIPT zjistí, že kanál byl nečinný po dobu delší než časový limit, provede okamžité ukončení činnosti obou připojení.

Systémy IBM MQ na obou koncích kanálu pozorují tyto abnormální podmínky ukončení činnosti buď jako selhání sítě, nebo jako ukončení kanálu jejich partnerem. Kanál je pak schopen restartovat a obnovit (pokud k selhání dojde během období nejistoty protokolu), jako by se MQIPT nepoužíval.

## **Bezpečnost zpráv**

Distribuovaná správa fronty IBM MQ zajišťuje, že jsou zprávy řádně doručovány. To je stále případ, kdy se MQIPT nachází mezi dvěma konci kanálu. Produkt MQIPT neukládá žádná data zprávy nebo se podílí na proceduře synchronizačního bodu, která zajišťuje správné doručení zprávy.

Pokud používáte rychlé a netrvalé zprávy IBM MQ, pokud se trasa MQIPT nezdaří nebo se znovu spustí, když se během přenosu nachází zpráva IBM MQ, může dojít ke ztrátě zprávy. Před restartem přenosové cesty se ujistěte, že všechny kanály IBM MQ používající přenosovou cestu MQIPT jsou neaktivní.

Další informace o bezpečnosti zpráv v produktu IBM MQ najdete v tématu [Bezpečnost zpráv](#).

## **Správci front s více instancemi a vysoká dostupnost**

Produkt MQIPT lze použít s správci front s více instancemi v prostředích s vysokou dostupností.

MQIPT nemá žádný trvalý stav, a proto neexistuje žádná výhoda, pokud by došlo k selhání MQIPT na jiný systém. Místo toho máte více instancí MQIPT s identickými konfiguračními soubory `mqipt.conf` spuštěnými na různých systémech. Monitorujte každou instanci MQIPT pro dostupnost a restartujte ji (na stejném systému), je-li to nutné. To poskytuje sadu identických instancí MQIPT, které lze použít ke směrování připojení. Poté je nutné zajistit, aby produkt IBM MQ mohl směřovat připojení k produktu MQIPT a že produkt MQIPT může předat tato připojení k cílovému správci front.

Odchozí kanály produktu IBM MQ lze směřovat na dostupnou instanci produktu MQIPT různými způsoby, například:

- Použijte prostředek pro vyrovnávání zátěže nebo směrovač vysoké dostupnosti, jako je IBM Network Dispatcher z produktu WebSphere Edge Components.
- V definici kanálu produktu IBM MQ zadejte více názvů připojení pomocí seznamu odděleného čárkami. Produkt IBM MQ se poté pokusí o připojení ke každé adrese MQIPT, dokud nenajde dostupnou instanci produktu MQIPT.

Také musíte směřovat připojení z produktu MQIPT do cílového správce front. Pokud konfigurace vysoké dostupnosti zajišťuje překonání selhání adresy IP pomocí cílového správce front, není vyžadována žádná speciální konfigurace produktu MQIPT: uveďte cílovou adresu IP ve vlastnosti přenosové cesty **Destination** a umožněte, aby operace překonání selhání přesunovala adresu IP se správcem front.

Pokud se však adresa IP správce front změní po překonání selhání, musíte zajistit, aby produkt MQIPT přesměroval připojení na správné místo určení. To lze provést jedním z několika způsobů:

- Zapište uživatelskou proceduru pro směrování, která kontroluje, která adresa IP a číslo portu jsou přístupné, a pak přepište cíl přenosové cesty pro každé připojení. Některé vzorové uživatelské procedury směrování jsou k dispozici s produktem MQIPT; lze je pro tento účel upravit.
- Použijte prostředek pro rozložení zátěže s vysokou dostupností k přesměrování připojení.
- Definujte více tras MQIPT , jeden pro každou adresu IP a port, kde může být správce front spuštěn. Pak nasměruje připojení produktu IBM MQ k různým trasám produktu MQIPT , například tím, že vypíše všechny adresy IP přenosové cesty a čísla portů v čárkami odděleném seznamu v názvu připojení odchozího kanálu.

Je také důležité vyladit všechny koncové komponenty na síťové cestě:

1. Pokusy o připojení k nedostupným systémům se musí okamžitě nezdařit, takže se pokusy o opětovné připojení mohou přesunout na první dostupné místo určení.

Pro trasy MQIPT SSL vyladíte vlastnost přenosové cesty **SSLClientConnectTimeout** tak, aby se zajistilo selhání připojení výzvy k zadání pro nedostupná místa určení. Podrobné informace o parametrech ladění programu IBM MQ naleznete v dokumentaci produktu IBM MQ . Podrobnosti o ladění TCP/IP pro operační systém naleznete také v dokumentaci k operačnímu systému. Ve všech případech by neúspěšné pokusy o připojení měly rychle vrátit selhání sítě (například paket resetu TCP), nebo by měly být bez zbytečného odkladu časově neomezené.

2. Aktivní připojení k systému, který selhal, musí být rychle přerušeny tak, aby bylo možné vytvořit nová připojení.

Měli byste také zvážit dopad překonání selhání v době, kdy připojení aktivně používají produkt MQIPT. Během překonání selhání je pravděpodobné, že síťová připojení budou přerušeny. U aplikací typu klient můžete použít funkci automatického opětovného připojení klienta IBM MQ k opětovnému vytvoření přerušené spojení. V případě kanálů zpráv můžete zadat krátký interval opakování, aby se kanál rychle znovu připojil. Další informace o automatickém opětovném připojení klienta a konfiguraci opakovaných pokusů o kanál zpráv naleznete v dokumentaci produktu IBM MQ .



## Poznámky

---

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 Velké Británie nebo kterékoliv jiné země, kde taková opatření odporují místním zákonům:** SPOLEČNOST INTERNATIONAL BUSINESS MACHINES CORPORATION TUTO PUBLIKACI POSKYTUJE TAKOVOU, "JAKÁ 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 VYPLÝVAJÍCÍCH Z OKOLNOSTÍ. 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  
Koordinátor spolupráce softwaru, oddělení 49XA  
148 00 Praha 4-Chodby

148 00 Praha 4-Chodov  
U.S.A.

Poskytnutí takových informací může být podmíněno dodržáním určitých podmínek a požadavků zahrnujících v některých případech uhrazení stanoveného poplatku.

IBM poskytuje licencovaný program popsany v těchto informacích a veškeré dostupné licencované materiály na základě podmínek smlouvy IBM Customer Agreement, IBM International Program License Agreement 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 programátorských rozhraní, jsou-li poskytovány, jsou určeny k tomu, aby vám pomohly vytvořit aplikační software pro použití s tímto programem.

Tato příručka obsahuje informace o zamýšlených programovacích rozhraních, které umožňují zákazníkům 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:** Nepoužívejte tyto informace o diagnostice, úpravách a ladění jako programátorské 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 IBM je k dispozici na webu na stránce "Copyright and trademark information" [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml). Ostatní 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 obsahuje software vyvinutý v rámci projektu Eclipse Project (<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: