

7.5

*IBM WebSphere MQ ' nun
yapılandırılması*

IBM

Not

Bu bilgileri ve desteklediđi ürünü kullanmadan önce, "[Özel notlar](#)" sayfa 443 bölümündeki bilgileri okuyun.

Bu basım, yeni basımlarında tersi belirtilmediđi sürece, IBM® WebSphere MQ 'ın 7. yayın düzeyi 5 'i ve sonraki tüm yayın ve deđişiklik düzeyleri için geçerlidir.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

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

İçindekiler

Yapılandırılıyor.....	5
UNIX, Linuxve Windows üzerinde birden çok kuruluş yapılandırılması.....	5
Birden çok kuruluş ortamında uygulamaların bağlanması.....	6
Birincil kuruluşun değiştirilmesi.....	15
Bir kuyruk yöneticisinin kuruluşla ilişkilendirilmesi.....	16
Bir sistemde IBM WebSphere MQ kuruluşlarını bulma.....	18
Kuyruk yöneticilerinin yaratılması ve yönetilmesi.....	19
Varsayılan kuyruk yöneticisi yaratılması.....	21
Var olan bir kuyruk yöneticisinin varsayılan olarak kullanılması.....	23
Kuyruk yöneticisi yaratıldıktan sonra yapılanış kütüklerinin yedeklenmesi.....	23
Kuyruk yöneticisinin başlatılması.....	24
Kuyruk yöneticisinin durdurulması.....	24
Kuyruk yöneticisinin yeniden başlatılması.....	26
Kuyruk yöneticisinin silinmesi.....	26
Dağıtılmış kuyruğa alma kullanılarak uygulamaların bağlanması.....	27
IBM WebSphere MQ dağıtılmış ileti sistemi teknikleri.....	28
Dağıtılmış kuyruk yönetimine giriş.....	47
UNIX, Linuxve Windows 'ta kanalların izlenmesi ve denetlenmesi.....	72
Sunucu ve istemciler arasındaki bağlantıların yapılandırılması.....	96
Hangi iletişim tipinin kullanılacağına karar verilmesi.....	98
Genişletilmiş bir işlem istemcisinin yapılandırılması.....	100
MQI kanallarının tanımlanması.....	109
Farklı altyapılarda sunucu bağlantısı ve istemci bağlantısı tanımlamaları yaratılması.....	111
Sunucuda sunucu bağlantısı ve istemci bağlantısı tanımlamaları yaratılması.....	114
Kanal çıkışları.....	118
İstemcinin kuyruk paylaşım grubuna bağlanması.....	122
Yapılanış kütüğü kullanılarak istemci yapılandırılması.....	124
Ortam değişkenlerini kullanarak istemci yapılandırılması.....	142
Kuyruğa alınan yayınlama/abone olma denetleniyor.....	149
Kuyruğa alınan yayınlama/abone olma iletisi özniteliklerini ayarlama.....	149
Kuyruğa alınan yayınlama/abone olma başlatılıyor.....	150
Kuyruğa alınan yayınlama/abone olma durduruluyor.....	151
Akış eklenmesi.....	151
Akış silinmesi.....	152
Abonelik noktası eklenmesi.....	153
Kuyruk yöneticisinin bir sıradüzene bağlanması.....	154
Bir kuyruk yöneticisinin sıradüzenden bağlantısının kesilmesi.....	156
Kuyruk yöneticisi kümesinin yapılandırılması.....	156
Erişim denetimi ve birden çok küme iletim kuyruğu.....	157
Dağıtılmış kuyruğa alma ile karşılaştırma.....	158
Bir kümenin bileşenleri.....	161
Tam havuzları tutmak için küme kuyruğu yöneticilerinin seçilmesi.....	173
Kümenin düzenlenmesi.....	175
Küme adlandırma kuralları.....	175
Çakışan kümeler.....	176
Kümeleme İpuçları.....	177
Kümede iletişim kurulması.....	179
Havuz bilgilerinin saklanması.....	180
IBM WebSphere MQ kümelerini yönetme.....	181
İletileri kümelere/kümelerden yönlendirme.....	244
İş yükü yönetimi için kümeleri kullanma.....	258
Kümeleme: En iyi uygulamalar.....	273

Kullanılabilirlik, kurtarma ve yeniden başlatma.....	303
Otomatik istemci yeniden bağlantısı.....	304
Konsol ileti izlemesi.....	309
Yüksek kullanılabilirlikli yapılandırmalar.....	310
İletilerin kaybolmadığından emin olmak (günlüğe kaydetme).....	391
IBM WebSphere MQ kuyruk yöneticisi verilerinin yedeklenmesi ve geri yüklenmesi.....	405
Yapılandırma bilgilerini değiştirme.....	410
UNIX, Linuxve Windows sistemlerinde yapılandırma bilgilerinin değiştirilmesi.....	410
IBM WebSphere MQ yapılandırma bilgilerini değiştirmeye ilişkin öznitelikler.....	416
Kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi.....	422
yapılandırmaHP Integrity NonStop Server.....	439
Gateway sürecine genel bakış.....	439
Ağ geçidi, Yol altında çalışacak şekilde yapılandırılıyor.....	439
İstemci kullanıma hazırlama kütüğünün yapılandırılması.....	441
Kanallara izin verilmesi.....	441
Özel notlar.....	443
Programlama arabirimi bilgileri.....	444
Ticari Markalar.....	444

Yapılandırılıyor

Bir ya da daha çok bilgisayarda bir ya da daha çok kuyruk yöneticisi yaratın ve bunları geliştirme, test etme ve üretim sistemlerinizde, iş verilerinizi içeren iletileri işlemek için yapılandırın.

IBM WebSphere MQ' u yapılandırmadan önce, [IBM WebSphere MQ Teknik genel bakış](#) başlıklı konuda IBM WebSphere MQ kavramları hakkında bilgi edinin. [Planning \(Planlama\)](#) içinde IBM WebSphere MQ ortamınızı nasıl planlamanız hakkında bilgi edinin.

Kuyruk yöneticilerinizi ve ilgili kaynaklarını IBM WebSphere MQ içinde yaratmak, yapılandırmak ve denetlemek için kullanabileceğiniz çeşitli yöntemler vardır. Bu yöntemler komut satırı arabirimlerini, grafik kullanıcı arabirimini ve denetim API 'sini içerir. Bu arabirimler hakkında daha fazla bilgi için bkz. [IBM WebSphere MQ Yönetimi](#).

Kuyruk yöneticisini yaratma, başlatma, durdurma ve silme ile ilgili yönergeler için bkz. [“Kuyruk yöneticilerinin yaratılması ve yönetilmesi” sayfa 19.](#)

IBM WebSphere MQ kuruluşlarınızı ve uygulamalarınızı birbirine bağlamak için gereken bileşenlerin nasıl yaratılabileceği hakkında bilgi için bkz. [“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27.](#)

Farklı yöntemler kullanarak istemcilerinizi bir IBM WebSphere MQ sunucusuna nasıl bağlayabilmeye ilişkin yönergeler için bkz. [“İstemci ile sunucu arasındaki bağlantıların yapılandırılması” sayfa 96.](#)

Kuyruk yöneticisi kümesinin nasıl yapılandırılabilmesiyle ilgili yönergeler için bkz. [“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156.](#)

Yapılanış bilgilerini değiştirerek, IBM WebSphere MQ ya da kuyruk yöneticisinin davranışını değiştirebilirsiniz. Daha fazla bilgi için [“IBM WebSphere MQ ve kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi” sayfa 410](#) başlıklı konuya bakın. Genel olarak, bu ürün belgelerinde belirtilenler dışında, herhangi bir yapılandırma değişikliklerinin yürürlüğe girmesi için kuyruk yöneticisini yeniden başlatmanız gerekmez.

İlgili kavramlar

[WebSphere MQ teknik genel bilgileri](#)

İlgili görevler

[Yerel WebSphere MQ nesnelere yönetme](#)

[Uzak WebSphere MQ nesnelere yönetme](#)

[Planlama](#)

UNIX, Linux, and Windows üzerinde birden çok kuruluş yapılandırılması

Aynı sistemde birden çok kuruluş kullanırken, kuruluşları ve kuyruk yöneticilerini yapılandırmanız gerekir.

Bu bilgiler UNIX, Linux®, and Windows için geçerlidir.

Kuruluşlarınızı yapılandırmak için aşağıdaki bağlantılarda yer alan bilgileri kullanın:

- [“Birincil kuruluş değiştiriliyor” sayfa 15](#)
- [“Kuyruk yöneticisinin bir kuruluşla ilişkilendirilmesi” sayfa 16](#)
- [“Uygulamaların birden çok kuruluş ortamında bağlanması” sayfa 6](#)

İlgili kavramlar

[Birden çok kuruluş](#)

İlgili görevler

[Birincil kuruluş seçilmesi](#)

[Kuruluş adı seçilmesi](#)

Uygulamaların birden çok kuruluş ortamında bağlanması

UNIX, Linux, and Windows sistemlerinde, IBM WebSphere MQ Version 7.1ya da sonraki bir sürümü, kitaplıklar yüklenirse, IBM WebSphere MQ , başka herhangi bir işlem yapmak için gereken kitaplıkları otomatik olarak kullanır. IBM WebSphere MQ Uygulamanın bağlandığı kuyruk yöneticisiyle ilişkilendirilmiş kuruluşlardan kitaplıkları kullanır.

Uygulamaların IBM WebSphere MQ' e bağlanma şeklini açıklamak için aşağıdaki kavramlar kullanılır:

bağlama

Uygulama derlendiğinde, uygulama çalıştırıldığında o zaman yüklenen işlev dışı aktarmasını almak için uygulama IBM WebSphere MQ kitaplıklarıyla bağlantılıdır.

Yükleniyor

Uygulama çalıştırılırsa, IBM WebSphere MQ kitaplıkları bulunur ve yüklenir. Kitaplıkların yerini belirlemek için kullanılan özel mekanizma işletim sistemine göre değişir ve uygulamanın nasıl oluşturulduyla ilgili olarak değişir. Birden çok kuruluş ortamında kitaplıkların nasıl bulunabileceği ve yükleneceği hakkında daha fazla bilgi için bkz. [“IBM WebSphere MQ Version 7.1 ya da sonraki kitaplıkların yüklenmesi” sayfa 8.](#)

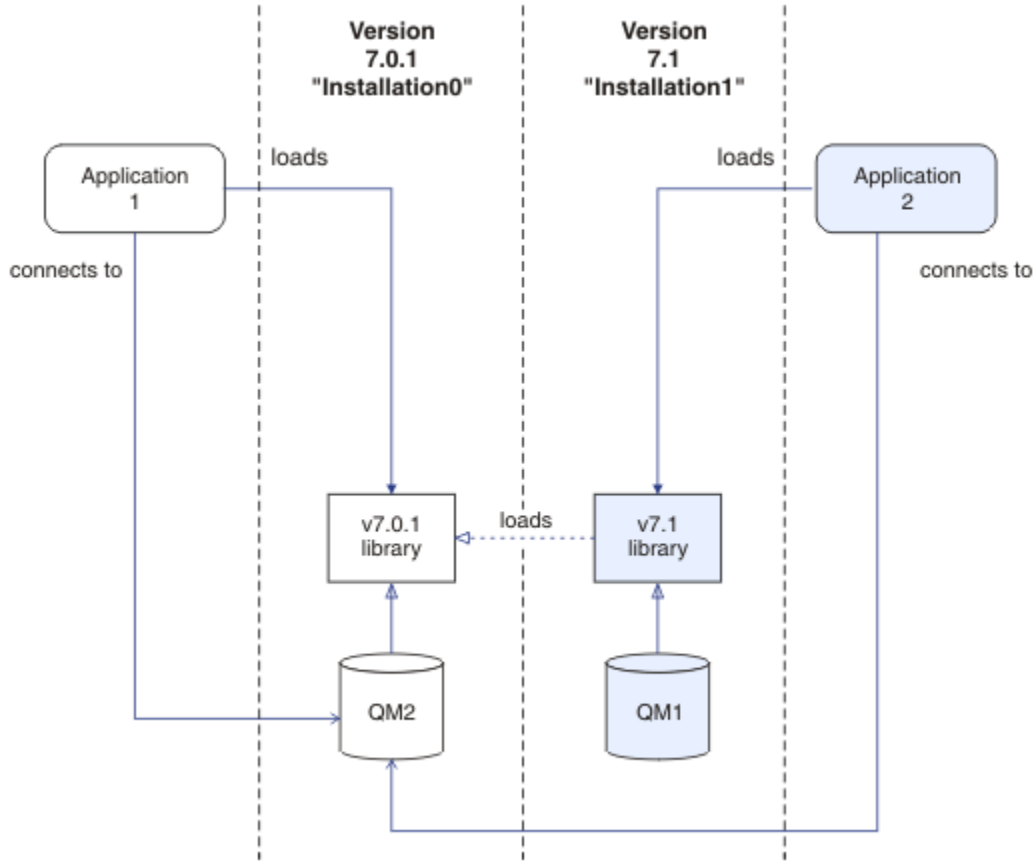
Bağlantı kuruluyor

Uygulama çalışan bir kuyruk yöneticisine bağlandığında (örneğin, bir MQCONN ya da MQCONNX çağrısını kullanarak, yüklenen IBM WebSphere MQ kitaplıklarını kullanarak bağlanır.)

Bir sunucu uygulaması bir kuyruk yöneticisine bağlandığında, yüklenen kitaplıkların kuyruk yöneticisiyle ilişkili kuruluştan gelmesi gerekir. Bir sistemde birden çok kuruluş olduğunda, bu kısıtlama, işletim sisteminin yüklenecek IBM WebSphere MQ kitaplıklarını bulmak için kullandığı mekanizmayı seçerken yeni güçlükler sunar:

- Bir kuyruk yöneticisiyle ilişkilendirilmiş kuruluşu değiştirmek için **setmqm** komutu kullanıldığında, yüklenmesi gereken kitaplıklar değişir.
- Bir uygulama, farklı kuruluşlara ait olan birden çok kuyruk yöneticisine bağlandığında, birden çok kitaplık kümesinin yüklenmesi gerekir.

Ancak, IBM WebSphere MQ Version 7.1ya da sonraki bir sürümü varsa, kitaplıklar konumlandırılır ve yüklenirse, IBM WebSphere MQ daha sonra herhangi bir işlem yapmak zorunda kalmaksızın uygun kitaplıkları yükler ve kullanır. Uygulama bir kuyruk yöneticisine bağlandığında, IBM WebSphere MQ , kuyruk yöneticisinin ilişkilendirildiği kuruluşun kitaplıklarını yükler.



Şekil 1. Uygulamaların birden çok kuruluş ortamında bağlanması

Örneğin, Şekil 1 sayfa 7 içinde sürüm 7.0.1 kuruluşu (Installation0) ve bir sürüm 7.1 kuruluşu (Installation1) olan birden çok kuruluş ortamı gösterilir. Bu kuruluşlara iki uygulama bağlı, ancak bunlar farklı kitaplık sürümlerini yüklemektedir.

Application 1, bir sürüm 7.0.1 kitaplığını doğrudan yükler. application 1 QM2' e bağlandığında, sürüm 7.0.1 kitaplıkları kullanılır. application 1, QM1' e bağlanmayı denerse ya da QM2, Installation1 ile ilişkilendirilmişse, application 1 bir 2059 (080B) (RC2059): MQRC_Q_MGR_NOT_AVAM hatasıyla başarısız olur. Sürüm 7.0.1 kitaplığı, diğer kitaplık sürümlerini yükleme yeteneğine sahip olmadığından uygulama başarısız olur. That is, if version 7.0.1 libraries are directly loaded, you cannot use a queue manager associated with an installation at a later version of IBM WebSphere MQ.

Application 2, bir sürüm 7.1 kitaplığını doğrudan yükler. application 2 QM2' a bağlandığında, sürüm 7.1 kitaplığı daha sonra sürüm 7.0.1 kitaplığını yükler ve kullanır. application 2, QM1 ile bağlantı kurarsa ya da QM2 Installation1 ile ilişkilendirilmişse, sürüm 7.1 kitaplığı yüklenir ve uygulama beklendiği gibi çalışır.

Geçiş senaryoları ve birden çok kurulumu sahip uygulamalar, UNIX, Linux ve Windows üzerinde çok kuruluşlu kuyruk yöneticisi varlığında daha fazla ayrıntı olarak kabul edilir.

IBM WebSphere MQ Version 7.1 kitaplıklarının nasıl yükleneceği hakkında daha fazla bilgi için bkz. "IBM WebSphere MQ Version 7.1 ya da sonraki kitaplıkların yüklenmesi" sayfa 8.

Destek ve sınırlamalar

Aşağıdaki sürümlerden herhangi biri 7.1 ya da sonraki bir sürümse, kitaplıklar bulunur ve yüklenirse, IBM WebSphere MQ uygun kitaplıkları otomatik olarak yükleyebilir ve kullanabilir:

- C sunucusu kitaplıkları

- C++ sunucu kitaplıkları
- XA sunucusu kitaplıkları
- COBOL sunucusu kitaplıkları
- COM + sunucu kitaplıkları
- .NET yönetilmeyen kipteki

IBM WebSphere MQ ayrıca, bağı tanımları kipinde Java ve JMS uygulamaları için uygun kitaplıkları otomatik olarak yükler ve kullanır.

Birden çok kuruluş kullanan uygulamalar için bir dizi kısıtlama vardır. Daha fazla bilgi için [“Birden çok kuruluş kullanan uygulamalara ilişkin kısıtlamalar” sayfa 11](#) başlıklı konuya bakın.

İlgili kavramlar

[“Kuyruk yöneticisinin bir kuruluşla ilişkilendirilmesi” sayfa 16](#)

Bir kuyruk yöneticisi yarattığınızda, bu yönetici otomatik olarak **crtmqm** komutunu veren kuruluşla ilişkilendirilir. On UNIX, Linux, and Windows, you can change the installation associated with a queue manager using the **setmqm** command.

[“Birden çok kuruluş kullanan uygulamalara ilişkin kısıtlamalar” sayfa 11](#)

Birden çok kuruluş ortamında CICS sunucu kitaplıkları, hızlı yol bağlantıları, ileti tanıtıcıları ve çıkışlar kullanılırken kısıtlamalar vardır.

[“IBM WebSphere MQ Version 7.1 ya da sonraki kitaplıkların yüklenmesi” sayfa 8](#)

IBM WebSphere MQ kitaplıklarının nasıl yükleneceğine karar verirken, ortamınız da dahil olmak üzere, var olan uygulamalarınızı değiştirebileceğiniz, birincil kuruluş isteyip istemediğiniz, IBM WebSphere MQ ' in kurulu olduğu ve IBM WebSphere MQ konumunun değişip değişmeyeceği bir dizi faktörün göz önünde bulundurulması gerekir.

İlgili görevler

[Birincil kuruluş seçilmesi](#)

[“Birincil kuruluş değiştiriliyor” sayfa 15](#)

Bir kuruluşu birincil kuruluş olarak ayarlamak ya da ayarlamayı kaldırmak için **setmqinst** komutunu kullanabilirsiniz.

IBM WebSphere MQ Version 7.1 ya da sonraki kitaplıkların yüklenmesi

IBM WebSphere MQ kitaplıklarının nasıl yükleneceğine karar verirken, ortamınız da dahil olmak üzere, var olan uygulamalarınızı değiştirebileceğiniz, birincil kuruluş isteyip istemediğiniz, IBM WebSphere MQ ' in kurulu olduğu ve IBM WebSphere MQ konumunun değişip değişmeyeceği bir dizi faktörün göz önünde bulundurulması gerekir.

IBM WebSphere MQ Version 7.1 kitaplıklarının yeri ve yüklenmesi, kuruluş ortamınıza bağlıdır:

- UNIX and Linux sistemlerinde, varsayılan konuma IBM WebSphere MQ Version 7.1 kopyası kuruluysa, var olan uygulamalar önceki sürümlerle aynı şekilde çalışmaya devam eder. Ancak, uygulamaların /usr/lib' ta simgesel bağlantılara gereksinim duyarsa, birincil kuruluş olarak bir sürüm 7.1 kuruluşu seçmeniz ya da simgesel bağlantıları el ile oluşturmanız gerekir.
- IBM WebSphere MQ Version 7.1 varsayılan olmayan bir konuma kurulduysa, bu durumda IBM WebSphere MQ Version 7.0.1 da kuruluysa, doğru kitaplıkların yüklenebilmesi için var olan uygulamalarınızı değiştirmeniz gerekebilir.

How IBM WebSphere MQ Version 7.1, or later, libraries can be located and loaded also depends on how any existing applications are set up to load libraries. Kitaplıkların nasıl yüklenebileceğiyle ilgili daha fazla bilgi için bkz. [“İşletim sistemi kitaplığı yükleme mekanizmaları” sayfa 10.](#)

En iyi şekilde, kuyruk yöneticisinin, işletim sistemi tarafından yüklenen IBM WebSphere MQ kitaplığıyla ilişkili olduğundan emin olmanız gerekir.

IBM WebSphere MQ kitaplıklarını yüklemeye ilişkin yöntemler platforma göre değişir ve her yöntemin avantaj ve dezavantajları vardır.

Çizelge 1. Kitaplıkların yüklenmesine ilişkin seçeneklerin yararları ve sakıncaları

Altyapı	Seçenek	Yararlar	Dezavantajlar
UNIX and Linux sistemleri	<p>Uygulamanın gömülü yürütme ortamı arama yolunu (RPath) ayarlayın ya da değiştirin.</p> <p>Bu seçenek, uygulamayı yeniden derlemenizi ve bağlamanızı gerektirir. Uygulamaların derlenmesi ve bağlanması hakkında daha fazla bilgi için WebSphere MQ uygulamasının oluşturulması başlıklı konuya bakın.</p>	<ul style="list-style-type: none">Değişikliğin kapsamı nettir.	<ul style="list-style-type: none">Uygulamayı yeniden derleyebilmeli ve bağlantılandırılabilmeniz.IBM WebSphere MQ ' in konumu değişirse, RPath 'i değiştirmeniz gerekir.
UNIX and Linux sistemleri	<p><code>LD_LIBRARY_PATH</code> ortam değişkenini (AIX üzerinde <code>LIBPATH</code>), <code>setmqenv</code> ya da <code>crtmqenv</code> komutunu kullanarak <code>-k</code> ya da <code>-l</code> seçeneğini kullanın.</p>	<ul style="list-style-type: none">Var olan uygulamalarda değişiklik yapılması gerekmez.Bir uygulamadaki gömülü RPaths ögesini geçersiz kılar.IBM WebSphere MQ ' in konumu değişirse, değişkeni değiştirmek kolaydır.	<ul style="list-style-type: none">setuid and setgid applications, or applications built in other ways, might ignore <code>LD_LIBRARY_PATH</code> for security reasons.Ortama özgü olduğu için, uygulamanın çalıştırıldığı her ortamda ayarlanmalıdır.<code>LD_LIBRARY_PATH</code> ' a güvenen diğer uygulamalar üzerinde olası etki.HP-UX: Uygulama derlendiğinde kullanılan seçenekler, <code>LD_LIBRARY_PATH</code> kullanımı geçersiz kılabilir. Daha fazla bilgi için bkz. HP-UX için çalışma zamanı bağlantılandırma noktaları.Linux: Uygulamayı oluşturmak için kullanılan derleyici, <code>LD_LIBRARY_PATH</code> kullanımı geçersiz kılabilir. Daha fazla bilgi için bkz. Linux için çalışma zamanı bağlantılandırma noktaları.

Çizelge 1. Kitaplıkların yüklenmesine ilişkin seçeneklerin yararları ve sakıncaları (devamı var)

Altyapı	Seçenek	Yararlar	Dezavantajlar
Windows sistemleri	PATH değişkenini <code>setmqenv</code> ya da <code>crtmqenv</code> kullanarak ayarlayın.	<ul style="list-style-type: none">Var olan uygulamalar için herhangi bir değişiklik gerekli değil.IBM WebSphere MQ ' in konumu değişirse, değişkeni değiştirmek kolaydır.	<ul style="list-style-type: none">Ortama özgü olduğu için, uygulamanın çalıştırıldığı her ortamda ayarlanmalıdır.Diğer uygulamalar üzerinde olası etki.
UNIX, Linux, and Windows sistemleri	Birincil kuruluğu bir sürüm 7.1ya da sonraki bir sürüme ayarlayın. Bkz. " Birincil kuruluş değiştiriliyor " sayfa 15. Birincil kuruluşla ilgili ek bilgi için Birincil kuruluşun seçilmesi başlıklı konuya bakın.	<ul style="list-style-type: none">Var olan uygulamalar için herhangi bir değişiklik gerekli değil.IBM WebSphere MQ ' in konumu değişirse, birincil kuruluşu değiştirmek kolay olur.Gives similar behavior to previous versions of IBM WebSphere MQ.	<ul style="list-style-type: none">When WebSphere MQ version 7.0.1 is installed, you cannot set the primary installation to version 7.1, or later.UNIX and Linux: <code>/usr/lib</code> varsayılan arama yolunda değilse çalışmaz.

Library loading considerations for HP-UX

Önceki IBM WebSphere MQ sürümlerine ilişkin ürün belgelerindeki örnek derleme komutları, 64 bit uygulamalar için `-W1`, `+noenvvar` bağlantı seçeneğini içerir. Bu seçenek, paylaşılan kitaplıkları yüklemek için `LD_LIBRARY_PATH` kullanımını devre dışı bırakır. Uygulamalarınızın RPath içinde belirtilen konumdan farklı bir konumdan IBM WebSphere MQ kitaplıklarını yüklemesini istiyorsanız, uygulamalarınızı güncellemeniz gerekir. Uygulamaları `-W1`, `+noenvvar` bağlantı seçeneği olmadan yeniden derleyerek ve bağlantıyla ya da `chatx` komutunu kullanarak güncelleştirebilirsiniz.

Uygulamalarınızın kitaplıkları nasıl yüklediğini öğrenmek için bkz. "[İşletim sistemi kitaplığı yükleme mekanizmaları](#)" sayfa 10.

Library loading considerations for Linux

Bazı gcc sürümleri (örneğin, sürüm 3.2.x) kullanılarak derlenen uygulamalar, `LD_LIBRARY_PATH` ortam değişkeni kullanılarak geçersiz kılınamayan bir RPath yerleşik RPath olabilir. You can determine if an application is affected by using the `readelf -d applicationName` command. RPATH simgesi varsa ve RUNPATH simgesi yoksa RPath geçersiz kılınmaz.

Library loading considerations for Solaris

Önceki IBM WebSphere MQ sürümlerine ilişkin ürün belgelerindeki örnek derleme komutları, `-lmqmc` `-lmqzse` bağlantı seçeneklerini içerir. Bu kitaplıkların uygun sürümleri şu anda otomatik olarak IBM WebSphere MQ tarafından yüklenir. IBM WebSphere MQ , varsayılan olmayan bir konuma kurulduysa ya da sistemde birden çok kuruluş varsa, uygulamalarınızı güncellemeniz gerekir. Uygulamaları `-lmqmc` `-lmqzse` bağlantı seçenekleri olmadan yeniden derleyerek ve bağlantıyla güncelleyebilirsiniz.

İşletim sistemi kitaplığı yükleme mekanizmaları

Windows sistemlerinde, kitaplıkları bulmak için birkaç dizin aranır:

- Uygulamanın yüklendiği dizin.
- Yürürlükteki dizin.

- *YOL* ortam değişkenindeki dizinler, hem genel *YOL* değişkeni, hem de yürürlükteki kullanıcının *YOL* değişkenindeki dizinler.

UNIX and Linux sistemlerinde, yüklenecek kitaplıkların yerini belirlemek için kullanılan birçok yöntem vardır:

- *LD_LIBRARY_PATH* ortam değişkeni (AIX'ta *LIBPATH* ve HP-UX' ta *SHLIB_PATH*) kullanılıyor. Bu değişken ayarlandıysa, gerekli WebSphere MQ kitaplıkları için arama yapılan bir dizin kümesini tanımlar. Bu dizinlerde bulunan kitaplıklar, diğer yöntemler kullanılarak bulunabilecek kitaplıkların tercihlerinde kullanılır.
- Gömülü bir arama yolu (RPath) kullanılıyor. Uygulama, IBM WebSphere MQ kitaplıklarını aramak için bir dizin kümesi içerebilir. *LD_LIBRARY_PATH* ayarlanmadıysa ya da gerekli kitaplıklar değişken kullanılarak bulunamazsa, RPath kitaplıkları aranır. Var olan uygulamalarınız bir RPath kullanırsa, ancak uygulamayı yeniden derleyemez ve bağlantı kuramıyorsanız, varsayılan konuma IBM WebSphere MQ Version 7.1 programını kurmalı ya da kitaplıkları bulmak için başka bir yöntem kullanmalısınız.
- Varsayılan kitaplık yolu kullanılıyor. If the WebSphere MQ libraries are not found after searching the *LD_LIBRARY_PATH* variable and RPath locations, the default library path is searched. Genellikle bu yol, */usr/lib* ya da */usr/lib64* içerir. Varsayılan kitaplık yolunda arama yapıldıktan sonra kitaplıklar bulunamazsa, eksik bağımlılıklar nedeniyle uygulama başlatılamaz.

Uygulamalarınızın yerleşik bir arama yoluna sahip olup olmadığını öğrenmek için işletim sistemi mekanizmalarını kullanabilirsiniz. Örneğin:

- AIX: **dump**
- HP-UX: **chatr**
- Linux: **readelf**
- Solaris: **elfdump**

İlgili kavramlar

[“Kuyruk yöneticisinin bir kuruluşla ilişkilendirilmesi” sayfa 16](#)

Bir kuyruk yöneticisi yarattığınızda, bu yönetici otomatik olarak **crtmqm** komutunu veren kuruluşla ilişkilendirilir. On UNIX, Linux, and Windows, you can change the installation associated with a queue manager using the **setmqm** command.

[“Birden çok kuruluş kullanan uygulamalara ilişkin kısıtlamalar” sayfa 11](#)

Birden çok kuruluş ortamında CICS sunucu kitaplıkları, hızlı yol bağlantıları, ileti tanıtıcıları ve çıkışlar kullanılırken kısıtlamalar vardır.

[“Uygulamaların birden çok kuruluş ortamında bağlanması” sayfa 6](#)

UNIX, Linux, and Windows sistemlerinde, IBM WebSphere MQ Version 7.1'ya da sonraki bir sürümü, kitaplıklar yüklenirse, IBM WebSphere MQ , başka herhangi bir işlem yapmak için gereken kitaplıkları otomatik olarak kullanır. IBM WebSphere MQ Uygulamanın bağlandığı kuyruk yöneticisiyle ilişkilendirilmiş kuruluştan kitaplıkları kullanır.

İlgili görevler

[Birincil kuruluş seçilmesi](#)

[“Birincil kuruluş değiştiriliyor” sayfa 15](#)

Bir kuruluşu birincil kuruluş olarak ayarlamak ya da ayarlamayı kaldırmak için **setmqinst** komutunu kullanabilirsiniz.

Birden çok kuruluş kullanan uygulamalara ilişkin kısıtlamalar

Birden çok kuruluş ortamında CICS sunucu kitaplıkları, hızlı yol bağlantıları, ileti tanıtıcıları ve çıkışlar kullanılırken kısıtlamalar vardır.

CICS sunucu kitaplıkları

CICS sunucu kitaplıklarını kullanıyorsanız, IBM WebSphere MQ sizin için doğru kitaplık düzeyini otomatik olarak seçmez. Uygulamanızı, uygulamanın bağlandığı kuyruk yöneticisi için uygun kitaplık düzeyiyle

derlemeli ve bağlamanız gerekir. Ek bilgi için [Building libraries for use with TXSeries for Multiplatforms version 5](#) başlıklı konuya bakın.

İleti tanıtıcıları

MQHC_UNASSOCIATED_HCONN özel değerini kullanan ileti tanıtıcıları, bir süreçte yüklenen ilk kuruluşla birlikte kullanılmak üzere sınırlandırılır. If the message handle cannot be used by a particular installation, reason code MQRC_HMSG_NOT_AVAILABLE is returned.

Bu kısıtlama, ileti özelliklerini etkiler. Bir kuruluş üzerindeki bir kuyruk yöneticisinden ileti özelliklerini almak ve bunları farklı bir kurulumla bir kuyruk yöneticisine koymak için ileti tutamaçlarını kullanamazsınız. İleti tutamaçlarıyla ilgili daha fazla bilgi için bkz. [MQCRTMH-İleti tanıtıcısı yarat.](#)

Çıkışlar

Birden çok kuruluş ortamında, var olan çıkışlar, IBM WebSphere MQ Version 7.1 ile ya da daha sonraki kuruluşlarla kullanılmak üzere güncellenmelidir. Data conversion exits generated using the **crtmqcvx** command must be regenerated using the updated command.

Tüm çıkışlar MQIEP yapısı kullanılarak yazılmalı, IBM WebSphere MQ kitaplıklarını bulmak için yerleşik bir RPATH kullanamaz ve IBM WebSphere MQ kitaplıklarına bağlantı kuramaz. Ek bilgi için [Yazma ve derleme çıkışları ve kurulabilir hizmetler](#) başlıklı konuya bakın.

Hızlı yol

Birden çok kurulumla sahip bir sunucuda, IBM WebSphere MQ Version 7.1 ya da daha sonraki bir yayın düzeyiyle hızlı bir yol bağlantısı kullanan uygulamalar aşağıdaki kurallara uymalıdır:

1. Kuyruk yöneticisinin, uygulamanın IBM WebSphere MQ çalıştırma zamanı kitaplıklarını yüklediği kuruluşla aynı kuruluşla ilişkilendirilmesi gerekir. Uygulama, farklı bir kuruluşla ilişkilendirilmiş bir kuyruk yöneticisiyle hızlı bir yol bağlantısı kullanmıyor olmalıdır. Bir hata ile bağlantı sonuçlarını yapma girişimi ve neden kodu MQRC_INSTALLATION_MISMATCH.
2. Uygulamanın IBM WebSphere MQ yürütme zamanı kitaplıklarını yüklediği aynı kuruluşla ilişkili bir kuyruk yöneticisine hızlı olmayan yol bağlanması, bu koşulların herhangi biri doğru değilse, uygulamanın hızlı yolu bağlamasını önler:
 - Uygulama, aynı kuruluş ile ilişkili bir kuyruk yöneticisiyle ilk bağlantısını hızlı bir yol bağlantısıyla ilişkilendirir.
 - Ortam değişkeni AMQ_SINGLE_INSTALLATION ayarlıdır.
3. Hızlı olmayan yolun Version 7.1 ya da daha sonraki bir kuruluşla ilişkilendirilmiş bir kuyruk yöneticisine bağlanması, uygulamanın hızlı bir yol bağlayıp bağlanmadığına ilişkin bir etkisi yoktur.
4. Bir Version 7.0.1 kuruluşuyla ilişkilendirilmiş bir kuyruk yöneticisine bağlanma ve Version 7.1 ya da daha sonraki bir kuruluş ile ilişkilendirilmiş bir kuyruk yöneticisine hızlı yolu bağlamayı birleştiremezsiniz.

AMQ_SINGLE_INSTALLATION kümesiyle, bir kuyruk yöneticisi ile hızlı yol bağlantısı arasında bağlantı kurabilirsiniz. Aksi takdirde, hemen hemen aynı kısıtlamalar geçerli olur:

- Kuruluş, IBM WebSphere MQ çalıştırma zamanı kitaplıklarının yüklediği aynı olmalıdır.
- Aynı süreçle ilgili her bağlantı aynı kurulumla olmalıdır. Farklı bir kuruluşla ilişkilendirilmiş bir kuyruk yöneticisine bağlanmayı denerseniz, bağlantı neden kodu MQRC_INSTALLATION_MISMATCH ile başarısız olur. AMQ_SINGLE_INSTALLATION kümesi ile bu kısıtlamanın yalnızca hızlı yol bağlantıları değil, tüm bağlantılar için geçerli olduğunu unutmayın.
- Hızlı yol bağlantılarıyla yalnızca bir kuyruk yöneticisini bağlayın.

İlgili başvurular

[MQCONN-Connect kuyruk yöneticisi \(genişletilmiş\)](#)

[MQIEP yapısı](#)

[2583 \(0A17\) \(RC2583\): MQRC_INSTALLATION_MISMATCH](#)

2587 (0A1B) (RC2587): MQR_C_HMSG_NOT_VAR

2590 (0A1E) (RC2590): MQR_FASTPATH_NOT_AVAM

.NET uygulamalarının birden çok kuruluş ortamında bağlanması

Varsayılan olarak, uygulamalar birincil kurulumdan .NET yapıbirimlerini kullanır. Birincil kuruluş yoksa ya da birincil kuruluş düzeneklerini kullanmak istemiyorsanız, uygulama yapılandırma dosyasını ya da *DEVPATH* ortam değişkenini güncellemeniz gerekir.

Sistemde birincil kuruluş varsa, bu kurulumla ilişkin .NET yapıbirimleri ve ilke dosyaları genel derleme önbelleğine (GAC) kaydedilir. Diğer tüm kuruluşlar için .NET yapıbirimleri her kuruluşun kuruluş yolunda bulunabilir, ancak düzenekler GAC ' ye kaydedilmez. Bu nedenle, varsayılan olarak, uygulamalar birincil kurulumdan .NET yapıbirimleri kullanarak çalışır. Aşağıdaki durumlardan biri geçerliyse, uygulama yapılandırma dosyasını güncellemeniz gerekir:

- Birincil kurulumla sahip değilsiniz.
- Uygulamanın birincil kuruluş düzeneklerini kullanmasını istemeyin.
- Birincil kuruluş, IBM WebSphere MQ ürününün, uygulamanın derlendiği sürümden daha düşük bir sürümdür.

Uygulama yapılandırma dosyasının nasıl güncelleneceği hakkında bilgi için bkz. [“.NET uygulamalarının uygulama yapılandırma dosyası kullanılarak bağlanması” sayfa 13.](#)

Aşağıdaki durum doğruysa, *DEVPATH* ortam değişkenini güncellemeniz gerekir:

- Uygulamanızın montajları birincil olmayan bir kurulumdan kullanmasını istiyorsunuz, ancak birincil kuruluş birincil kuruluş dışındaki kuruluşla aynı sürümdedir.

DEVPATH değişkeninin nasıl güncelleneceği hakkında daha fazla bilgi için bkz. [“.NET uygulamalarının DEVPATH kullanılarak bağlanması” sayfa 14.](#)

.NET uygulamalarının uygulama yapılandırma dosyası kullanılarak bağlanması

Uygulama yapılandırma dosyası içinde, uygulamaları birincil kurulumdan olmayan düzenekleri kullanmak üzere yeniden yönlendirmek için çeşitli etiketler ayarlamamız gerekir.

Aşağıdaki çizelge, .NET uygulamalarının belirli yapıbirimleri kullanarak bağlanmasına olanak sağlamak için uygulama yapılandırma dosyasında yapılması gereken belirli değişiklikleri göstermektedir:

	Applications compiled with a lower version of IBM WebSphere MQ	Applications compiled with a higher version of IBM WebSphere MQ
Bir uygulamayı daha yüksek bir sürümle IBM WebSphere MQ birincil kuruluşu olan bir uygulamayı çalıştırmak için. (GAC ' de daha yüksek sürüm montajları):	Değişiklik gerekli değil	Değişiklik gerekli değil
Alt sürümdeki IBM WebSphere MQ birincil kuruluşu olan bir uygulamayı çalıştırmak için. (GAC ' de alt sürüm montajları):	Değişiklik gerekli değil	Uygulama yapılandırma dosyasında: <ul style="list-style-type: none">• GAC ' de yer alan düzeneklerin alt sürümünün kullanımını belirtmek için <code><bindingRedirect></code> etiketini kullanın.

Çizelge 2. Uygulamaların belirli düzenekleri kullanacak şekilde yapılandırılması (devamı var)

	Applications compiled with a lower version of IBM WebSphere MQ	Applications compiled with a higher version of IBM WebSphere MQ
IBM WebSphere MQ ' in birincil olmayan bir sürümü olan bir uygulamayı birincil olarak çalıştırmak için. (kuruluş klasöründeki daha yüksek sürüm yapıbirimleri):	Uygulama yapılandırma dosyasında: <ul style="list-style-type: none">• Daha yüksek sürüm düzeneklerinin yerini göstermek için <code><codebase></code> etiketini kullanın.• Daha yüksek sürüm düzeneklerinin kullanımını belirtmek için <code><bindingRedirect></code> etiketini kullanın.	Uygulama yapılandırma dosyasında: <ul style="list-style-type: none">• Daha yüksek sürüm düzeneklerinin yerini göstermek için <code><codebase></code> etiketini kullanın.
IBM WebSphere MQ birincil olmayan kuruluşun alt sürümüne sahip bir uygulamayı çalıştırmak için. (kuruluş klasöründeki alt sürüm yapıbirimleri):	Uygulama yapılandırma dosyasında: <ul style="list-style-type: none">• Alt sürüm düzeneklerinin yerini göstermek için <code><codebase></code> etiketini kullanın.• Include the tag <code><publisherpolicy Apply=no></code>	Uygulama yapılandırma dosyasında: <ul style="list-style-type: none">• Alt sürüm düzeneklerinin yerini göstermek için <code><codebase></code> etiketini kullanın.• Alt sürüm düzeneklerinin kullanımını belirtmek için <code><bindingRedirect></code> etiketini kullanın.• Include the tag <code><publisherpolicy Apply=no></code>

Örnek uygulama yapılandırma dosyası `NonPrimaryRedirect.config`, `MQ_INSTALLATION_PATH\tools\dotnet\samples\base` klasöründe teslim edilir. Bu dosya, birincil olmayan kuruluşa ilişkin IBM WebSphere MQ kuruluş yolu ile değiştirilebilir. Dosya, `<linkedConfiguration>` etiketini kullanan diğer yapılandırma kütüklerinde de doğrudan bulunabilir. Samples are provided for `nmqsget.exe.config` and `nmqsput.exe.config`. Her iki örnek de `<linkedConfiguration>` etiketini kullanır ve `NonPrimaryRedirect.config` dosyasını içerir.

.NET uygulamalarının DEVPATH kullanılarak bağlanması

You can find the assemblies using the `ÖLÜM` environment variable. `DEVPATH` değişkeni tarafından belirtilen düzenekler, GAC ' deki herhangi bir düzeneğin tercihinde kullanılır. See the appropriate Microsoft documentation on `ÖLÜM` for more information about when to use this variable.

To find the assemblies using the `ÖLÜM` environment variable, you must set the `ÖLÜM` variable to the folder that contains the assemblies you want to use. Daha sonra uygulama yapılandırma dosyasını güncellemeniz ve aşağıdaki yürütme ortamı yapılandırma bilgilerini eklemeniz gerekir:

```
<configuration>
  <runtime>
    <developmentMode developerInstallation="true" />
  </runtime>
</configuration>
```

İlgili kavramlar

[“Uygulamaların birden çok kuruluş ortamında bağlanması” sayfa 6](#)

UNIX, Linux, and Windows sistemlerinde, IBM WebSphere MQ Version 7.1ya da sonraki bir sürümü, kitaplıklar yüklenirse, IBM WebSphere MQ , başka herhangi bir işlem yapmak için gereken kitaplıkları

otomatik olarak kullanır. IBM WebSphere MQ Uygulamanın bağlandığı kuyruk yöneticisiyle ilişkilendirilmiş kuruluştan kitaplıkları kullanır.

[Birden çok kuruluş](#)

İlgili görevler

[Birincil kuruluş seçilmesi](#)

[.NET kullanımı](#)

Birincil kuruluş değiştiriliyor

Bir kuruluşu birincil kuruluş olarak ayarlamak ya da ayarlamayı kaldırmak için **setmqinst** komutunu kullanabilirsiniz.

Bu görev hakkında

Bu görev, UNIX, Linux, and Windows için geçerlidir.

Birincil kuruluş, gerekli sistem genelindeki konumların gönderme yaptığı kuruluştur. Birincil kuruluşla ilgili ek bilgi ve birincil kuruluşunuzu seçmekle ilgili dikkat edilmesi gereken noktalar için [Birincil kuruluşu seçme](#) başlıklı konuya bakın.

IBM WebSphere MQ Version 7.1 ya da daha sonraki bir kuruluş, IBM WebSphere MQ Version 7.0.1 kuruluşu ile birlikte var ise, IBM WebSphere MQ Version 7.0.1 kuruluşu birincil kullanıcı olmalıdır. IBM WebSphere MQ Version 7.1 ya da daha sonraki bir sürüm kurulduğunda birincil olarak işaretlenir ve IBM WebSphere MQ Version 7.1 ya da sonraki bir kuruluş birincil olarak kurulamaz.

Windows' ta kuruluş işlemi sırasında, kuruluşun birincil kuruluş olacağını belirtebilirsiniz. UNIX and Linux sistemlerinde, kuruluşu birincil kuruluş olarak ayarlamak için kuruluştan sonra bir **setmqinst** komutu yayınlamanız gerekir.

[“Birincil kuruluşu ayarlar” sayfa 15.](#)

[“Birincil kuruluşu ayarlamasını” sayfa 16.](#)

Birincil kuruluşu ayarlar

Yordam

Kuruluşu birincil kuruluş olarak ayarlamak için:

1. Aşağıdaki komutu girerek, kuruluşun önceden birincil kuruluş olup olmadığını denetleyin:

```
MQ_INSTALLATION_PATH/bin/dspmqinst
```

Burada `MQ_INSTALLATION_PATH` , IBM WebSphere MQ Version 7.1 ya da sonraki bir kuruluşun kuruluş yoludur.

2. Var olan bir IBM WebSphere MQ Version 7.1 ya da daha sonraki bir kuruluş birincil kuruluş olarak ayarlandıysa, [“Birincil kuruluşu ayarlamasını” sayfa 16](#) içindeki yönergeleri izleyerek ayarı kaldırın. If IBM WebSphere MQ Version 7.0.1 is installed on the system, the primary installation cannot be changed.

3. As root on UNIX and Linux systems, or a member of the Administrators group on Windows systems, enter one of the following commands:

- Birincil kuruluşu, birincil kuruluş olarak kullanmak istediğiniz kuruluş yolunu kullanarak ayarlamak için:

```
MQ_INSTALLATION_PATH/bin/setmqinst -i -p MQ_INSTALLATION_PATH
```

- Birincil kuruluşu, birincil kuruluş olarak kullanmak istediğiniz kuruluş adını kullanarak ayarlamak için:

```
MQ_INSTALLATION_PATH/bin/setmqinst -i -n installationName
```


4. Windows sistemlerinde sistemi yeniden başlatın.

Birincil kurulumu ayarlamasını

Yordam

Kurulumu birincil kurulum olarak kaldırmak için:

1. Aşağıdaki komutu girerek birincil kurulum olup olmadığını denetleyin:

```
MQ_INSTALLATION_PATH/bin/dspmqinst
```

Burada `MQ_INSTALLATION_PATH` , IBM WebSphere MQ Version 7.1 ya da sonraki bir kurulumun kurulum yoludur.

IBM WebSphere MQ Version 7.0.1 birincil kurulumu, birincil kurulumu yeniden ayarlayamazsınız.

2. As root on UNIX and Linux systems, or a member of the Administrators group on Windows systems, enter one of the following commands:

- Birincil kurulumun yolunu açmak için, artık birincil kurulum olmasını istemediğiniz kurulum yolunu kullanın:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -p MQ_INSTALLATION_PATH
```

- Birincil kurulumun adını kaldırmak için, artık birincil kurulum olmasını istemediğiniz kurulum adını kullanın:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -n installationName
```

İlgili kavramlar

[Yalnızca Windows üzerinde birincil kurulumla kullanılabilen özellikler](#)

[UNIX ve Linux üzerinde birincil kurulumla ilişkin dış kitaplık ve denetim komutu bağlantıları](#)

İlgili görevler

[Birincil kurulumun kaldırılması, büyütülmesi ve bakımının yapılması](#)

[Kurulum adı seçilmesi](#)

İlgili başvurular

[setmqinst](#)

Kuyruk yöneticisinin bir kurulumla ilişkilendirilmesi

Bir kuyruk yöneticisi yarattığınızda, bu yönetici otomatik olarak **crtmqm** komutunu veren kurulumla ilişkilendirilir. On UNIX, Linux, and Windows, you can change the installation associated with a queue manager using the **setmqm** command.

setmqm komutunu aşağıdaki şekillerde kullanabilirsiniz:

- Tek tek kuyruk yöneticilerinin eşdeğer WebSphere MQ sürümleri arasında taşınması. Örneğin, bir kuyruk yöneticisini testten üretim sistemine taşıma.
- Tek tek kuyruk yöneticilerinin daha eski bir WebSphere MQ sürümünden WebSphere MQ' nun daha yeni bir sürümüne geçirilmesi. Sürümler arasında kuyruk yöneticilerinin yeni düzeye geçirilmesi sizin bilmeniz gereken çeşitli etkilere sahiptir. Geçiş hakkında daha fazla bilgi için bkz. [Migrating and upgrading WebSphere MQ](#) konusuna bakın.

Bir kuyruk yöneticisini bir kurulumla ilişkilendirmek için:

1. Kuyruk yöneticisiyle ilişkilendirilmiş olan kurulumdan **endmqm** komutunu kullanarak kuyruk yöneticisini durdurun.
2. Kuyruk yöneticisini, bu kurulumdan **setmqm** komutunu kullanarak başka bir kurulumla ilişkilendirin.

For example, to set queue manager QMB to be associated with an installation with the name Installation2, enter the following command from Installation2:

```
MQ_INSTALLATION_PATH/bin/setmqm -m QMB -n Installation2
```

Burada `MQ_INSTALLATION_PATH` , Installation2 ' nin kurulu olduğu yoldur.

3. Start the queue manager using the **strmqm** command from the installation that is now associated with the queue manager.

Bu komut, gereken kuyruk yöneticisi geçişini gerçekleştirir ve kuyruk yöneticisinde kullanıma hazır olup olmadığını gösterir.

Kuyruk yöneticisinin ilişkilendirildiği kuruluş, kuyruk yöneticisini yalnızca o kuruluştan gelen komutlarla denetleyebilecek şekilde sınırlarla ilişkilendirir. Üç temel kural dışı durum vardır:

- **setmqm** , kuyruk yöneticisiyle ilişkili kuruluşu değiştirir. Bu komut, kuyruk yöneticisiyle ilişkilendirmek istediğiniz kuruluştan, kuyruk yöneticisinin şu anda ilişkili olduğu kuruluşu değil, bu komutu yayınlamalıdır. **setmqm** komutu tarafından belirtilen kuruluş adı, komutun yayınının verildiği kuruluşla eşleşmelidir.
- **strmqm** genellikle kuyruk yöneticisiyle ilişkili kuruluştan çıkılması gerekir. However, when a V7.0.1 or earlier queue manager is started on a V7.1 or later installation for the first time, **strmqm** can be used. Bu durumda, **strmqm** kuyruk yöneticisini başlatır ve bu komutu, komutun yayınının verildiği kuruluşla ilişkilendirir.
- **dspmq** , bir sistemdeki tüm kuyruk yöneticilerine ilişkin bilgileri görüntüler; yalnızca **dspmq** komutuyla aynı kuruluşla ilişkili kuyruk yöneticilerine değil, bir sistemdeki tüm kuyruk yöneticilerine ilişkin bilgileri görüntüler. `dspmq -o installation` komutu, hangi kuyruk yöneticilerinin hangi kuruluşlarla ilişkilendirildiği hakkında bilgi görüntüler.

HA ortamlarındaki kuyruk yöneticisi ilişkilendirmesi

HA ortamları için, **addmqinf** komutu kuyruk yöneticisini otomatik olarak, **addmqinf** komutunun yayınının yapıldığı kuruluşla ilişkilendirir. **strmqm** komutu, daha sonra **addmqinf** komutuyla aynı kuruluştan yayınlandığı sürece, başka bir kurulum gerekmez. Kuyruk yöneticisini farklı bir kuruluş kullanarak başlatmak için, önce **setmqm** komutunu kullanarak ilişkili kuruluşu değiştirmeniz gerekir.

Silinen kuruluşlarla ilişkili kuyruk yöneticileri

Bir kuyruk yöneticisinin ilişkilendirildiği kuruluş silindiye ya da kuyruk yöneticisi durum bilgisi kullanılamıyorsa, **setmqm** komutu kuyruk yöneticisini başka bir kuruluşla ilişkilendiremez. Bu durumda, aşağıdaki işlemleri gerçekleştirin:

1. Sisteminizdeki diğer kuruluşları görmek için **dspmqinst** komutunu kullanın.
2. Başka bir kuruluş belirtmek için `mqc.ini` içindeki `QueueManager` stanza alanının `InstallationName` alanını el ile değiştirin.
3. Kuyruk yöneticisini silmek için bu kuruluştan **dlmqm** komutunu kullanın.

İlgili kavramlar

[“Bir sistemde IBM WebSphere MQ kurulumları bulunması” sayfa 18](#)

Bir sistemde birden çok IBM WebSphere MQ kuruluşu varsa, hangi sürümlerin kurulu olduğunu ve nerede olduklarını denetleyebilirsiniz.

[“IBM WebSphere MQ yapılandırma dosyası, mqc.ini” sayfa 412](#)

IBM WebSphere MQ yapılandırma dosyası (`mqc.ini`), düğümdeki tüm kuyruk yöneticilerine ilişkin bilgileri içerir. Kuruluş sırasında otomatik olarak yaratılır.

İlgili görevler

[Birincil kuruluş seçilmesi](#)

İlgili başvurular

[setmqm](#)

[strmqm](#)
[dspmqr](#)
[dspmqrinst](#)

Bir sistemde IBM WebSphere MQ kurulumları bulunması

Bir sistemde birden çok IBM WebSphere MQ kurulumu varsa, hangi sürümlerin kurulu olduğunu ve nerede olduklarını denetleyebilirsiniz.

Sisteminizde IBM WebSphere MQ kurulumlarını bulmak için aşağıdaki yöntemleri kullanabilirsiniz:

- **dspmqr** komutunu kullanın. Bu komut, bir Version 7.0.1 kurulumundan yayınlandıysa, sistemdeki tüm kurulumlara ilişkin ayrıntıları sağlamaz.
- IBM WebSphere MQ ürününün kurulu olduğu yeri sorgulamak için platform kurulum araçlarını kullanın. Then use the **dspmqr** command from a Version 7.1 or later installation. The following commands are examples of commands you can use to query where IBM WebSphere MQ has been installed:

- AIX sistemlerinde **lspp** komutunu kullanabilirsiniz:

```
lspp -R ALL -l mqm.base.runtime
```

- HP-UX sistemlerinde **swlist** komutunu kullanabilirsiniz:

```
swlist -a location -a revision -l product MQSERIES
```

- Linux sistemlerinde **rpm** komutunu kullanabilirsiniz:

```
rpm -qa --qf "%{NAME}-%{VERSION}-%{RELEASE}\t%{INSTPREFIXES}\n" | grep MQSeriesRuntime
```

- Solaris sistemlerinde, **pkginfo** ve **pkgparam** komutlarını kullanabilirsiniz:

1. Aşağıdaki komutu girerek kurulu paketleri listele:

```
pkginfo | grep -w mqm
```

2. Listelenen her paket için şu komutu girin:

```
pkgparam pkgname BASEDIR
```

- Windows sistemlerinde, **wmic** komutunu kullanabilirsiniz. Bu komut, wmic istemcisini kurabilir:

```
wmic product where "(Name like '%MQ%') AND (not Name like '%bitSupport')\" get Name, Version, InstallLocation
```

- UNIX and Linux sistemlerinde, IBM WebSphere MQ ' in kurulduğu yeri bulmak için aşağıdaki komutu verin:

```
cat /etc/opt/mqm/mqmqrinst.ini
```

Daha sonra, Version 7.1 ya da sonraki bir kurulumdan **dspmqr** komutunu kullanın.

- Sistemdeki kurulumlara ilişkin ayrıntıları görüntülemek için, 32 bit Windows üzerinde aşağıdaki komutu verin:

```
reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation" /s
```

- 64-bit Windows işletim sistemi üzerinde şu komutu verin:

```
reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IBM\WebSphere MQ\Installation" /s
```

Not: **reg.exe** komutu yalnızca Version 7.1 kurulumlarına ya da sonraki kurulumlara ilişkin bilgileri görüntüler.

İlgili kavramlar

[Birden çok kurulum](#)

İlgili başvurular

[Dspmqr](#)

[dspmqrinst](#)

Kuyruk yöneticilerinin yaratılması ve yönetilmesi

İletileri ve kuyrukları kullanabilmeniz için önce en az bir kuyruk yöneticisi ve ilişkili nesnelere yaratmanız ve başlatmanız gerekir.

Kuyruk yöneticisi yaratılması

Bir kuyruk yöneticisi, özellikle sahip olduğu kuyruklar içinde, kendisiyle ilişkili kaynakları yönetir. It provides queuing services to applications for Message Queuing Interface (MQI) calls and commands to create, modify, display, and delete IBM WebSphere MQ objects.

To create a queue manager, use the IBM WebSphere MQ control command **crtmqm** (described in **crtmqm**). **crtmqm** komutu, gereken varsayılan nesnelere ve sistem nesnelere otomatik olarak yaratır (Sistem varsayılan nesnelere içinde açıklanmıştır). Varsayılan nesnelere, yaptığınız tüm nesne tanımlarının temelini oluşturur; kuyruk yöneticisi işlemi için sistem nesnelere gereklidir. Bir kuyruk yöneticisi ve nesnelere yarattığınız zaman, kuyruk yöneticisini başlatmak için **strmqm** komutunu kullanın.

Not: IBM WebSphere MQ , boşluk içeren makine adlarını desteklemez. If you install IBM WebSphere MQ on a computer with a machine name that contains spaces, you cannot create any queue managers.

Açık

Kuyruk yöneticisi yaratabilmek için, dikkate almanız gereken birkaç nokta vardır (özellikle de üretim ortamında). Aşağıdaki denetim listesi aracılığıyla çalışın:

Kuyruk yöneticisiyle ilişkili kuruluş

crtmqm komutu, bir kuyruk yöneticisini **crtmqm** komutunun yayınlandığı kuruluşla otomatik olarak ilişkilendirir. Kuyruk yöneticisiyle çalışan komutlar için, komutu kuyruk yöneticisiyle ilişkili kuruluştan yayınlamanız gerekir. **setmqm** komutunu kullanarak, kuyruk yöneticisinin ilişkili kuruluşunu değiştirebilirsiniz. Windows kuruluş programı, kuruluş işlemini gerçekleştiren kullanıcıyı **mqm** grubuna eklemeyiz; daha fazla bilgi için bkz. [UNIX, Linux ve Pencere sistemlerinde IBM WebSphere MQ yönetimi yetkisi](#).

Adlandırma kuralları

Tüm platformlarda kuyruk yöneticileriyle iletişim kurabilmeniz için büyük harfli adlar kullanın. İsimleri, tam olarak girdiğinizde atandığını unutmayın. Çok sayıda yazmanın rahatsızlıktan kaçınmak için, gereksiz uzun adlar kullanmayın.

Benzersiz bir kuyruk yöneticisi adı belirtin

Bir kuyruk yöneticisi yarattığınızda, ağızda başka bir kuyruk yöneticisinin aynı adı (*herhangi bir yerde*) sahip olmadığını doğrulayın. Kuyruk yöneticisi yaratıldığında, kuyruk yöneticisi adları denetlenmez; benzersiz olmayan adlar, dağıtılmış kuyruğa alma için kanal yaratmanıza engel olur.

Benzersizliği sağlamanın bir yolu, her kuyruk yöneticisi adının kendi benzersiz düğüm adıyla önekinin önlenmesinin bir yoludur. Örneğin, bir düğüm **ACCOUNTS** olarak adlandırılırsa, kuyruk yöneticinizi **ACCOUNTS.SATURN.QUEUE.MANAGER** olarak adlayabilirsiniz; burada **SATURN** , belirli bir kuyruk yöneticisini tanımlar ve **QUEUE.MANAGER** , tüm kuyruk yöneticilerine verebileceğiniz bir uzantıdır. Diğer bir seçenek olarak, bunu atlayabilirsiniz, ancak **ACCOUNTS.SATURN** ve **ACCOUNTS.SATURN.QUEUE.MANAGER** ' in *farklı* kuyruk yöneticisi adları olduğuna dikkat edin.

Diğer kuruluşlarla iletişim için IBM WebSphere MQ kullanıyorsanız, kendi kurumsal adınızı örnek olarak da ekleyebilirsiniz. Bu, örneklerde yapılmamaktadır, çünkü onları takip etmek daha da zorlaşmaktadır.

Not: Denetim komutlarındaki kuyruk yöneticisi adları büyük/küçük harfe duyarlıdır.

Bu, **jupiter.queue.manager** ve **JUPITER.queue.manager** adlarıyla iki kuyruk yöneticisi oluşturabilmenize izin verildiği anlamına gelir. Ancak, bu tür komplikasyonları önlemek daha iyi.

Kuyruk yöneticisi sayısını sınırla

Kaynak izin verdikçe, kuyruk yöneticisi olarak birden çok kuyruk yöneticisi yaratabilirsiniz. Ancak, her kuyruk yöneticisinin kendi kaynakları gerektirdiği için, her bir düğümde, her biri için on kuyruk yöneticisi bulunan on kuyruk yöneticisine sahip olmaktansa, bir düğüm üzerinde 100 kuyruk bulunan bir kuyruk yöneticisinin olması genellikle daha iyidir.

Üretim sistemlerinde, tek bir kuyruk yöneticisiyle birçok işlemci kullanılabilir, ancak daha büyük sunucu makineleri birden çok kuyruk yöneticisi ile daha etkili bir şekilde çalıştırılabilir.

Varsayılan kuyruk yöneticisi belirtin

Her düğümün bir varsayılan kuyruk yöneticisi olması gerekir; ancak, IBM WebSphere MQ düğümü bir düğüm olmadan bir düğüm için yapılandırılmalıdır. Varsayılan kuyruk yöneticisi, bir MQCONN çağrısında kuyruk yöneticisi adı belirtmemeleri durumunda, uygulamaların bağlan kuyruk yöneticidir. It is also the queue manager that processes MQSC commands when you invoke the runmqsc command without specifying a queue manager name.

Bir kuyruk yöneticisi varsayılan *yerine geçer* olarak belirtilirse, düğüme ilişkin var olan varsayılan kuyruk yöneticisi belirtilmelerinden biri olur.

Varsayılan kuyruk yönetimi 'in değiştirilmesi diğer kullanıcıları ya da uygulamaları etkileyebilir. Değişikliğin, şu anda bağlı olan uygulamalar üzerinde hiçbir etkisi yoktur, çünkü bunlar, diğer bir MQI çağrılarında özgün bağlantı çağrısından tanıtıcıyı kullanabilirler. Bu tanıtıcı, çağrılarının aynı kuyruk yöneticisine yönelmesini sağlar. *bundan sonra* ile bağlantı kuran tüm uygulamalar, varsayılan kuyruk yöneticisini yeni varsayılan kuyruk yöneticisine bağlamayı değiştirdiniz. Niyetiniz bu olabilir, ama varsayılanı değiştirmeden önce bunu dikkate almalısınız.

"Varsayılan kuyruk yöneticisi yaratılması" sayfa 21 içinde varsayılan bir kuyruk yöneticisi yaratılması anlatılır.

Bir ölü-mektup kuyruğu belirtin

Ölü-harfli kuyruk, iletilerin amaçlanan hedeflerine yönlendirilemezse, iletilerin yerleştirileceği bir yerel kuyruğudur.

Ağınızdaki her bir kuyruk yöneticisinde bir kuyruk-harf kuyruğunuzun olması önemlidir. Birini tanımlamadıysanız, uygulama programlarındaki hatalar kanalların kapatılmasına neden olabilir ve denetim komutlarına yanıt alınmayabilir.

Örneğin, bir uygulama başka bir kuyruk yöneticisine bir ileti koymaya çalışırsa, ancak yanlış kuyruk adı verirse, kanal durdurulur ve ileti iletim kuyruğunda kalır. Diğer uygulamalar bu kanalı kendi iletileri için kullanamaz.

Kuyruk yöneticilerinin ölü harf kuyrukları varsa, kanalların etkilenmemesi gerekir. Teslim edilmeyen ileti, alıcı uçtan alıcı uca yerleştirilir ve kanalı ve iletim kuyruğunu kullanılabilir kılar.

Bir kuyruk yöneticisi yarattığınızda, ölü-mektup kuyruğunun adını belirtmek için -u işaretini kullanın. Ayrıca, kullanılacak ölü-mektup kuyruğunu belirlemek üzere önceden tanımladığınız bir kuyruk yöneticisinin özniteliklerini değiştirmek için bir MQSC komutu da kullanabilirsiniz. MQSC komutunun bir örneği için Kuyruk yöneticileriyle çalışma başlıklı konuya bakın.

Varsayılan iletim kuyruğu belirle

İletim kuyruğu, aktarma işleminden önce uzak kuyruk yöneticilerine aktarılan iletilerin kuyruklanmış olduğu yerel bir kuyruktur. Varsayılan iletim kuyruğu, hiçbir iletim kuyruğu belirtik olarak tanımlanmadığında kullanılan kuyruğudur. Her kuyruk yöneticisine varsayılan bir iletim kuyruğu atayabilir.

Bir kuyruk yöneticisi yarattığınızda, varsayılan iletim kuyruğunun adını belirtmek için -d işaretini kullanın. Bu, kuyruğu gerçekte yaratmaz; bunu daha sonra açık bir şekilde yapmanız gerekir. Ek bilgi için Yerel kuyruklarla çalışma başlıklı konuya bakın.

Gerek duyduğunuz günlüğe kaydetme değiştiregelerini belirtin

You can specify logging parameters on the crtmqm command, including the type of logging, and the path and size of the log files.

Bir geliştirme ortamında, varsayılan günlüğe kaydetme parametreleri yeterli olmalıdır. Ancak, varsayılan değerleri değiştirebilirsiniz, örneğin:

- Büyük günlükleri destekleyemeyen, düşük uçlu bir sistem yapılanışınız var.
- Aynı anda kuyruklarınızda çok sayıda uzun ileti olup olmadığını tahmin etmiyorsunuz.
- Kuyruk yöneticiden geçen bir çok kalıcı ileti öngörüyor musunuz?

Günlüğe kaydetme deęiřtirgelerini ayarladıktan sonra, bunların bazıları yalnızca kuyruk yöneticisi silinerek ve aynı adla, ancak farklı günlük kaydı deęiřtirgeleriyle yeniden yaratılarak deęiřtirilebilir.

Parametreleri günlüğe kaydetme hakkında daha fazla bilgi için bkz. [“Kullanılabilirlik, kurtarma ve yeniden başlatma” sayfa 303.](#)

Yalnızca IBM WebSphere MQ for UNIX sistemleri için

You can create the queue manager directory `/var/mqm/qmgrs/<qmgr>`, even on a separate local file system, before you use the `crtmqm` command. `crtmqm` ı kullandığınızda, `/var/mqm/qmgrs/<qmgr>` dizini varsa, boşsa ve iyeliğinde `mqm` varsa, kuyruk yöneticisi verileri için kullanılır. Dizin `mqm` 'ye ait deęilse, yaratma işlemi bir First Failure Support Technology (FFST) iletiyle başarısız olur. Dizin boş deęilse, yeni bir izin yaratılır.

İlgili kavramlar

[“Yapılandırılıyor” sayfa 5](#)

Bir ya da daha çok bilgisayarda bir ya da daha çok kuyruk yöneticisi yaratın ve bunları geliştirme, test etme ve üretim sistemlerinizde, iş verilerinizi içeren iletileri işlemek için yapılandırın.

[“Kuyruk yöneticisi yaratıldıktan sonra yapılanış kütüklerinin yedeklenmesi” sayfa 23](#)

IBM WebSphere MQ yapılandırma bilgileri, Windows, UNIX and Linux sistemlerinde yapılandırma dosyalarında depolanır.

[“Kuyruk Yöneticisinin Başlatılması” sayfa 24](#)

Bir kuyruk yöneticisi yarattığınızda, komutları ya da MQI çağrılarını etkinleřtirmek için bir kuyruk yöneticisi başlatmalısınız.

[“Kuyruk Yöneticisinin Durdurulması” sayfa 24](#)

Kuyruk yöneticisini durdurmanın üç yolu vardır: Quiesced shutdown ve immediate shutdown, and a preemptive shutdown.

[“Kuyruk yöneticisinin yeniden başlatılması” sayfa 26](#)

Bir kuyruk yöneticisini yeniden başlatmak için **strmqm** komutunu kullanabilir ya da Linux (x86 ve x86-64 platformları) sistemlerinde IBM WebSphere MQ for Windows ve IBM WebSphere MQ platformlarında bir kuyruk yöneticisini IBM WebSphere MQ Explorer 'dan yeniden başlatın.

[“IBM WebSphere MQ ve kuyruk yöneticisi yapılanış bilgilerinin deęiřtirilmesi” sayfa 410](#)

Change the behavior of IBM WebSphere MQ or an individual queue manager to suit the needs of your installation.

[Sistem ve varsayılan nesnelere](#)

İlgili görevler

[“Var olan bir kuyruk yöneticisini varsayılan olarak oluřturma” sayfa 23](#)

Var olan bir kuyruk yöneticisini varsayılan kuyruk yöneticisi yapabilirsiniz. Bunu yapma řekliniz, kullandığınız platforma baęlıdır.

[“Kuyruk Yöneticisinin Silinmesi” sayfa 26](#)

You can delete a queue manager using the **dlmqm** command or by using the WebSphere MQ Explorer.

distributed Varsayılan kuyruk yöneticisi yaratılması

Varsayılan kuyruk yöneticisi, uygulamaların bir MQCONN çağrısında kuyruk yöneticisi adı belirtmemeleri durumunda baęlanacağı kuyruk yöneticidir. It is also the queue manager that processes MQSC commands when you invoke the **runmqsc** command without specifying a queue manager name. To create a queue manager, you use the IBM WebSphere MQ control command **crtmqm**.

Başlamadan önce

Varsayılan bir kuyruk yöneticisi yaratmadan önce, [“Kuyruk yöneticilerinin yaratılması ve yönetilmesi” sayfa 19](#) içinde açıklanan önemli noktaları okuyun.

UNIX UNIX and Linux üzerinde bir kuyruk yöneticisi yaratmak için **crtmqm** kullandığınızda, /var/mqm/qmgrs/<qmgr> dizini zaten varsa, bu dizin mqm 'dir ve boş olur, kuyruk yöneticisi verileri için kullanılır. Dizin mqm tarafından sahiplenmediyse, kuyruk yöneticisinin yaratılması First Failure Support Technology (FFST) iletilisiyle başarısız olur. Dizin boş değilse, kuyruk yöneticisi verileri için yeni bir dizin yaratılır.

Bu dikkat, /var/mqm/qmgrs/<qmgr> dizini ayrı bir yerel dosya sisteminde zaten varolduğunda da geçerlidir.

Bu görev hakkında

crtmqm komutunu kullanarak bir kuyruk yöneticisi yarattığınızda, komut gereken varsayılan nesnelere ve sistem nesnelere otomatik olarak yaratır. Varsayılan nesnelere, kuyruk yöneticisi işlemi için yaptığınız ve sistem nesnelere için gereken nesne tanımlamalarının temelini oluşturur.

Komutta ilgili deęiřtirgeleri de ekleyerek, örneęin, kuyruk yöneticisi tarafından kullanılacak varsayılan iletim kuyruęunun adını ve çıkış kuyruęu kuyruęunun adını da tanımlayabilirsiniz.

Windows Windows' ta, kuyruk yöneticisinin birden çok örneęini başlatmak için **crtmqm** komutunun **sax** seçeneęini kullanabilirsiniz.

crtmqm komutu ve sözdizimiyle ilgili daha fazla bilgi için bkz. [crtmqm](#).

Yordam

- Varsayılan kuyruk yöneticisi yaratmak için **crtmqm** komutunu **-q** işaretiyle kullanın. Aşağıdaki **crtmqm** komutu örneęi, SATURN . QUEUE . MANAGER adlı varsayılan bir kuyruk yöneticisi yaratır:

```
crtmqm -q -d MY.DEFAULT.XMIT.QUEUE -u SYSTEM.DEAD.LETTER.QUEUE SATURN.QUEUE.MANAGER
```

Burada:

-q

Bu kuyruk yöneticisinin varsayılan kuyruk yöneticisi olduğunu gösterir.

-d MY.DEFAULT.XMIT.QUEUE

Bu kuyruk yöneticisi tarafından kullanılacak varsayılan iletim kuyruęunun adı.

Not: IBM WebSphere MQ , sizin için varsayılan bir iletim kuyruęu oluşturmaz; bunu kendiniz tanımlamanız gerekir.

-u SYSTEM.DEAD.LETTER.QUEUE

Is the name of the default dead-letter queue created by IBM WebSphere MQ on installation.

SATURN.QUEUE.MANAGER

Bu kuyruk yöneticisinin adı. Bu parametre, **crtmqm** komutunda belirtilen son parametre olmalıdır.

Sonraki adım

When you have created a queue manager and its objects, use the **strmqm** command to [kuyruk yöneticisini başlatma](#).

İlgili kavramlar

“Kuyruk yöneticisi yaratıldıktan sonra yapılan iş kütüklerinin yedeklenmesi” sayfa 23

IBM WebSphere MQ yapılandırma bilgileri, Windows, UNIX and Linux sistemlerinde yapılandırma dosyalarında depolanır.

Kuyruk yöneticileriyle çalışılması

Yerel kuyruklarla çalışma

İlgili başvurular

Sistem ve varsayılan nesnelere

Var olan bir kuyruk yöneticisini varsayılan olarak oluşturma

Var olan bir kuyruk yöneticisini varsayılan kuyruk yöneticisi yapabilirsiniz. Bunu yapma şekliniz, kullandığınız platforma bağlıdır.

WebSphere MQ for Windows ve WebSphere MQ for Linux (x86 ve x86-64 platformları) sistemleri

Bu görev hakkında

Var olan bir kuyruk yöneticisi için WebSphere MQ for Pencere ve WebSphere MQ for Linux (x86 ve x86-64 platformları) sistemlerine ilişkin varsayılan kuyruk yöneticisini edinmek için aşağıdaki yönergeleri kullanın:

Yordam

1. IBM WebSphere MQ Explorer 'ı açın.
2. IBM WebSphere MQ nesnesini farenin sağ düğmesiyle tıklayın ve `Properties...` seçeneğini belirleyin. WebSphere MQ paneline ilişkin Özellikler görüntülenir.
3. Varsayılan kuyruk yöneticisi adı alanına varsayılan kuyruk yöneticisi adını yazın.
4. Tamam (OK) düğmesini tıklayın.

UNIX and Linux sistemleri

Bu görev hakkında

Varsayılan bir kuyruk yöneticisi oluşturduğunuzda, bu yönetici adı WebSphere MQ yapılandırma dosyasında (`mqs.ini`) `DefaultQueueManager` stanza 'nın Name özneliğine eklenir. Stanza ve içerikleri mevcut değilse otomatik olarak oluşturulur.

Yordam

- Var olan bir kuyruk yöneticisini varsayılan olarak yapmak için, Name öznelindeki kuyruk yöneticisi adını, yeni varsayılan kuyruk yöneticisinin adına çevirin. Bunu el ile, bir metin düzenleyicisi kullanarak yapabilirsiniz.
- Düğümde varsayılan bir kuyruk yöneticiniz yoksa ve varsayılan bir kuyruk yöneticisi oluşturmak istiyorsanız, varsayılan olarak gerekli adı taşıyan bir `DefaultQueueManager` stanza yaratın.
- Yanlışlıkla başka bir kuyruk yöneticisi yaparsanız ve özgün varsayılan kuyruk yöneticisine geri dönmek istiyorsanız, istenmeyen varsayılan kuyruk yöneticisini istediğiniz gibi değiştirerek, `mqs.ini` içindeki `DefaultQueueManager` stanza dosyasını düzenleyin.

Sonraki adım

Yapılandırma dosyalarına ilişkin bilgi için bkz. [“IBM WebSphere MQ ve kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi” sayfa 410](#).

Kuyruk yöneticisi yaratıldıktan sonra yapılanış kütüklerinin yedeklenmesi

IBM WebSphere MQ yapılandırma bilgileri, Windows, UNIX and Linux sistemlerinde yapılandırma dosyalarında depolanır.

Windows ve Linux (x86 ve x86-64) sistemlerinde, yapılandırma dosyalarında değişiklik yapmak için IBM WebSphere MQ Explorer ' i kullanın.

Windows sistemlerinde, yapılandırma dosyalarında değişiklik yapmak için amqmdain komutunu da kullanabilirsiniz. Bkz. [amqmdain](#)

İki tip yapılandırma dosyası vardır:

- Ürünü kurduğunuzda, IBM WebSphere MQ yapılandırma dosyası (mq5.ini) oluşturulur. Bir kuyruk yöneticisini her yarattığınızda ya da sildiğiniz her defasında güncellenen kuyruk yöneticilerinin listesini içerir. Düğüm başına bir mq5.ini dosyası var.
- Yeni bir kuyruk yöneticisi yarattığınızda, otomatik olarak yeni bir kuyruk yöneticisi yapılanış kütüğü (qm.ini) yaratılır. Kuyruk yöneticisine ilişkin yapılanış değiştirgelerini içerir.

Kuyruk yöneticisi yarattıktan sonra, yapılanış kütüklerinizi yedeklein. Daha sonra, sorun yaratan başka bir kuyruk yöneticisi yaraladığınızda, sorunun kaynağını kaldırdığınızda yedeklemeleri yeniden yürürlüğe girebilirsiniz. Genel bir kural olarak, yeni bir kuyruk yöneticisi yarattığınız her defasında yapılanış kütüklerinizi yedeklemenizi sağlar.

Yapılandırma dosyalarıyla ilgili daha fazla bilgi için bkz. "[IBM WebSphere MQ ve kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi](#)" sayfa 410.

Kuyruk Yöneticisinin Başlatılması

Bir kuyruk yöneticisi yarattığınızda, komutları ya da MQI çağrılarını etkinleştirmek için bir kuyruk yöneticisi başlatmalısınız.

Bir kuyruk yöneticisini başlatmak için **strmqm** komutunu kullanın.

Not: Çalışmakta olduğunuz kuyruk yöneticisiyle ilişkilendirilmiş kurulumdan **strmqm** komutunu kullanmanız gerekir. You can find out which installation a queue manager is associated with using the `dspmqs -o installation` command.

Örneğin, bir kuyruk yöneticisini başlatmak için QMB aşağıdaki komutu girin:

```
strmqm QMB
```

WebSphere MQ for Windows ve WebSphere MQ for Linux (x86 ve x86-64 altyapıları) sistemlerinde, aşağıdaki gibi bir kuyruk yöneticisi başlatabilirsiniz:

1. IBM WebSphere MQ Explorer 'ı açın.
2. Select the queue manager from the Navigator View.
3. Start tüğesini tıklatın. Kuyruk yöneticisi başlatılır.

Kuyruk yöneticisi başlatma işlemi birkaç saniyeden daha uzun sürerse WebSphere MQ , başlatma ilerleyişini ayrıntılı olarak açıklayan bilgi iletisi iletileri verir.

strmqm komutu, kuyruk yöneticisi çalışmaya başlayıncaya ve bağlantı isteklerini kabul etmeye hazır oluncaya kadar denetimi döndürmez.

Kuyruk yöneticisinin otomatik olarak başlatılması

In WebSphere MQ for Pencereler you can start a queue manager automatically when the system starts using the IBM WebSphere MQ Explorer. Daha fazla bilgi için bkz. [IBM WebSphere MQ Explorerkomutunu kullanarak yönetim](#).

Kuyruk Yöneticisinin Durdurulması

Kuyruk yöneticisini durdurmanın üç yolu vardır: Quiesced shutdown ve immediate shutdown, and a preemptive shutdown.

Kuyruk yöneticisini durdurmak için **endmqm** komutunu kullanın.

Not: Çalışmakta olduğunuz kuyruk yöneticisiyle ilişkilendirilmiş kurulumdan **endmqm** komutunu kullanmanız gerekir. You can find out which installation a queue manager is associated with using the `dspmqr -o` installation command.

Örneğin, QMBadlı bir kuyruk yöneticisini durdurmak için aşağıdaki komutu girin:

```
endmqm QMB
```

WebSphere MQ for Windows ve WebSphere MQ for Linux (x86 ve x86-64 platformları için) sistemlerinde, kuyruk yöneticisini aşağıdaki gibi durdurabilirsiniz:

1. IBM WebSphere MQ Explorer 'ı açın.
2. Select the queue manager from the Navigator View.
3. Stop . . . öğesini tıklayın. End Queue Manager panosu görüntülenir.
4. Denetlenen ya da Hemen seçeneğini belirleyin.
5. OKögesini tıklayın. Kuyruk yöneticisi durur.

Susturulmuş sona erdirme

Varsayılan olarak, **endmqm** komutu, belirtilen kuyruk yöneticisinin susturulmuş bir şekilde sona erdirilmesini sağlar. Bu işlem tamamlanacak bir süre alabilir. Tüm bağlı uygulamaların bağlantısı kesilinceye kadar, susturulmuş bir kapatma işlemi bekler.

Uygulamaların durmasını bildirmek için bu sona erdirme tipini kullanın. Bu sorunu ortaya çıkardığınızda:

```
endmqm -c QMB
```

Tüm uygulamalar durduğunda size söylenmez. (Bir `endmqm -c QMB` komutu, bir `endmqm QMB` komutuna eşdeğerdir.)

Ancak, aşağıdaki durumlarda sorun varsa:

```
endmqm -w QMB
```

komut, tüm uygulamalar duruncaya ve kuyruk yöneticisi sona erinceye kadar bekler.

Anında kapatma

Hemen sona erdirme işlemi için, yürürlükteki MQI çağrılarında izin verilir, ancak yeni çağrılar başarısız olur. Bu sona erdirme tipi, uygulamaların kuyruk yöneticisinden bağlantısını kesmesini beklemez.

Hemen sona erdirmek için şunu yazın:

```
endmqm -i QMB
```

Önleyici kapatma

Not: **endmqm** komutunu kullanarak kuyruk yöneticisini durdurma girişimleri başarısız olursa, bu yöntemi kullanmayın. Bu yöntem, bağlı uygulamalar için önceden kestirilemeyecek sonuçlar verebilir.

Hemen sona erdirme çalışmazsa, -p işaretini belirterek bir *önleyici* kapatmaya başvurmanız gerekir.

Örneğin:

```
endmqm -p QMB
```

Bu işlem kuyruk yöneticisini hemen durdurur. Bu yöntem yine de işe yaramazsa, alternatif bir çözüm için [Kuyruk yöneticisini el ile durdurma](#) başlıklı konuya bakın.

endmqm komutunun ve aksamlarının ayrıntılı bir açıklaması için bkz. [endmqm](#).

Kuyruk yöneticisini kapatmaya ilişkin sorunlarınız varsa

Kuyruk yöneticisini kapatma sırasında sorunlar genellikle uygulamalar tarafından neden kaynaklanır. Örneğin, uygulamalar:

- MQI dönüş kodlarını düzgün bir şekilde işaretlemeyin
- Susturma bildirimini istenmemesi
- Kuyruk yöneticisinden bağlantıyı kesmeden sona erdir (bir MQDISC çağrısı yayınlanarak)

Kuyruk yöneticisini durdurduğunuzda bir sorun oluşursa, Ctrl-C ' yi kullanarak **endmqm** komutundan ayrılabilirsiniz. Daha sonra başka bir **endmqm** komutu yayınlayabilirsiniz, ancak bu kez gereksinim duyduğunuz sona erdirmeye ilişkin bir işarette birlikte bu işaret kullanılabilir.

Kuyruk yöneticisinin yeniden başlatılması

Bir kuyruk yöneticisini yeniden başlatmak için **strmqm** komutunu kullanabilir ya da Linux (x86 ve x86-64 platformları) sistemlerinde IBM WebSphere MQ for Windows ve IBM WebSphere MQ platformlarında bir kuyruk yöneticisini IBM WebSphere MQ Explorer 'dan yeniden başlatın.

Kuyruk yöneticisini yeniden başlatmak için şunu yazın:

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

Linux (x86 ve x86-64 platformları) sistemlerinde IBM WebSphere MQ for Windows ve IBM WebSphere MQ üzerinde, bir kuyruk yöneticisini başlatma işlemini başlatırken aşağıdaki gibi yeniden başlatabilirsiniz:

1. IBM WebSphere MQ Explorer 'ı açın.
2. Select the queue manager from the Navigator View.
3. Start öğesini tıklayın. Kuyruk yöneticisi yeniden başlatılır.

Kuyruk yöneticisinin yeniden başlatılması birkaç saniyeden daha uzun sürerse IBM WebSphere MQ , başlatma ilerlemesine ilişkin bilgi iletileri aralıklarla devam eder.

Kuyruk Yöneticisinin Silinmesi

You can delete a queue manager using the **dltmqm** command or by using the WebSphere MQ Explorer.

Başlamadan önce

Kuyruk yöneticisini durdurun.

Yordam

- Şu komutu verin: **dltmqm QMB**

Not: Çalışmakta olduğunuz kuyruk yöneticisiyle ilişkilendirilmiş kurulumdan **dltmqm** komutunu kullanmanız gerekir. You can find out which installation a queue manager is associated with using the **dspmqs -o installation** command.

Kuyruk yöneticisinin silinmesine ilişkin adımlar

Bu görev hakkında

WebSphere MQ for Windows ve WebSphere MQ for Linux (x86 ve x86-64 altyapıları) sistemlerinde, bir kuyruk yöneticisini aşağıdaki gibi silebilirsiniz:

Yordam

1. WebSphere MQ Explorer olanağını açın.
2. Navigator görünümünde kuyruk yöneticisini seçin.

3. Kuyruk yöneticisi durdurulmamışsa, durdurun.
 - a) Kuyruk yöneticisini sağ tıklayın.
 - b) **Durdurdüğmesini** tıklayın.
4. Kuyruk yöneticisini sağ tıklayın.
5. **Sil** düğmesini tıklayın.

Sonuçlar

Kuyruk yöneticisi silindi.



Uyarı:

- Kuyruk yöneticisi silindiğinde, kuyruk yöneticisiyle ilişkili tüm kaynakları da sildiğiniz için, kuyruk yöneticisi tüm kuyruklar ve bunların iletileri ve tüm nesne tanımlamaları da içinde olmak üzere, köklü bir adımdır. **dltmqm** komutunu kullanırsanız, fikrinizi değiştirmenize olanak tanıyan bir bilgi istemi görüntülenmez; Enter tuşuna bastığınızda ilişkili kaynakların tümü kaybedilir.
- Pencereler için WebSphere MQ 'ta bir kuyruk yöneticisi silinerek, kuyruk yöneticisini otomatik başlatma listesinden de kaldırır ("Kuyruk Yöneticisinin Başlatılması" sayfa 24' ta açıklanmıştır). Komut tamamlandığında, bir WebSphere MQ queue manager ending ileti görüntülenir; kuyruk yöneticisinin silindiği söylenmez.
- Bir küme kuyruk yöneticisinin silinmesi, küme kuyruk yöneticisinin kümeden kaldırılmasına neden olmaz. Ek bilgi için **dltmqm** açıklamasındaki notta bakın.


dltmqm komutunun ve aksamlarının açıklaması için bkz. [dltmqm](#). Yalnızca güvenilir denetimcilerin bu komutu kullanma yetkisine sahip olduğundan emin olun. (Güvenlikle ilgili bilgi için bkz. [Pencereler, UNIX and Linux sistemleri üzerinde güvenliğin ayarlanması](#) .)

Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eşitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluşları arasındaki iletişim ile ilgili daha ayrıntılı bilgi sağlanmaktadır.

Before reading this section it is helpful to have an understanding of channels, queues, and the other concepts introduced in [İletişim kavramları](#) .

Dağıtılmış kuyruklama kullanarak uygulamalarınızı bağlamak için aşağıdaki bağlantılarda yer alan bilgileri kullanın:

- ["Başka bir kuyruk yöneticiye ileti gönderilmesi" sayfa 49](#)
- ["Tetikleme kanalları" sayfa 66](#)
- ["İletilerin güvenliği" sayfa 64](#)
- ["IBM WebSphere MQ dağıtılmış-ileti sistemi teknikleri" sayfa 28](#)
- ["Dağıtılmış kuyruk yönetimine giriş" sayfa 47](#)
-  ["UNIX, Linux, and Windows üzerindeki kanalların izlenmesi ve denetlenmesi" sayfa 72](#)

İlgili kavramlar

["İstemci ile sunucu arasındaki bağlantıların yapılandırılması" sayfa 96](#)

WebSphere MQ MQI istemcileri ve sunucuları arasındaki iletişim bağlantılarını yapılandırmak için iletişim protokolünüze karar verin, bağlantının her iki ucundaki bağlantıları tanımlayın, bir dinleyici başlatın ve kanalları tanımlayın.

["IBM WebSphere MQ ve kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi" sayfa 410](#)

Change the behavior of IBM WebSphere MQ or an individual queue manager to suit the needs of your installation.

İlgili görevler

["Kuyruk yöneticisi kümesinin yapılandırılması" sayfa 156](#)

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

IBM WebSphere MQ dağıtılmış-ileti sistemi teknikleri

Bu bölümdeki alt konular, kanal planlaması yaparken kullanılan teknikleri açıklar. Bu alt konularda, kuyruk yöneticinizin nasıl bağlanacağını planlamanıza yardımcı olacak ve uygulamalarınız arasındaki ileti akışını planlamanıza yardımcı olacak teknikler açıklanmaktadır.

İleti kanalı planlama örnekleri için aşağıdaki başlara bakın:

- [Dağıtılmış platformlar için ileti kanalı planlama örneği](#)

İlgili kavramlar

[“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27](#)

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eşitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluşları arasındaki iletişim ile ilgili daha ayrıntılı bilgi sağlanmaktadır.

[Kanallar](#)

[İletinin kuyruğa alınması](#)

[İletişim kavramları](#)

İlgili başvurular

[Örnek yapılanış bilgileri](#)

İleti akışı denetimi

İleti akışı denetimi, kuyruk yöneticileri arasında ileti rotalarının ayarlanmasını ve bakımının içinde yer alan bir görevdir. Çok sekmeli bir çok sayıda kuyruk yöneticisi için bu, çok sekmeli yollar için önemlidir. Bu kısımda, ileti akışı denetimini gerçekleştirmek için sisteminizdeki kuyruklar, diğer ad kuyruğu tanımları ve ileti kanallarının nasıl kullanılacağı ele alınmıştır.

İleti akışını, [“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27](#) içinde kullanıma sunulan bir dizi teknik kullanarak kontrol edin. Kuyruk yöneticiniz bir kümedeyse, ileti akışı [“İleti akışı denetimi” sayfa 28](#) içinde açıklandığı gibi farklı teknikler kullanılarak denetlenir.

İleti akışı denetimini gerçekleştirmek için aşağıdaki nesnelere kullanabilirsiniz:

- İletim kuyrukları
- İleti kanalları
- Uzak kuyruk tanımlaması
- Kuyruk yöneticisi diğer adı tanımlaması
- Yanıtlama kuyruğu diğer adı tanımlaması

Kuyruk yöneticisi ve kuyruk nesnelere Nesnelere ' de anlatılır. İleti kanalları [Dağıtılmış kuyruğa alma bileşenlerinde](#) açıklanmıştır. Aşağıdaki teknikler, sisteminizde ileti akışları yaratmak için bu nesnelere kullanılır:

- İletileri Uzak Kuyruklara Koyma
- Belirli iletim kuyrukları yoluyla yönlendirme
- İletileri alma
- İletilerin sisteminiz aracılığıyla geçirilmesi
- İleti akışlarını ayırma
- İleti akışının başka bir hedefe çevrilmesi
- Yanıt kuyruğu adının diğer ad adına çözülmesi

Not

Bu bölümde anlatılan tüm kavramlar, bir ağdaki tüm düğümler için geçerlidir ve ileti kanallarının gönderilmesi ve alınması da içerilir. Bu nedenle, çoğu örnekte yalnızca bir düğüm gösterilir. Bu kural dışı durum, örnek bir ileti kanalının diğer ucunda denetimcinin belirttik olarak işbirliği yapılmasını gerektirdiği durumlarda.

Tek tek tekniklere geçmeden önce, ad çözme kavramlarını ve uzak kuyruk tanımlarını kullanmanın üç yolunu yeniden özetlemek yararlı olur. Bkz. [İletişim kavramları](#).

İlgili kavramlar

[“İletim üstbilgisindeki kuyruk adları” sayfa 29](#)

Hedef kuyruk adları, hedef kuyruğa ulaşıncaya kadar iletim üstbilgisinde iletiyle birlikte hareket eder.

[“Kuyruk yöneticisi ve yanıtın diğer adlara nasıl yaratılacağı” sayfa 29](#)

Bu konuda, uzak kuyruk tanımlaması yaratabileceğiniz üç yol açıklanmaktadır.

İletim üstbilgisindeki kuyruk adları

Hedef kuyruk adları, hedef kuyruğa ulaşıncaya kadar iletim üstbilgisinde iletiyle birlikte hareket eder.

Uygulama tarafından kullanılan kuyruk adı, mantıksal kuyruk adı, kuyruk yöneticisi tarafından hedef kuyruk adına çözülebilir. Diğer bir deyişle, fiziksel kuyruk adı. Bu hedef kuyruk adı, hedef kuyruğa ulaşıncaya kadar, ileti, ayrı bir veri alanında, iletim üstbilgisinde iletilerle birlikte hareket eder. Daha sonra, iletim üstbilgisi çıkarılır.

Paralel hizmet sınıflarını yaratırken bu kuyruk adının kuyruk yöneticisi kısmını değiştirdiniz. Hizmet sınıfı şartmalarının sonuna ulaşıldığında kuyruk yöneticisi adını özgün adına döndürmeyi unutmayın.

Kuyruk yöneticisi ve yanıtın diğer adlara nasıl yaratılacağı

Bu konuda, uzak kuyruk tanımlaması yaratabileceğiniz üç yol açıklanmaktadır.

Uzak kuyruk tanımlaması nesnesi üç farklı şekilde kullanılır. [Çizelge 3 sayfa 30](#) Bu üç yoldan her birinin nasıl tanımlanacak açıklanır:

- Yerel kuyruk adını yeniden tanımlamak için uzak kuyruk tanımlaması kullanılıyor.

Uygulama, kuyruk açılırken yalnızca kuyruk adı sağlar ve bu kuyruk adı, uzak kuyruk tanımlamasının adıdır.

Uzak kuyruk tanımlaması, hedef kuyruğun ve kuyruk yöneticisinin adlarını içerir. İsteğe bağlı olarak, tanımlama, kullanılacak iletim kuyruğunun adını içerebilir. İletim kuyruğu adı sağlanmazsa, kuyruk yöneticisi iletim kuyruğu adı için, uzak kuyruk tanımlamasından alınan kuyruk yöneticisi adını kullanır. Bu adda bir iletim kuyruğu tanımlanmadıysa, ancak varsayılan iletim kuyruğu tanımlıysa, varsayılan iletim kuyruğu kullanılır.

- Kuyruk yöneticisi adını yeniden tanımlamak için uzak kuyruk tanımlaması kullanılıyor.

Uygulama ya da kanal programı, kuyruğu açarken uzak kuyruk yöneticisi adı ile birlikte kuyruk adı sağlar.

Kuyruk yöneticisi adıyla aynı adı taşıyan bir uzak kuyruk tanımlaması sağladıysanız ve bu tanımda kuyruk adını boş bıraktığınızda, kuyruk yöneticisi, açık çağrıdaki kuyruk yöneticisi adını tanımda kuyruk yöneticisi adıyla değiştirir.

Ayrıca, tanımlama, kullanılacak iletim kuyruğunun adını da içerebilir. İletim kuyruğu adı sağlanmazsa, kuyruk yöneticisi iletim kuyruğu adı için kuyruk yöneticisi adını uzak kuyruk tanımlamasından alır. Bu adda bir iletim kuyruğu tanımlanmadıysa, ancak varsayılan iletim kuyruğu tanımlıysa, varsayılan iletim kuyruğu kullanılır.

- Bir yanıtlama kuyruğu adını yeniden tanımlamak için uzak kuyruk tanımlaması kullanılıyor.

Bir uygulamanın kuyruğa her ileti yerleştirdiği her defasında, yanıt iletileri için bir yanıtlama kuyruğu adı verebilir, ancak kuyruk yöneticisi adı boş olur.

Yanıt kuyruğunda aynı adı taşıyan bir uzak kuyruk tanımlaması sağlıyorsanız, yerel kuyruk yöneticisi yanıtın yerini, tanımınızdaki kuyruk adıyla olan yanıtlama kuyruğu adını alır.

Tanımda bir kuyruk yöneticisi adı sağlayabilirsiniz, ancak iletim kuyruğu adı giremez.

Çizelge 3. Uzak kuyruk tanımlaması nesnesini kullanmanın üç yolu			
Kullanım	Kuyruk yöneticisi adı	Kuyruk adı	İletim kuyruğu adı
1. Uzak kuyruk tanımlaması (AÇIK arama sırasında)			
Çağrıya verildi	boş ya da yerel QM	(*) gerekli	-
Tanımda sağlanan	zorunlu	zorunlu	isteğe bağlı
2. Kuyruk yöneticisi diğer adı (AÇIK arama için)			
Çağrıya verildi	(*) yerel QM için gerekli ve değil	zorunlu	-
Tanımda sağlanan	zorunlu	blank (boşluk)	isteğe bağlı
3. Yanıtla-Kuyruk diğer adı (PUT çağrısında)			
Çağrıya verildi	blank (boşluk)	(*) gerekli	-
Tanımda sağlanan	isteğe bağlı	isteğe bağlı	blank (boşluk)
Not: (*) bu ismin, tanım nesnesinin adı olduğu anlamına gelir.			

Resmi bir açıklama için bkz. [Kuyruk adı çözünürlüğü](#).

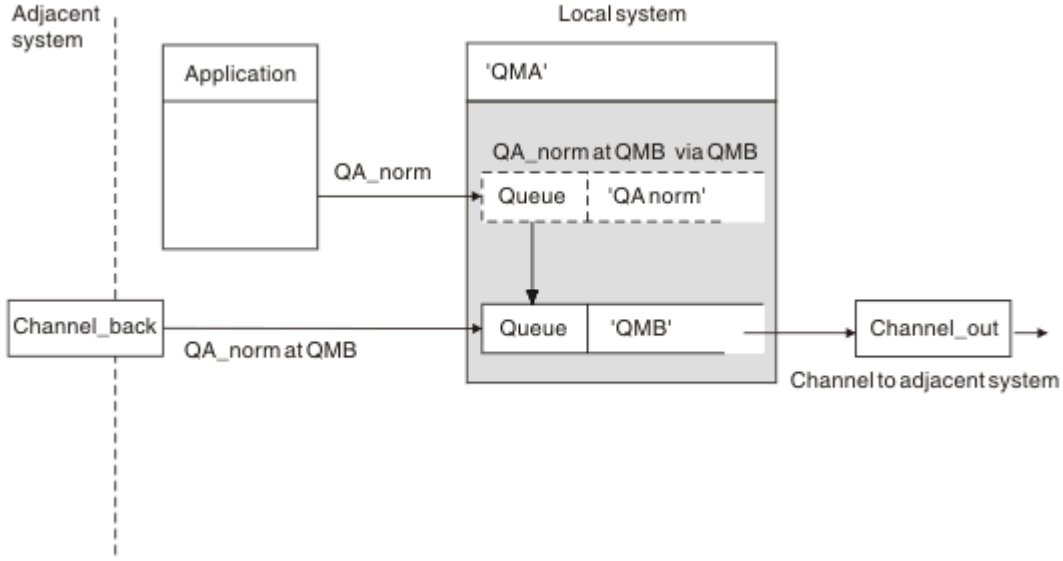
İletileri Uzak Kuyruklara Koyma

Bir kuyruk adını, bitişik bir kuyruk yöneticisine bir kuyruk adını çözmek için uzak kuyruk tanımlama nesnelere kullanabilirsiniz.

Dağıtılmış kuyruklama ortamında, iletim kuyruğu ve kanalı, iletilerin yerel sisteminizdeki uygulamalardan kaynaklanıp kaynaklanmadığını ya da bitişik bir sistemden gelen kanallara ulaşmasını sağlamak için tüm iletilerin odak noktasıdır. Şekil 2 sayfa 31 'QA_norm' adlı mantıksal bir kuyruğa ileti yerleştiren bir uygulama gösterir. Ad çözünürlüğü, QMB iletim kuyruğunu seçmek için 'QA_norm' uzak kuyruk tanımını kullanır. Daha sonra, 'QMB' de QA_norm ' belirten iletilere bir iletim üstbilgisi ekler.

'Channel_back' üzerindeki bitişik sistemden gelen iletiler, örneğin, fiziksel kuyruk adı 'QA_norm/QMB' olan bir iletim üstbilgisine sahip olur. Bu iletiler, QMB iletim kuyruğunda değiştirilmeden yerleştirilir.

Kanal, iletileri bitişik bir kuyruk yöneticisine taşır.



Şekil 2. Uzak kuyruk tanımlaması, bir kuyruk adını bitişik bir kuyruk yöneticisine aktarma kuyruğuna çözmek için kullanılır.

WebSphere MQ sistem denetimcisiyseniz, aşağıdakileri yapmak gerekir:

- İleti kanalını bitişik sistemden tanımla
- İleti kanalını bitişik sisteme tanımlayın
- İletim kuyruğunda QMB ' nin yaratılması
- Uygulamalar tarafından hedef kuyruk adı, hedef kuyruk yöneticisi adı ve iletim kuyruğu adına göre kullanılan kuyruk adını çözmek için 'QA_norm' uzak kuyruk nesnesini tanımlayın.

Kümeleme ortamında, yalnızca yerel kuyruk yöneticisinde bir küme alıcılı kanal tanımlamanız gerekir. Bir iletim kuyruğu ya da uzak kuyruk nesnesi tanımlamanıza gerek yoktur. Bilgi için bkz. [Kümeler](#).

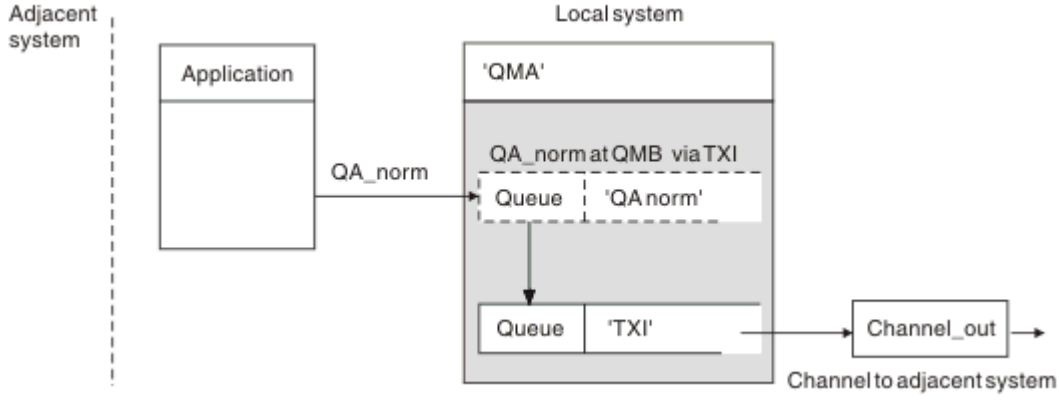
Ad çözme hakkında daha fazla bilgi

Uzak kuyruk tanımlamasının etkisi, bir fiziksel hedef kuyruğu adı ve kuyruk yöneticisi adı tanımlamalıdır. Bu adlar, iletilerin iletim üstbilgilerine yerleştirilir.

Bitişik bir sistemdeki gelen iletiler, özgün kuyruk yöneticisi tarafından yürütülen bu tip bir çözme tipine zaten sahip. Bu nedenle, fiziksel hedef kuyruğu adını ve kuyruk yöneticisi adını gösteren iletim üstbilgisini elde ederler. Bu iletiler uzak kuyruk tanımlarından etkilenmez.

İletim kuyruğunun seçilmesi

Uzak kuyruk tanımlamasını, aynı bitişik kuyruk yöneticisine ileti göndermek için farklı bir iletim kuyruğuna olarak tanıyacak şekilde kullanabilirsiniz.



Şekil 3. Uzak kuyruk tanımlaması, farklı bir iletim kuyruğunun kullanılmasına izin verir.

Dağıtılmış bir kuyrukte, bir ileti akışını bir kanaldan diğerine değiştirmeniz gerektiğinde, “İletileri Uzak Kuyruklara Koyma” sayfa 30’ünde Şekil 2 sayfa 31’inde gösterildiği gibi aynı sistem yapılandırmasını kullanın. Bu konuda Şekil 3 sayfa 32, iletileri farklı bir iletim kuyruğuna farklı bir kanal üzerinden, aynı bitişik kuyruk yöneticisine göndermek için uzak kuyruk tanımlamasını nasıl kullandığınızı gösterir.

Şekil 3 sayfa 32’inde gösterilen yapılandırma için, 'QA_norm' adlı uzak kuyruk nesnesini ve 'TX1' iletim kuyruğunu sağlamanız gerekir. Uzak kuyruk yöneticisinde 'QA_norm' kuyruğunu, 'TX1' iletim kuyruğunu ve 'QMB_priority' kuyruk yöneticisini seçmek için 'QA_norm' değerini belirtmeniz gerekir. Sistemin bitişikindeki kanal tanımında 'TX1' değerini belirtin.

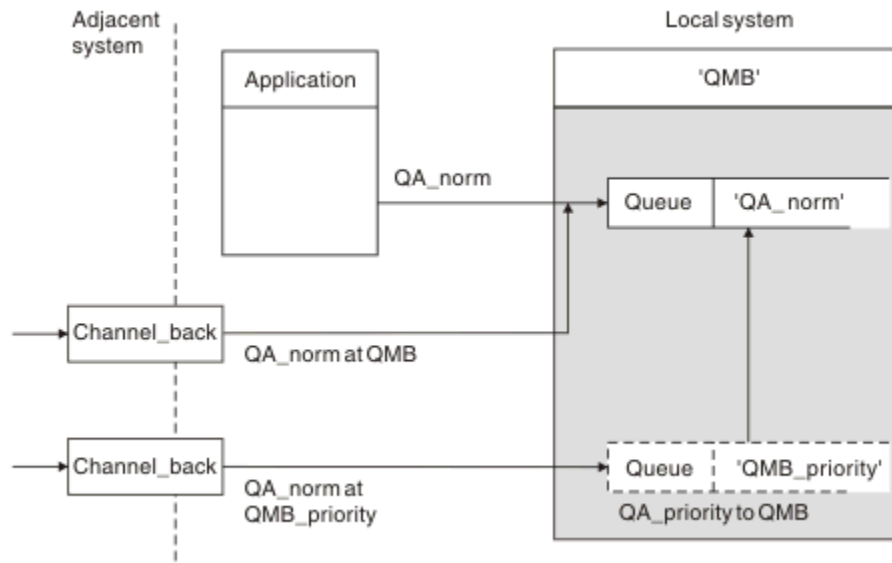
İletiler 'TX1' iletim kuyruğuna yerleştirilir; 'QMB_priority' QA_NORM ' ögesini içeren bir iletim üstbilgisi gönderilir ve kanal üzerinden bitişik sisteme gönderilir.

Kuyruk yöneticisi diğer adı gerekeceği için, channel_back bu şeklin dışında bırakıldı.

Kümeleme ortamında, bir iletim kuyruğu ya da uzak kuyruk tanımlaması tanımlamanıza gerek yoktur. Daha fazla bilgi için “Küme kuyrukları” sayfa 162 başlıklı konuya bakın.

İletileri alma

Kuyruk yöneticisini, diğer kuyruk yöneticilerinden gelen iletileri almak için yapılandırabilirsiniz. Amaçsız ad çözme işlemi oluşmadığından emin olmalısınız.



Şekil 4. Doğrudan iletiler alınıyor ve diğer ad kuyruk yöneticisi adı çözülüyor

İletiler gönderileceği gibi, sistem denetimcisinin de bitişik kuyruk yöneticilerinden alınacak iletileri düzenlemesini de gerekir. Alınan iletiler, iletim üstbilgisinde hedef kuyruk yöneticisinin ve kuyruğun fiziksel adını içerir. Bunlar, hem kuyruk yöneticisi adını, hem de kuyruk adını belirten yerel bir uygulamadan gelen iletiyle aynı şekilde davranılır. Bu tedaviden dolayı, sisteminize giren iletilerin, kasıtlı olarak gerçekleştirilen bir ad çözme işlemi gerçekleştirmediğinden emin olmanız gerekir. Bu senaryo için bkz. [Şekil 4 sayfa 32](#).

Bu yapılandırma için şunları hazırlamanız gerekir:

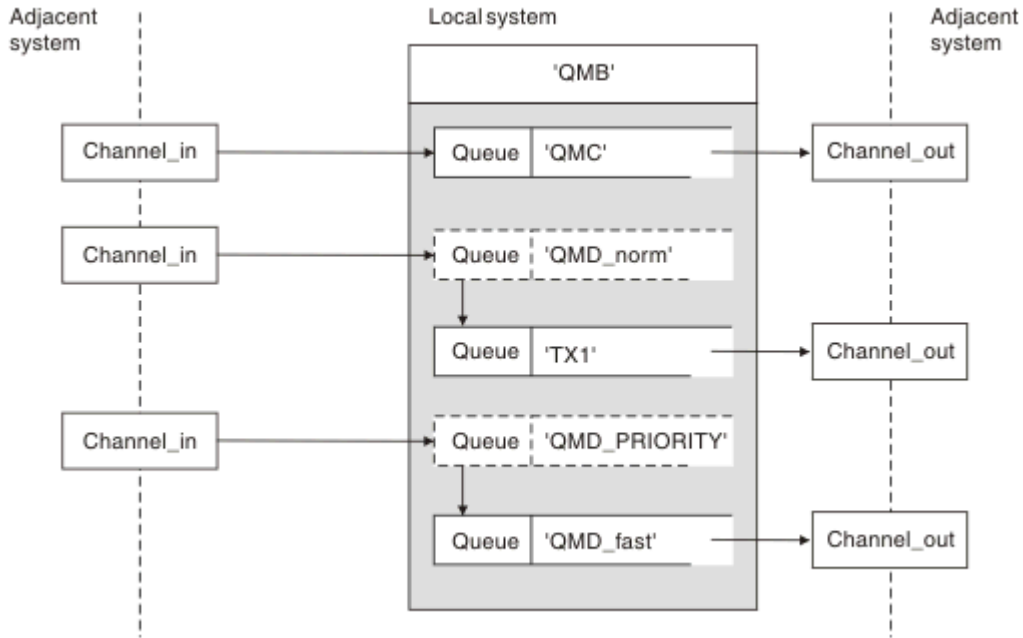
- Bitişik kuyruk yöneticilerinden ileti almak için ileti kanalları
- Gelen ileti akışını ('QMB_priority'), yerel kuyruk yöneticisi adına ('QMB') çözümlemek için bir kuyruk yöneticisi diğer adı tanımlaması
- Yerel kuyruk ('QA_norm') yoksa,

Alıcı diğer ad kuyruk yöneticisi adları alınıyor

Bu şekildeki kuyruk yöneticisi diğer adı tanımlamasının kullanımı farklı bir hedef kuyruk yöneticisi seçmedi. Bu yerel kuyruk yöneticiden geçen ve 'QMB_priority' olarak adreslenen iletiler kuyruk yöneticisi 'QMB' için amaçlanır. Diğer ad kuyruk yöneticisi adı, ayrı ileti akışını yaratmak için kullanılır.

İletilerin sisteminiz aracılığıyla geçirilmesi

Kuyruk yöneticisi için bir diğer ad kullanarak ya da bir iletim kuyruğu seçerek, sistem üzerinden iletileri üç şekilde kullanarak sistem üzerinden iletebilirsiniz.



Şekil 5. İletilerin sisteminiz aracılığıyla iletilmesinin üç yöntemi

“İletileri alma” sayfa 32’de Şekil 4 sayfa 32’de gösterilen teknik, bir diğer ad akışının nasıl yakalandığını gösterdi. Şekil 5 sayfa 33’de anlatılan teknikleri bir araya getirerek nasıl oluşturulabileceğini göstermektedir.

Yapılandırma, farklı hedefler içeren üç ileti gönderen bir kanalı gösterir:

1. QB saat: QMC
2. QB saat: QMD_norm
3. QB saat: QMD_PRIORITY

İlk ileti akışını, sistem tarafından değiştirilmeden geçirmeniz gerekir. İkinci ileti akışını farklı bir iletim kuyruğu ve kanaldan geçirmeniz gerekir. For the second message flow you must also resolve messages for the alias queue manager name QMD_noım to the queue manager QMD. Üçüncü ileti akışı, başka bir değişiklik olmadan farklı bir iletim kuyruğu seçiyor.

Kümeleme ortamında, iletiler bir küme iletim kuyruğundan geçirilir. Olağan durumda tek bir iletim kuyruğu (SYSTEM. CLUSTER. TRANSMIT. QUEUE), tüm iletileri kuyruk yöneticisinin üyesi olduğu tüm kümelerdeki tüm kuyruk yöneticilerine aktarır; bkz. [Bir kuyruk yöneticisi kümesi](#). Kuyruk yöneticisinin üyesi olduğu kümelerdeki kuyruk yöneticilerinin tümü ya da bazıları için ayrı iletim kuyrukları tanımlayabilirsiniz.

Aşağıdaki yöntemlerde, dağıtılmış kuyruğa alma ortamı için geçerli olan teknikler açıklanmaktadır.

Bu yöntemleri kullan

Bu yapılandırmalar için şunları hazırlamanız gerekir:

- Giriş kanalı tanımlamaları
- Çıkış kanalı tanımlamaları
- İletim kuyrukları:
 - QMC
 - TX1
 - QMD_fast
- Kuyruk yöneticisi diğer ad tanımlamaları:
 - QMD_noım ile QMD_noım , QMD - TX1arasında
 - QMD_PRIORITY ile QMD_PRIORITY , QMD_PRIORITY - QMD_fastarasında

Not: Örnekte gösterilen ileti akışlarının hiçbiri hedef kuyruğu değiştirmiyor. Kuyruk yöneticisi adı diğer adları ileti akışlarının ayrılmasını sağlar.

Yöntem 1: Gelen yer adını kullan

Başka bir yer adı (QMCgibi) içeren bir iletim üstbilgisiyle ileti alırsınız. En basit yapılandırma, bu adı taşıyan bir iletim kuyruğu oluşturmaktan QMC. İletim kuyruğunda hizmet veren kanal, iletiyi değiştirmeden sonraki hedefe aktarır.

Yöntem 2: Kuyruk yöneticisi için diğer ad kullan

İkinci yöntem ise, kuyruk yöneticisi diğer ad nesnesi tanımlamasını kullanmakla birlikte, yeni bir yer adı (QMD) ve belirli bir iletim kuyruğunu (TX1) belirtin. Bu eylem:

- Terminates the alias message flow setup by the queue manager name alias QMD_noım, that is, the named class of service QMD_noım.
- Bu iletilerdeki iletim üstbilgilerini QMD_noım 'den QMD' e dönüştürür.

Yöntem 3: İletim kuyruğu seçilmesi

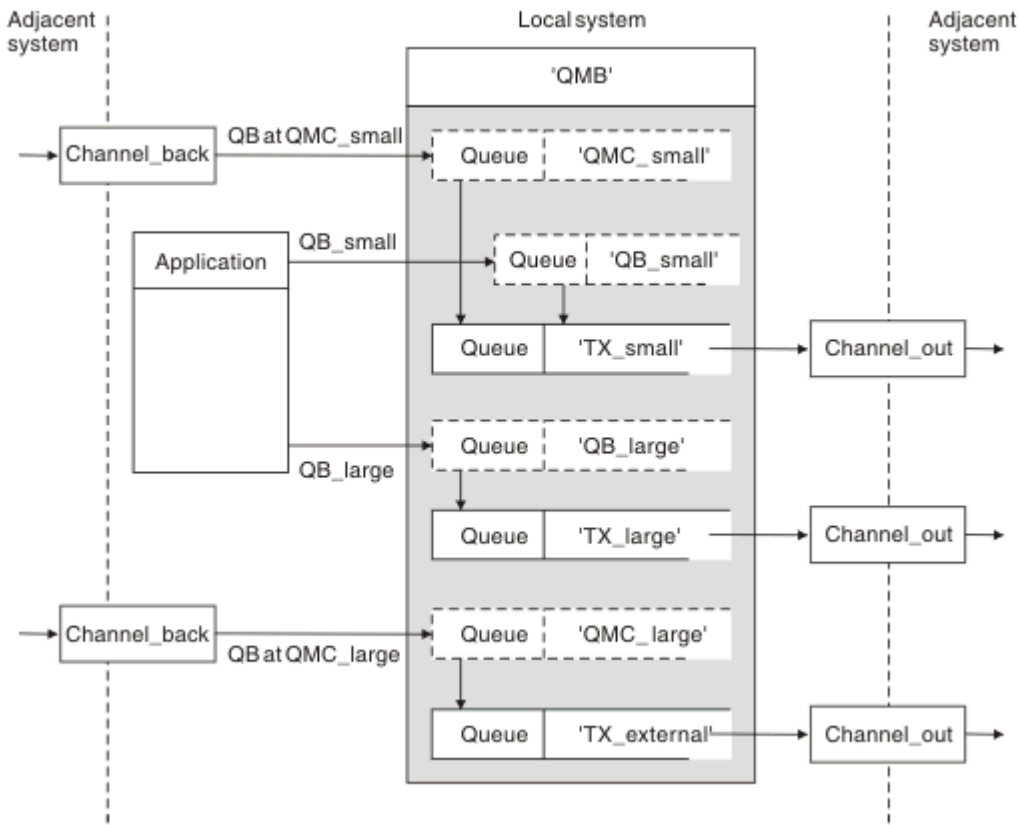
Üçüncü yöntem, hedef konumla aynı adı taşıyan bir kuyruk yöneticisi diğer adı nesnesine sahip olmak (QMD_PRIORITY). Belirli bir iletim kuyruğunu (QMD_fast) ve bu nedenle başka bir kanalı seçmek için kuyruk yöneticisi diğer adı tanımlamasını kullanın. Bu iletilerdeki iletim üstbilgileri değişmeden kalır.

İleti akışlarını ayırma

Aynı kuyruk yöneticisine ileti göndermek için ayrı ileti akışları yaratmak için bir kuyruk yöneticisi diğer adı kullanabilirsiniz.

Dağıtık kuyruklama ortamında, aynı kuyruk yöneticisine yönelik iletileri farklı ileti akışlarına ayırma gereksinimi, bir dizi nedenden kaynaklanabilir. Örneğin:

- Büyük, orta ve küçük iletiler için ayrı bir akış sağlamanız gerekebilir. Bu gerek kümeleme ortamında da geçerlidir ve bu durumda çakışan kümeler yaratabilirsiniz. Örneğin, yapabildiğiniz birçok neden vardır:
 - Farklı organizasyonların kendi yönetimlerine sahip olmasını sağlamak.
 - Bağımsız uygulamaların ayrı olarak denetlenmesine izin vermek için.
 - Bir hizmet sınıfı oluşturmak için. Örneğin, ÖĞRENCILER adlı kümenin bir alt kümesi olan STAFF adlı bir kümenin olabilir. STAFF kümesinde tanımlanan bir kuyruğa ileti yerleştirdiğinizde, sınırlı bir kanal kullanılır. ÖĞRENCILER kümesi içinde yayınlanan bir kuyruğa ileti yerleştirdiğinizde, ya genel bir kanal ya da bir sınırlı kanal kullanılabilir.
 - Test ve üretim ortamları yaratmak için.
- Gelen iletileri, yerel olarak üretilen iletilerin yolundan farklı yollara yönlendirmek gerekebilir.
- Kuruluşunuz belirli saatlerde (örneğin, gecelik) iletilerin hareketini zamanlamak ve sonra zamanlanan zamana kadar ayrılmış kuyruklarda saklanmak zorunda kalabilirler.



Şekil 6. İleti akışlarını ayırma

Şekil 6 sayfa 35 içinde gösterilen örnekte, gelen iki akış, 'QMC_küçük' ve 'QMC_large' kuyruk yöneticisi adlarının diğer ad kuyruk yöneticisi adlarına sahip olur. Bu akışları, yerel kuyruk yöneticisi için bu akışları yakalamak üzere bir kuyruk yöneticisi diğer adı tanımlamasıyla sağlıyoruz. İki uzak kuyrukta bir uygulamanız var ve bu ileti akışlarının ayrı tutulmasını istiyor musunuz? Aynı yeri ('QMC') belirten, ancak farklı iletim kuyruklarını belirten iki uzak kuyruk tanımlaması sağladınız. Bu tanımlama akımları ayrı tutar ve iletim üstbilgilerinde aynı hedef kuyruk yöneticisi adına sahip oldukları sürece, ek hiçbir şey gerekmez. Aşağıdakileri sağladınız:

- Gelen kanal tanımlamaları
- İki uzak kuyruk tanımlaması QB_küçük ve QB_large
- İki kuyruk yöneticisi diğer ad tanımlaması QMC_küçük ve QMC_large

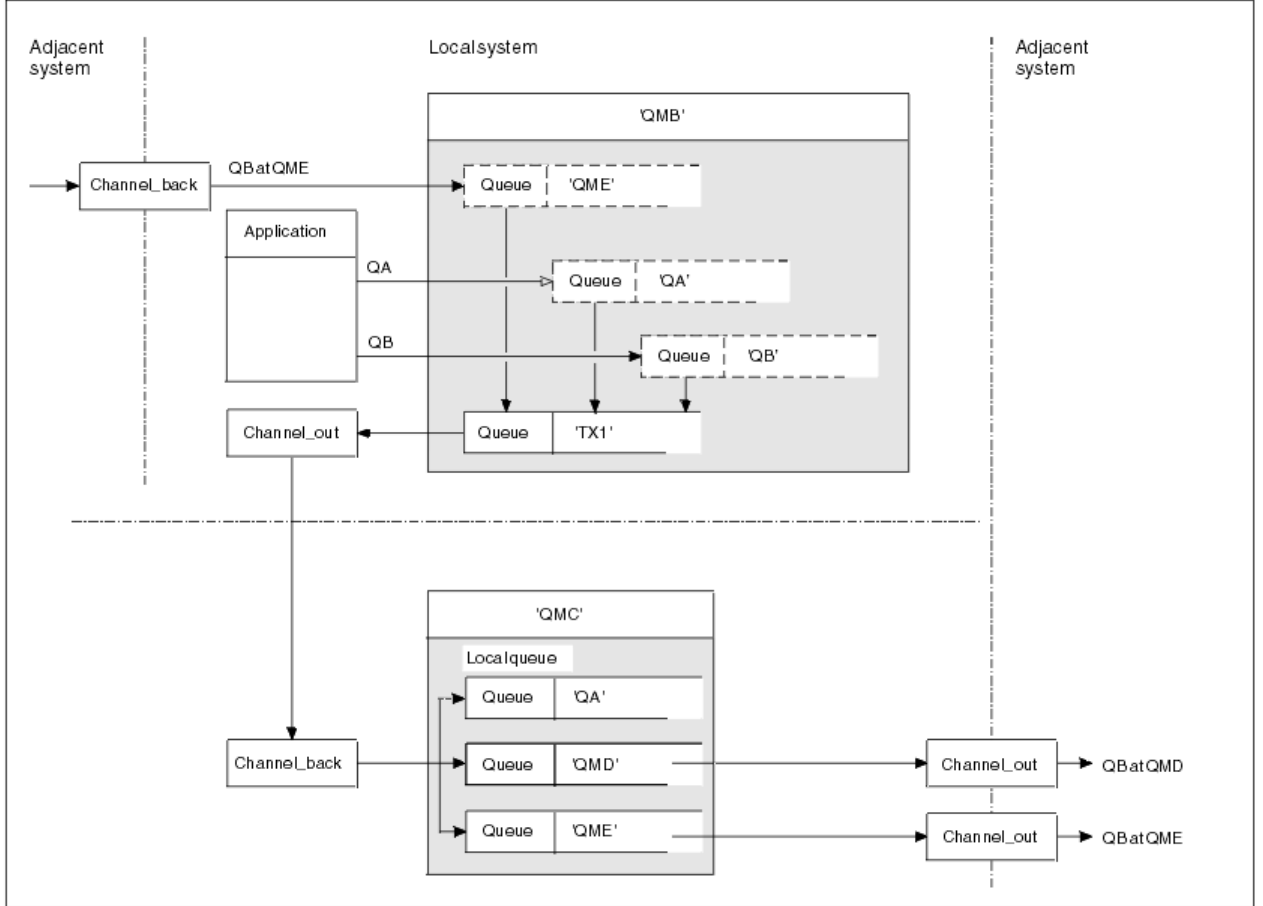
- Üç gönderen kanal tanımlaması
- Üç iletim kuyruğu: TX_küçük, TX_large ve TX_external

Bitişik sistemlerle koordinasyon

Ayrı bir ileti akışı yaratmak için bir kuyruk yöneticisi diğer adı kullandığınızda, bu etkinliği ileti kanalının uzak ucundaki sistem denetimcisiyle birlikte koordine etmek ve ilgili kuyruk yöneticisi diğer adının burada kullanılabilir olduğunu doğrulamak için bu etkinliği koordine etmek gerekir.

İletilerin farklı konumlara yoğunlaştırılması

Çeşitli konumlar için yazılmış mesajları tek bir kanala konsantre edebilirsiniz.



Şekil 7. İleti akışlarının bir kanalda birleştirilmesi

Şekil 7 sayfa 36 , bir kanalda çeşitli konumlara gönderilmek üzere gönderilen iletilerin yoğunlaştırılması için dağıtılmış kuyruklama tekniğini gösterir. İki olası kullanım şöyle olabilir:

- İleti trafiğinin bir ağ geçidi üzerinden yoğunlaştırılması
- Döğümler arasında geniş bant genişliği otoyollarını kullanma

Bu örnekte, farklı kaynaklardan (yerel ve bitişik) ve farklı hedef kuyruklarına ve kuyruk yöneticilerine sahip olan iletiler, 'TX1' iletim kuyruğu QMC ' ye aktarılmış. Kuyruk yöneticisi QMC, iletileri hedeflere göre teslim eder. Bir tanesi, kuyruk yöneticisi QMD 'ye yönelik onward iletimi için' QMD ' iletim kuyruğuna ayarlandı. Başka bir ileti, kuyruk yöneticisi QME 'ye gönderilmek üzere onward iletimi için' QME ' iletim kuyruğuna ayarlandı. Diğer iletiler 'QA' yerel kuyruğuna konabiliyor.

Aşağıdakileri sağlamanız gerekir:

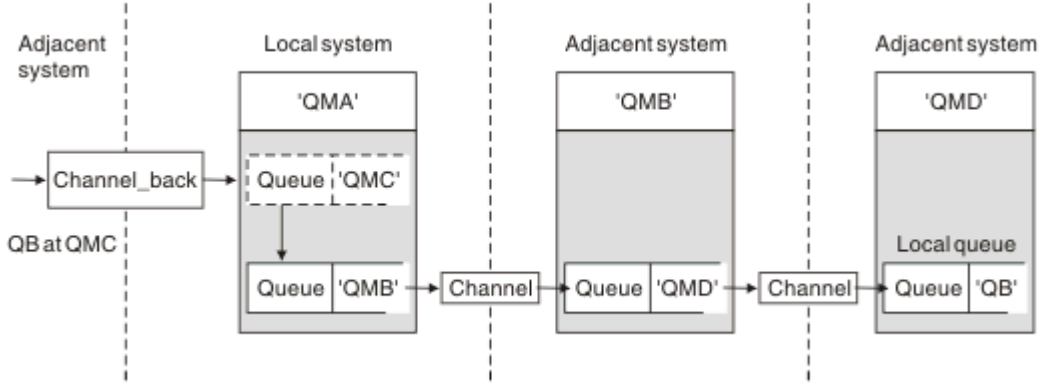
- Kanal tanımları
- İletim kuyruğu TX1
- Uzak kuyruk tanımlamaları:
 - QA ile 'QA at QMC- TX1'
 - QB with 'QB at QMD- TX1'
- Kuyruk yöneticisi diğer adı tanımlaması:
 - QME with 'QME- TX1'

QMC ' yi yapılandırılan tamamlayıcı yönetici aşağıdakileri sağlamalıdır:

- Aynı kanal adı ile kanal tanımlaması alınıyor
- İletim kuyruğu QMD, ilişkili gönderme kanalı tanımlaması ile
- İlişkili gönderme kanalı tanımlamasıyla QME iletim kuyruğu
- Yerel kuyruk nesnesi QA.

İleti akışlarının başka bir hedefe yönlendirmesi

Kuyruk yöneticisi diğer adlarını ve iletim kuyruklarını kullanarak belirli iletilerin hedefini yeniden tanımlayabilirsiniz.



Şekil 8. İleti akımları başka bir hedefe yönlendiriyor

Şekil 8 sayfa 37 , belirli iletilerin hedefini nasıl yeniden tanımlayabileceğiniz gösterilir. QMA 'ya gelen iletiler,' QB 'deki QB' ye gönderilmektedir. Genellikle QMA 'ya varırlar ve QMC denilen bir iletim kuyruğuna yerleştirilirler, ki bu da bir kanaldan QMC' ye kadar olan bir parçadır. QMA, iletileri QMD 'ye yönlendirmeli, ancak QMD' ye yalnızca QMB ' ye ulaşabilmelidir. Bu yöntem, bir hizmeti bir konumdan başka bir konuma taşımamız ve abonelerin yeni adrese ayarlanıncaya kadar geçici olarak ileti göndermeye devam etmesini sağlamak gerektiğinde kullanışlıdır.

Belirli bir kuyruk yöneticisine yönelik gelen iletileri farklı bir kuyruk yöneticisine yönlendiren yöntem:

- Hedef kuyruk yöneticisini başka bir kuyruk yöneticisine değiştirmek ve bitişik sisteme bir iletim kuyruğu seçmek için kuyruk yöneticisi diğer adı
- Bitişik kuyruk yöneticisine hizmet vermek için bir iletim kuyruğu
- Hedef kuyruk yöneticisine yönlendiren onward için bitişik kuyruk yöneticisinde bir iletim kuyruğu

Aşağıdakileri sağlamanız gerekir:

- Geri dönme_geri tanımlaması
- QMD 'de QMD ile QMB arasında QMC' nin kuyruk yöneticisi diğer ad nesnesi tanımlaması
- Channel_out tanımlaması
- İlişkili iletim kuyruğu QMB

QMB 'yi yapılandıran tamamlayıcı yönetici aşağıdakileri sağlamalıdır:

- Karşılık gelen channel_back tanımlaması
- İletim kuyruğu, QMD
- QMD 'ye ilişkin kanal tanımlaması

Kümeleme ortamında diğer adları kullanabilirsiniz. Bilgi için bkz. [“Kuyruk yöneticisi diğer adları ve kümeleri” sayfa 255.](#)

Dağıtım Listesine İleti Gönderilmesi

Bir uygulamanın birkaç noktaya ileti göndermesi için tek bir MQPUT çağrısını kullanabilirsiniz.

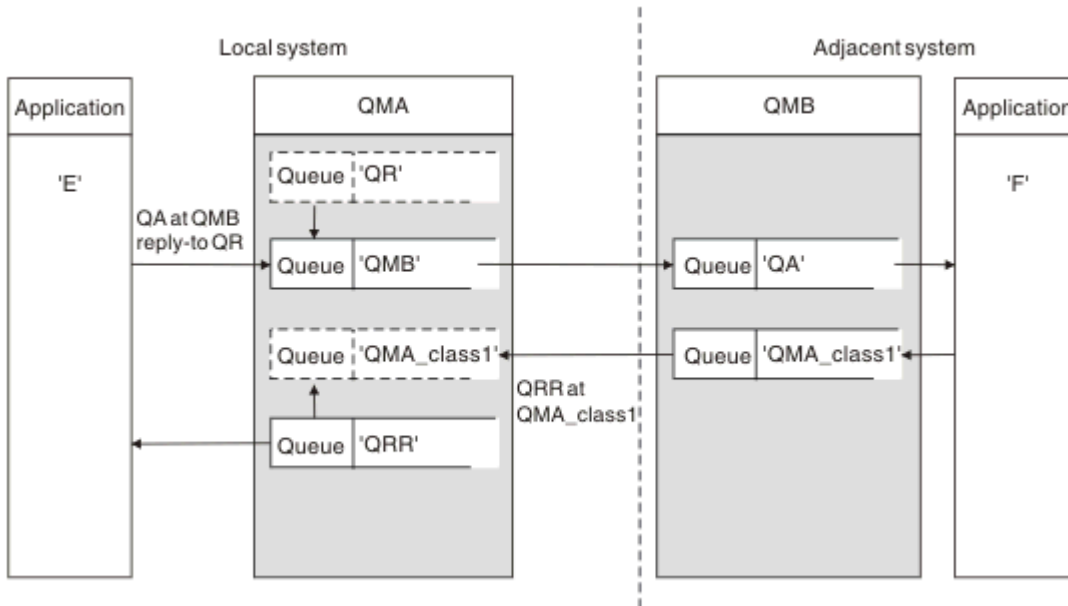
In WebSphere MQ on all platforms except z/OS, an application can send a message to several destinations with a single MQPUT call. Bunu hem dağıtım ortamında hem de kümeleme ortamında yapabilirsiniz. Hedefleri, [Dağıtım listeleri](#) içinde açıklandığı şekilde, dağıtım listesindeki bir hedef listesine tanımlamanız gerekir.

Tüm kuyruk yöneticileri dağıtım listelerini desteklemez. MCA bir iş ortağıyla bir bağlantı kurduğunda, iş ortağının dağıtım listelerini destekleyip desteklemediğini belirler ve iletim kuyruğunda bir işareti buna göre ayarlar. Bir uygulama, bir dağıtım listesi için yazılmış bir iletiyi göndermeye çalışırsa, ancak iş ortağı dağıtım listelerini desteklemiyorsa, MCA 'nın iletiyi göndermesi ve her hedef için bir kez iletim kuyruğuna yerleştirilmesi gerekir.

Alıcı MCA, dağıtım listesine gönderilen iletilerin, istenen tüm hedeflerde güvenli bir şekilde alınabilmesini sağlar. Herhangi bir hedef arızalanırsa, MCA hangilerinin başarısız olduğunu belirler. Daha sonra, kendileri için kural dışı durum raporları oluşturabilir ve iletileri yeniden bunlara göndermeyi deneyebilirler.

Yanıtın gönderileceği kuyruk

Bir yanıtla kuyruğu kullanarak, tam bir uzak kuyruk işleme döngüsü yaratabilirsiniz.



Şekil 9. Yanıtla-PUT çağrısı sırasında kuyruk adı yerine koyma değeri

Şekil 9 sayfa 38 içinde bir yanıt kuyruğu kullanılarak tam bir uzak kuyruk işleme döngüsü gösterilmektedir. Bu döngü hem dağıtım ortamına alma hem de kümeleme ortamı için geçerlidir. Ayrıntılar, [Çizelge 7 sayfa 45](#) içinde gösterildiği gibidir.

Uygulama QA 'da QA' yı açar ve bu kuyruğa ileti yerleştirir. İletiler, kuyruk yöneticisi adı belirtilmeden QR 'nin kuyruk adına yanıt verilir. Kuyruk yöneticisi QMA yanıt kuyruğu nesnesini bulur ve QR 'nin diğer

adını ve QMA_class1kuyruk yöneticisi adını kullanarak QR yanıt kuyruğu nesnesini bulur. Bu adlar, iletilerin yanıtlanma alanlarına yerleştirilir.

QMB 'deki uygulamalardan gelen yanıt iletileri, QMA_class1adresindeki QRR' ye adreslenir. Kuyruk yöneticisi diğer ad tanımı QMA_class1 , kuyruk yöneticisi tarafından iletilerin kendisine ve QRR kuyruğuna veri akışını kuyruğa almak için kullanılır.

Bu senaryoda, uygulamaların yanıt iletileri için bir hizmet sınıfı seçmesi için uygulama verme biçiminiz gösterilmektedir. The class is implemented by the transmission queue QMA_class1 at QMB, together with the queue manager alias definition, QMA_class1 at QMA. Bu şekilde, bir uygulamanın yanıtlanma kuyruğunu, akışlar, uygulama içermeden ayrılmadan önce ayrılabilir. Uygulama her zaman bu hizmet sınıfı için QR ' yi seçer. Hizmet sınıfını, yanıtlanma kuyruğu tanımlaması QR ile değiştirme olanağınız vardır.

Şunları yaratmanız gerekir:

- Yanıtlanma kuyruğu tanımlaması QR
- İletim kuyruğu nesnesi QMB
- Channel_out tanımlaması
- Geri dönme_geri tanımlaması
- Kuyruk yöneticisi diğer adı tanımlaması QMA_class1
- Yerel kuyruk nesnesi (QRR) yoksa

Bitişik sistemdeki tamamlayıcı sistem yöneticisi aşağıdaki özellikleri yaratmalıdır:

- Kanal tanımlaması alınıyor
- İletim kuyruğu nesnesi QMA_class1
- İlişkili gönderme kanalı
- Yerel kuyruk nesnesi QA.

Uygulama programlarınız şunları kullanır:

- Yanıtlanma çağrılarında QR kuyruğuna yanıt verin
- Alma çağrılarında QRR kuyruk adı

Bu şekilde, uygulamayı zorunlu kılmadan, hizmet sınıfını gerektiği gibi değiştirebilirsiniz. You change the reply-to alias 'QR', together with the transmission queue 'QMA_class1' and queue manager alias 'QMA_class1'.

İleti kuyruğa konduğunda yanıt adı başka bir nesne bulunamazsa, boş yanıt kuyruk yöneticisi adı alanına yerel kuyruk yöneticisi adı eklenir. Yanıtın gönderileceği kuyruk adı değişmeden kalır.

Ad çözme kısıtlaması

Özgün ileti yerleştirildiğinde 'QMA' yanıt kuyruğu için ad çözümlemesi gerçekleştirildiği için, 'QMB' konumunda başka ad çözme işlemine izin verilmez. İleti, yanıtlanma uygulaması tarafından yanıtlanacak yanıtın fiziksel adıyla birlikte konmaya devam eder.

Uygulamalar, yanıtlanma kuyruğu için kullandıkları adın, geri dönüş iletilerinin bulunacağı gerçek kuyruğun adından farklı olduğunun bilincinde olmalıdır.

Örneğin, yanıt kuyruğu diğer adı 'C1_alias' ve 'C2_alias' olan uygulamaların kullanımı için iki hizmet sınıfı sağlandığında, uygulamalar bu adları ileti koyma çağrılarında yanıt kuyruğunda adları olarak kullanır. Ancak, uygulamalar gerçekte 'C1_alias' ve 'C2_alias' için 'C2' için 'C1' kuyruklarında ileti görüntülenmesini bekliyor.

Ancak, bir uygulama, yanıt iletilerini almak için kullanması gereken gerçek kuyruğun adını denetlemek üzere, yanıt diğer ad kuyruğunda bir sorgu çağrısı yapabilir.

İlgili kavramlar

“Kuyruk yöneticisi ve yanıtın diğer adlara nasıl yaratılacağı” sayfa 29

Bu konuda, uzak kuyruk tanımlaması yaratabileceğiniz üç yol açıklanmaktadır.

“Yanıtlama kuyruğu diğer adı örneği” sayfa 40

Bu örnek, döndürülen iletiler için farklı bir rota (iletim kuyruğu) seçmek için bir yanıtlama diğer adı kullanılmasını gösterir. Bu olanağın kullanılması, başvurularla işbirliği içinde değiştirilmek üzere yanıt kuyruğu adının değiştirilmesini gerektirir.

“Örnek nasıl çalışır” sayfa 42

Örneğin bir açıklaması ve kuyruk yöneticisinin yanıtlama kuyruğu diğer adını nasıl kullandığını gösteren açıklamalar.

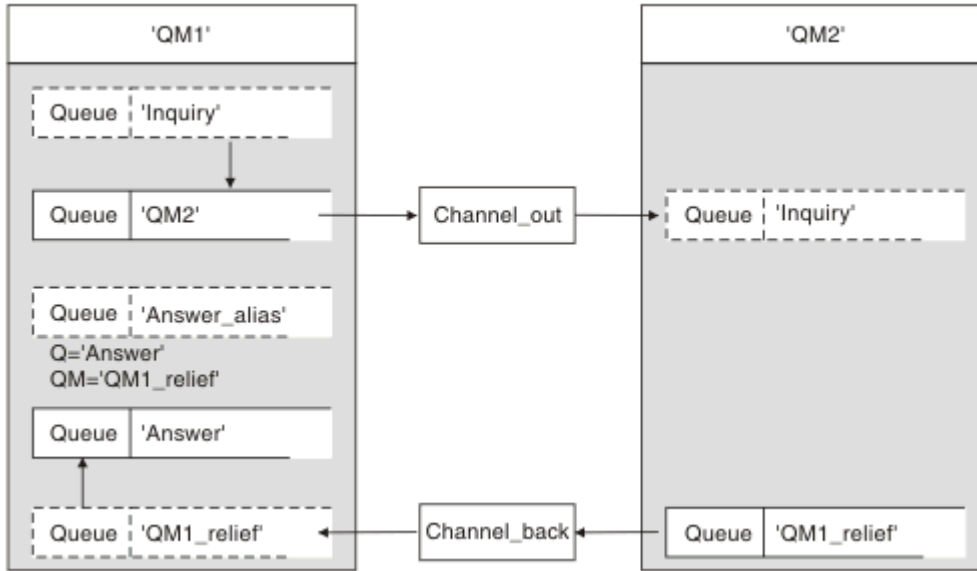
“Yanıtlama-kuyruk diğer adı alt-yolu” sayfa 42

Bir uygulamadan gelen bir uygulama aracılığıyla, aynı uygulama aracılığıyla bir iletiyi uzak bir kuyruğa koyarak, yanıt iletilerini diğer ad yanıtından kuyruğa alma işlemi yoluyla birlikte gönderiniz.

Yanıtlama kuyruğu diğer adı örneği

Bu örnek, döndürülen iletiler için farklı bir rota (iletim kuyruğu) seçmek için bir yanıtlama diğer adı kullanılmasını gösterir. Bu olanağın kullanılması, başvurularla işbirliği içinde değiştirilmek üzere yanıt kuyruğu adının değiştirilmesini gerektirir.

Şekil 10 sayfa 40’inde gösterildiği gibi, dönüş rotasının iletim kuyruğu, kanal ve kuyruk yöneticisi diğer adı da içinde olmak üzere yanıt iletileri için kullanılabilir olması gerekir.



Şekil 10. Yanıtlama kuyruğu diğer adı örneği

Bu örnek, 'QM2' sunucusundaki sunucu uygulamalarına ileti gönderen 'QM1' konumundaki istekte bulunana ilişkin uygulamalar içindir. Sunucudaki iletiler, 'QM1_relief' iletim kuyruğunu kullanarak alternatif bir kanaldan geri döndürüleceklerdir (varsayılan dönüş kanalı, 'QM1' iletim kuyruğunda hizmet verir).

Yanıtlama kuyruğu diğer adı, 'Answer_alias' adlı uzak kuyruk tanımlamasının belirli bir kullanışıdır. QM1 konumundaki uygulamalar, 'Sorgu' kuyruğuna koydukları tüm iletilerin yanıtlama alanına bu adı ('Answer_alias') içerir.

Reply-to queue definition 'Answer_alias' is defined as 'Answer at QM1_relief'. QM1 konumundaki uygulamalar, yanıtlarının 'Yanıt' adlı yerel kuyrukta görünmesini bekliyor.

Server applications at QM2 use the reply-to field of received messages to obtain the queue and queue manager names for the reply messages to the requester at QM1.

Bu örnekte kullanılan tanımlamalar: QM1

QM1 adresindeki WebSphere MQ sistem yöneticisi, diğer nesnelere birlikte yanıtlama kuyruğunun 'Yanıt' yaratıldığını doğrulamalıdır. '*' ile işaretlenen kuyruk yöneticisi diğer adının adı, yanıt kuyruğu diğer adı tanımlamasındaki kuyruk yöneticisi adıyla aynı olmalıdır; ayrıca '*' ile de işaretlenir.

Nesne	Tanımlama	
Yerel iletim kuyruğu	QM2	
Uzak kuyruk tanımlaması	Nesne adı	Sorgu
	Uzak kuyruk yöneticisi adı	QM2
	Uzak kuyruk adı	Sorgu
	İletim kuyruğu adı	QM2 (VARSAYILAN)
Kuyruk yöneticisi diğer adı	Nesne adı	QM1_relief *
	Kuyruk yöneticisi adı	QM1
	Kuyruk adı	(boş)
Yanıtlama kuyruğu diğer adı	Nesne adı	Answer_alias
	Uzak kuyruk yöneticisi adı	QM1_relief *
	Uzak kuyruk adı	Yanıt

Tanımları şu adresten al: QM1

Uygulamalar yanıtı, yanıt kuyruğu diğer adı ile doldurur ve kuyruk yöneticisi adı alanını boş bırakır.

Alan	İçerik
Kuyruk adı	Sorgu
Kuyruk yöneticisi adı	(boş)
Yanıtın gönderileceği kuyruk adı	Answer_alias
Yanıtın gönderileceği kuyruk yöneticisi	(boş)

Bu örnekte kullanılan tanımlamalar: QM2

QM2 adresindeki WebSphere MQ sistem yöneticisi, gelen iletiler için yerel kuyruğun var olduğunu ve yanıt iletileri için doğru olarak adlandırılmış iletim kuyruğunun kullanılabilir olduğunu doğrulamalıdır.

Nesne	Tanımlama
Yerel kuyruk	Sorgu
İletim kuyruğu	QM1_relief

Tanımları şu adresten al: QM2

QM2 adresindeki uygulamalar, yanıt kuyruğu adını ve kuyruk yöneticisi adını özgün iletiden alır ve yanıt iletisini yanıtlama kuyruğuna koyarken bunları kullanır.

Alan	İçerik
Kuyruk adı	Yanıt
Kuyruk yöneticisi adı	QM1_relief

Örnek nasıl çalışır

Örneğin bir açıklaması ve kuyruk yöneticisinin yanıtlama kuyruğu diğer adını nasıl kullandığını gösteren açıklamalar.

Bu örnekte, QM1 konumundaki requester uygulamaları, put çağrısının ilgili alanında yanıt kuyruğu olarak her zaman 'Ansfere_alias' seçeneğini kullanır. Her zaman mesajlarını 'Cevap' adlı kuyruktan alır.

The reply-to queue alias definitions are available for use by the QM1 system administrator to change the name of the reply-to queue 'Answer', and of the return route 'QM1_relief'.

Changing the queue name 'Answer' is normally not useful because the QM1 applications are expecting their answers in this queue. Ancak, QM1 sistem yöneticisi gerektiği şekilde dönüş rotasını (hizmet sınıfı) değiştirebilir.

Kuyruk yöneticisinin yanıtlama kuyruğu diğer adını nasıl kullandığı

Queue manager QM1 retrieves the definitions from the reply-to queue alias when the reply-to queue name, included in the put call by the application, is the same as the reply-to queue alias, and the queue manager part is blank.

Kuyruk yöneticisi, put çağrısındaki yanıtlama kuyruğu adının yerine, tanımdan kuyruk adı ile değiştirilir. Put çağrısında boş kuyruk yöneticisi adının yerine, tanımdan kuyruk yöneticisi adı konur.

Bu adlar, ileti tanımlayıcısında iletiyle birlikte taşınır.

Çizelge 4. Yanıtlama kuyruğu diğer adı		
Alan adı	Çağır	İletim üstbilgisi
Yanıtın gönderileceği kuyruk adı	Answer_alias	Yanıt
Yanıtın gönderileceği kuyruk yöneticisi adı	(boş)	QM1_relief

Yanıtlama-kuyruk diğer adı alt-yolu

Bir uygulamadan gelen bir uygulama aracılığıyla, aynı uygulama aracılığıyla bir iletiyi uzak bir kuyruğa koyarak, yanıt iletilisini diğer ad yanıtından kuyruğa alma işlemi yoluyla birlikte gönderiniz.

Bu örneği tamamlamak için, sürece bir bakalım.

1. Uygulama 'Sorgu' adında bir kuyruk açar ve iletiyi bu kuyruğa koyar. Uygulama, ileti tanımlayıcısının yanıtlama alanlarını aşağıdaki gibi ayarlar:

Yanıtın gönderileceği kuyruk adı	Answer_alias
Yanıtın gönderileceği kuyruk yöneticisi adı	(boş)

2. Queue manager 'QM1' responds to the blank queue manager name by checking for a remote queue definition with the name 'Answer_alias'. Hiçbiri bulunamazsa, kuyruk yöneticisi ileti tanımlayıcısının yanıt kuyruğu yöneticisi alanına kendi adını ('QM1') yerleştirir.
3. Kuyruk yöneticisi, 'Ansfere_alias' adlı bir uzak kuyruk tanımlaması bulursa, bu, tanımdan kuyruk adı ve kuyruk yöneticisi adlarını alır (kuyruk adı= 'Yanıt' ve kuyruk yöneticisi adı= 'QM1_relief'). Daha sonra, bunları ileti tanımlayıcısının yanıtlama alanlarına yerleştirir.
4. 'QM1' kuyruk yöneticisi, amaçlanan hedef kuyruğun 'QM2' kuyruk yöneticisinde olduğunu ve iletinin 'QM2' iletim kuyruğuna yerleştirdiğini saptamak için uzak kuyruk tanımlamasını 'Sorgu' kullanır. 'QM2', 'QM2' kuyruk yöneticisinde kuyruklar için gönderilen iletiler için varsayılan iletim kuyruğu adıdır.
5. 'QM1' kuyruk yöneticisi iletiyi iletim kuyruğuna yerleştirirken, iletiye bir iletim üstbilgisi ekler. Bu üstbilgi hedef kuyruğunun adını, 'Sorgu' adını ve hedef kuyruk yöneticisini ('QM2') içerir.
6. İleti, 'QM2' kuyruk yöneticisine ulaşır ve 'Sorgu' yerel kuyruğuna yerleştirilir.
7. Bir uygulama bu kuyruktan iletiyi alır ve iletiyi işler. Uygulama bir yanıt iletilisi hazırlar ve bu yanıt iletilisini yanıt kuyruğu adına özgün iletinin ileti tanımlayıcısından koyar:

Yanıtın gönderileceği kuyruk adı

Yanıt

Yanıtın gönderileceği kuyruk yöneticisi adı

QM1_relief

8. 'QM2' kuyruk yöneticisi put komutunu gerçekleştiriyor. Kuyruk yöneticisi adının ('QM1_relief') uzak bir kuyruk yöneticisi olduğunu bulmak, iletiyi, aynı adı taşıyan ('QM1_relief') ileti kuyruğuna yerleştirir. İletiyeye, hedef kuyruk adı 'Yanıt' ve hedef kuyruk yöneticisi 'QM1_relief' adını içeren bir iletim üstbilgisi verilir.
9. İleti, 'QM1' kuyruk yöneticisine aktarılır. Kuyruk yöneticisi, 'QM1_relief' kuyruk yöneticisi adının bir diğer ad olduğunu algılar, 'QM1_relief' fiziksel kuyruk yöneticisi adının 'QM1' diğer ad tanımlamasından çıkarılır.
10. Kuyruk yöneticisi 'QM1' sonra ileti, 'Yanıt' iletim üstbilgisinde bulunan kuyruk adına yerleştirir.
11. Uygulama, yanıt iletisini 'Yanıt' kuyruğundan çıkarır.

Ağ oluşturma konuları

Dağıtılmış-kuyruklama ortamında, ileti hedefleri yalnızca bir kuyruk adı ve kuyruk yöneticisi adı ile adreslendiğinden, bazı kurallar geçerlidir.

1. Kuyruk yöneticisi adı verildiğinde ve yerel kuyruk yöneticisi adından farklı bir ad olmalıdır:
 - Aynı adda bir iletim kuyruğu kullanılabilir olmalıdır. Bu iletim kuyruğu, iletileri başka bir kuyruk yöneticisine taşıyan bir ileti kanalının bir parçası olmalıdır ya da
 - Kuyruk yöneticisi diğer adı aynı olan kuyruk yöneticisi adını ya da başka bir kuyruk yöneticisi adını ve isteğe bağlı iletim kuyruğunu çözmek için bir kuyruk yöneticisi diğer adı olmalıdır.
 - İletim kuyruğu adı çözülemezse ve varsayılan iletim kuyruğu tanımlandıysa, varsayılan iletim kuyruğu kullanılır.
2. Yalnızca kuyruk adı belirtildiğinde, yerel kuyruk yöneticisinde aynı adı taşıyan bir kuyruğun bulunduğu bir kuyruk kullanılabilir. Bu kuyruk, bir iletim kuyruğunu bitişik bir kuyruk yöneticisine, kuyruk yöneticisi adına ve isteğe bağlı bir iletim kuyruğuna kadar çözen uzak bir kuyruk tanımlaması olabilir.

Bu işin kümeleme ortamında nasıl çalıştığını görmek için, ürün belgelerinin [Kümelerin çalışma şekli](#) bölümünde uygun konulara bakın.

İletileri bir kuyruk yöneticisinden dağıtılmış kuyruklama ortamında başka bir kuyruk yöneticisinden başka bir ileti kanalına taşımanın senaryosunu göz önünde bulundurun.

Taşınmakta olan iletiler ağdaki başka bir kuyruk yöneticisinden kaynaklandığı için, hedef olarak bilinmeyen bir kuyruk yöneticisi adına sahip bazı iletiler gelebilir. Bu sorun, bir kuyruk yöneticisi adı değiştirildiğinde ya da sistemden kaldırıldığında ortaya çıkabilir.

Kanal programı, bu iletiler için bir iletim kuyruğu bulamadığında ve iletileri, teslim edilemeyen ileti (dead-letter) kuyruğuna yerleştirdiğinde bu durumu tanır. Bu iletileri aramak ve doğru hedefe iletilmeleri için düzenlemek sizin sorumluluğunuz. Diğer bir yöntem olarak, bunları düzenleyen edinilebileceği orijinalator 'a geri gönderin.

Özgün iletide rapor iletileri istendiyse, kural dışı durum raporları bu koşullarda oluşturulur.

Ad çözme kuralı

Hedef kuyruğun kimliğini değiştiren (yani, fiziksel adın mantıksal olarak değişen mantıksal ad), yalnızca bir kez ve yalnızca kaynak kuyruk yöneticisinde oluşan bir ad çözümlenmesi.

Çeşitli diğer ad olasılıklarının sonraki kullanımı, yalnızca ileti akışları ayrılırken ve birleştirildiğinde kullanılmalıdır.

İade yönlendirmesi

İletiler, kuyruğun ve kuyruk yöneticisinin adı biçiminde bir dönüş adresi içerebilir. Bu dönüş adresi formu, hem dağıtım kuyruğuna alma ortamında hem de kümeleme ortamında kullanılabilir.

Bu adres olağan olarak, iletiyi yaratan uygulama tarafından belirtilir. Kullanıcı çıkış uygulamaları da içinde olmak üzere, iletiyi işleyen herhangi bir uygulama tarafından değiştirilebilir.

Bu adresin kaynağından bağımsız olarak, iletiyi işleyen herhangi bir uygulama yanıt, durum ya da rapor iletilerini kaynak uygulamaya geri döndürmek için bu adresi kullanmayı seçebilir.

Bu yanıt iletilerinin yöneltme şekli, özgün iletinin yöneltmesinden farklı değildir. Diğer kuyruk yöneticilerine yarattığınız ileti akışlarının karşılık gelen dönüş akışlarına gereksinim duyduğunu bilmeniz gerekir.

Fiziksel ad çakışmaları

Hedef yanıt kuyruğu adı, özgün kuyruk yöneticisinde fiziksel bir kuyruk adına çözülmüş. Yanıt veren kuyruk yöneticisinde yeniden çözümlenmemelidir.

Bu, yalnızca fiziksel ve mantıksal kuyruk adlarında ağ çapında bir sözleşme tarafından önlenebilecek ad çakışması sorunları için olası bir olasıdır.

Kuyruk adı çevirilerini yönetme

Bir kuyruk yöneticisi diğer adı tanımlaması ya da uzak kuyruk tanımlaması yarattığınızda, bu adı taşıyan her ileti için ad çözümlemesi gerçekleştirilir. Bu durum yönetilmeli.

Bu açıklama, bitişik sistemlere ileti kanalları olan tek bir sistem ile ilgilenen uygulama tasarımcıları ve kanal planlayıcıları için sağlanmıştır. Kanal planlama ve kontrolünün yerel bir görünümünü ele alır.

Bir kuyruk yöneticisi diğer adı tanımlaması ya da uzak bir kuyruk tanımlaması yarattığınızda, iletinin kaynağı ne olursa olsun, o adı taşıyan her ileti için ad çözümlemesi gerçekleştirilir. Bir kuyruk yöneticisi ağında çok sayıda kuyruk içerebilecek olan bu durumu denetlemek için, aşağıdaki çizelgelerin çizelgelerini alıkoyabilirsiniz:

- Çözümlenen kuyruk adlarına, çözümlenen kuyruk yöneticisi adlarına ve çözülmüş iletim kuyruğu adlarına göre, kaynak kuyrukların ve kaynak kuyruk yöneticilerinin adları
- Aşağıdakine göre kaynak kuyrukların adları:
 - Çözülmüş hedef kuyruk adları
 - Çözülmüş hedef kuyruk yöneticisi adları
 - İletim kuyrukları
 - İleti kanalı adları
 - Bitişik sistem adları
 - Yanıtlama kuyruğu adları

Not: Bu bağlamda *kaynak* teriminin kullanılması, kuyruk adına ya da uygulama tarafından sağlanan kuyruk yöneticisi adına ya da ileti koymak için bir kuyruk açılırken kanal programı anlamına gelir.

Bu tabloların her birine bir örnek olarak [Çizelge 5 sayfa 44](#), [Çizelge 6 sayfa 45](#)ve [Çizelge 7 sayfa 45](#) içinde gösterilir.

Bu tablolardaki adlar, bu bölümdeki örneklerden türetilmiştir ve bu tablo, bir düğümde kuyruk adı çözümünün pratik bir örneği olarak tasarlanmamaktadır.

Kuyruk açıldığında kaynak kuyruk belirtildi	Kuyruk açıldığında kaynak kuyruk yöneticisi belirtildi	Çözülmüş kuyruk adı	Çözülmüş kuyruk yöneticisi adı	Çözülmüş iletim kuyruğu adı	Çözünürlük tipi
SA_norm	-	SA_norm	QMB	QMB	Uzak kuyruk

Çizelge 5. Kuyruk yöneticisi QMA ' da kuyruk adı çözümlemesi (devamı var)

Kuyruk açıldığında kaynak kuyruk belirtildi	Kuyruk açıldığında kaynak kuyruk yöneticisi belirtildi	Çözülmüş kuyruk adı	Çözülmüş kuyruk yöneticisi adı	Çözülmüş iletim kuyruğu adı	Çözünürlük tipi
(herhangi biri)	QMB	-	-	QMB	(yok)
SA_norm	-	SA_norm	QMB	TX1	Uzak kuyruk
QB	QMC	QB	QMD	QMB	Kuyruk yöneticisi diğer adı

Çizelge 6. Kuyruk yöneticisi QMB ' de kuyruk adı çözümlemesi

Kuyruk açıldığında kaynak kuyruk belirtildi	Kuyruk açıldığında kaynak kuyruk yöneticisi belirtildi	Çözülmüş kuyruk adı	Çözülmüş kuyruk yöneticisi adı	Çözülmüş iletim kuyruğu adı	Çözünürlük tipi
SA_norm	-	SA_norm	QMB	-	(yok)
SA_norm	QMB	SA_norm	QMB	-	(yok)
SA_norm	QMB_PRIORITY	SA_norm	QMB	-	Kuyruk yöneticisi diğer adı
(herhangi biri)	QMC	(herhangi biri)	QMC	QMC	(yok)
(herhangi biri)	QMD_norm	(herhangi biri)	QMD_norm	TX1	Kuyruk yöneticisi diğer adı
(herhangi biri)	QMD_PRIORITY	(herhangi biri)	QMD_PRIORITY	QMD_fast	Kuyruk yöneticisi diğer adı
(herhangi biri)	QMC_küçük	(herhangi biri)	QMC_küçük	TX_küçük	Kuyruk yöneticisi diğer adı
(herhangi biri)	QMC_large	(herhangi biri)	QMC_large	TX_Dış	Kuyruk yöneticisi diğer adı
QB_küçük	QMC	QB_küçük	QMC	TX_küçük	Uzak kuyruk
QB_büyük	QMC	QB_büyük	QMC	TX_büyük	Uzak kuyruk
(herhangi biri)	QME	(herhangi biri)	QME	TX1	Kuyruk yöneticisi diğer adı
QA	QMC	QA	QMC	TX1	Uzak kuyruk
QB	QMD	QB	QMD	TX1	Uzak kuyruk

Çizelge 7. Yanıtlama-kuyruk yöneticisi QMA ' da kuyruk adı çevirisi

Uygulama tasarımı		Yanıt-diğer ad tanımlaması	
Yerel QMGR	İletilere ilişkin kuyruk adı	Yanıt kuyruğu diğer adı	Yeniden tanımlansa:
QMA	QRR	QR	QRR- QMA_class1

Kanal ileti sırası numaralandırması

Kanal, iletilerin teslim edildiğinden emin olmak için sıra numaralarını kullanır, çoğaltılmadan teslim edilir ve iletim kuyruğundan alındıkları sırayla saklanır.

Sıra numarası, kanalın gönderilmesi sırasında oluşturulur ve kullanılmadan önce bir artırılır. Bu, yürürlükteki sıra numarasının gönderilen son iletinin numarası olduğu anlamına gelir. Bu bilgiler, DISPLAY CHSTATUS (bkz. MQSC başvurusu) kullanılarak görüntülenebilir. Sıra numarası ve LUWID adı verilen bir tanıttıcı, bir toplu işte aktarılan son ileti için kalıcı depoda saklanır. Bu değerler, kanal başlatma sırasında, bağlantının her iki ucunun hangi iletilerin başarıyla aktarıldığını kabul ettiğini doğrulamak için kullanılır.

İletilerin sıralı olarak alınması

Bir uygulama, aynı hedef kuyruğa bir ileti dizisi koyarsa, aşağıdaki koşullar karşılanırsa, bu iletiler bir MQGET işlemi sırasıyla **tek** uygulaması tarafından sırayla alınabilir:

- Tüm put istekleri aynı uygulamadan yapıldı.
- Put istekleri aynı iş biriminden ya da bir iş birimi dışında yapılan tüm istekler.
- İletilerin hepsi aynı önceliğe sahip.
- İletilerin hepsi aynı kalıcılığa sahip.
- Uzak kuyruğa alma için, yapılandırma, koyma isteğini, kuyruk yöneticisi aracılığıyla, iletişim yoluyla hedef kuyruk yöneticisine ve hedef kuyruğa kadar yalnızca tek bir yol olabilir.
- İletiler, bir ölüme ilişkin kuyruğa konmaz (örneğin, bir kuyruk geçici olarak dolduysa).
- İletiyi alan uygulama, örneğin belirli bir *MsgId* ya da *CorrelId* belirtilerek ya da ileti önceliklerini kullanarak alma sırasını kasıtlı olarak değiştirmez.
- İletileri hedef kuyruktan almak için yalnızca tek bir uygulama işlemi gerçekleştiriyor. Birden fazla uygulama varsa, bu uygulamaların her bir sıra içindeki tüm iletileri bir gönderme uygulaması tarafından elde etmek için tasarlanabilmeleri gerekir.

Not: Diğer görevlerden ve iş birimlerinden gelen iletiler, sıranın tek bir iş birimi içinden konduğu yerde bile diziyle birlikte spermlerle birlikte kullanılabilir.

Bu koşullar karşılanamazsa ve hedef kuyruklardaki iletilerin sırası önemliyse, bu durumda uygulama, iletilerin sırasını garanti etmek için iletinin bir parçası olarak kendi ileti sıra numarasını kullanacak şekilde kodlanabilir.

Hızlı, kalıcı olmayan iletilerin alınması sırası

Bir hızlı kanaldaki kalıcı olmayan iletiler, aynı kanaldaki kalıcı iletileri geçersiz kılabilir ve bu iletiler sırasız olarak gelebilir. Alma MCA, kalıcı olmayan iletileri hedef kuyruğa hemen koyar ve bunları görünür kılar. Kalıcı iletiler bir sonraki eşitleme noktasına kadar görünür kılınmaz.

Geri döngü sınaması

Geri döngü sınaması, z/OS dışı platformlarda, bir iletişim bağlantısını gerçekten başka bir makineye bağlanmadan sınamanızı sağlayan bir tekniktir.

İki kuyruk yöneticisi arasında ayrı makinelerde olduğu gibi bir bağlantı kurunuz, ancak aynı makineden başka bir işleme döngüleyerek bağlantıyı sınyadınız. Bu teknik, etkin bir ağ gerektirmeden, iletişim kodunuzu sinabileceğiniz anlamına gelir.

Bu şekilde, kullanmakta olduğunuz ürün ve protokollere bağlıdır.

Windows sistemlerinde, "geriçevrim" bağdaştırıcısını kullanabilirsiniz.

Ek bilgi için kullanmakta olduğunuz ürünlere ilişkin belgelere bakın.

Rota izleme ve etkinlik kaydı

Bir iletinin, iki şekilde bir dizi kuyruk yöneticisi aracılığıyla yönlendirmesini onaylayabilirsiniz.

You can use the WebSphere MQ display route application, available through the control command dspmqrt, or you can use activity recording. Bu konuların her ikisi de [İzleme başvurusu](#) içinde anlatılır.

Dağıtılmış kuyruk yönetimine giriş

Distributed queue Management (DQM), kuyruk yöneticileri arasında iletişimi tanımlamak ve denetlemek için kullanılır.

Dağıtılmış kuyruk yönetimi:


- Kuyruk yöneticileri arasında iletişim kanallarını tanımlamanızı ve denetlemenizi sağlar
- İletilerin iletim kuyruğu olarak bilinen bir *yemel kuyruktipinden* yerel bir sistemdeki iletişim bağlantılarına ve bir hedef kuyruk yöneticisinde yerel kuyruklara yönelik iletişim bağlantılarından ileti taşınması için size bir ileti kanalı hizmeti sağlar.
- Panoları, komutları ve programları kullanarak, kanalların işleyişi ve sorunların tanınması için gereken olanakları sağlar

Kanal tanımları, kanal adlarını iletim kuyruklarıyla, iletişim bağlantısı tanıtıcılarıyla ve kanal öznelikleri ile ilişkilendirir. Kanal tanımları farklı platformlarda farklı şekillerde uygulanmaktadır. İleti gönderme ve alma işlemi, iletişimi başlatmak ve denetlemek için kanal tanımlamalarını kullanan *ileti kanalı araçları* (MCA 'lar) adıyla bilinen programlarla denetlenir.

Sırayla MCA 'lar DQM' nin kendisi tarafından denetlenir. Yapı, platforma bağlıdır, ancak genellikle dinleyici ve tetikleyici izleyicileri içerir, işletmen komutları ve panolarıyla birlikte.

İleti kanalı , iletilerin bir kuyruk yöneticisinden diğerine taşınması için tek yönlü bir iletişim tüneldir. Bu nedenle bir ileti kanalının iki uç noktası vardır ve MCA ' ların bir çiftiyle gösterilir. Her uç noktada, ileti kanalının sonuna ilişkin bir tanımlama vardır. Örneğin, bir son bir göndereni, diğer ucunu da bir alıcı olarak tanımlardı.

Kanalları tanımlamaya ilişkin ayrıntılar için bkz:

-  [“UNIX, Linux, and Windows üzerindeki kanalların izlenmesi ve denetlenmesi” sayfa 72](#)

İleti kanalı planlama örnekleri için aşağıdaki başlara bakın:

-  [Dağıtılmış platformlar için ileti kanalı planlama örneği](#)

Kanal çıkışlarına ilişkin bilgi için [İleti alışverişi kanallarına ilişkin kanal çıkışı programları](#) başlıklı konuya bakın.

İlgili kavramlar

[“İleti gönderme ve alma” sayfa 48](#)

Aşağıdaki şekil, ileti iletilildiğinde varlıklar arasındaki ilişkileri ayrıntılandıran dağıtılmış kuyruk yönetimi modelini göstermektedir. Ayrıca, denetim akışının akışını da gösterir.

[“Kanal denetimi işlevi” sayfa 52](#)

Kanal denetim işlevi, kanalları tanımlamanız, izlemeniz ve denetlemeniz için gerekli olanakları sağlar.

[“Bir ileti teslim edilemezse ne olur?” sayfa 65](#)

Bir ileti teslim edilemediğinde, MCA bunu birkaç şekilde işleyebilir. Yeniden deneyebilir, gönderene geri dönüşebilir ya da onu ölü harf kuyruğuna yerleştirebilir.

[“Başlatma ve yapılandırma dosyaları” sayfa 69](#)

Kanal kullanıma hazırlama verilerinin işlenmesi, WebSphere MQ platformunuza bağlıdır.

[“İletiler için veri dönüştürme” sayfa 70](#)

WebSphere MQ iletileri, farklı kuyruk yöneticilerindeki kuyruklar arasında gönderildiklerinde veri dönüştürmesi gerektirebilir.

[“Kendi ileti kanalı araçlarınızı yazma” sayfa 70](#)

WebSphere MQ , kendi Message Channel Agent (MCA) programlarınızı yazmanızı ya da bir bağımsız yazılım üreticisinden birini kurmanızı sağlar.

“Dağıtılmış kuyruk yönetimi için dikkate alınması gereken diğer konular” sayfa 71

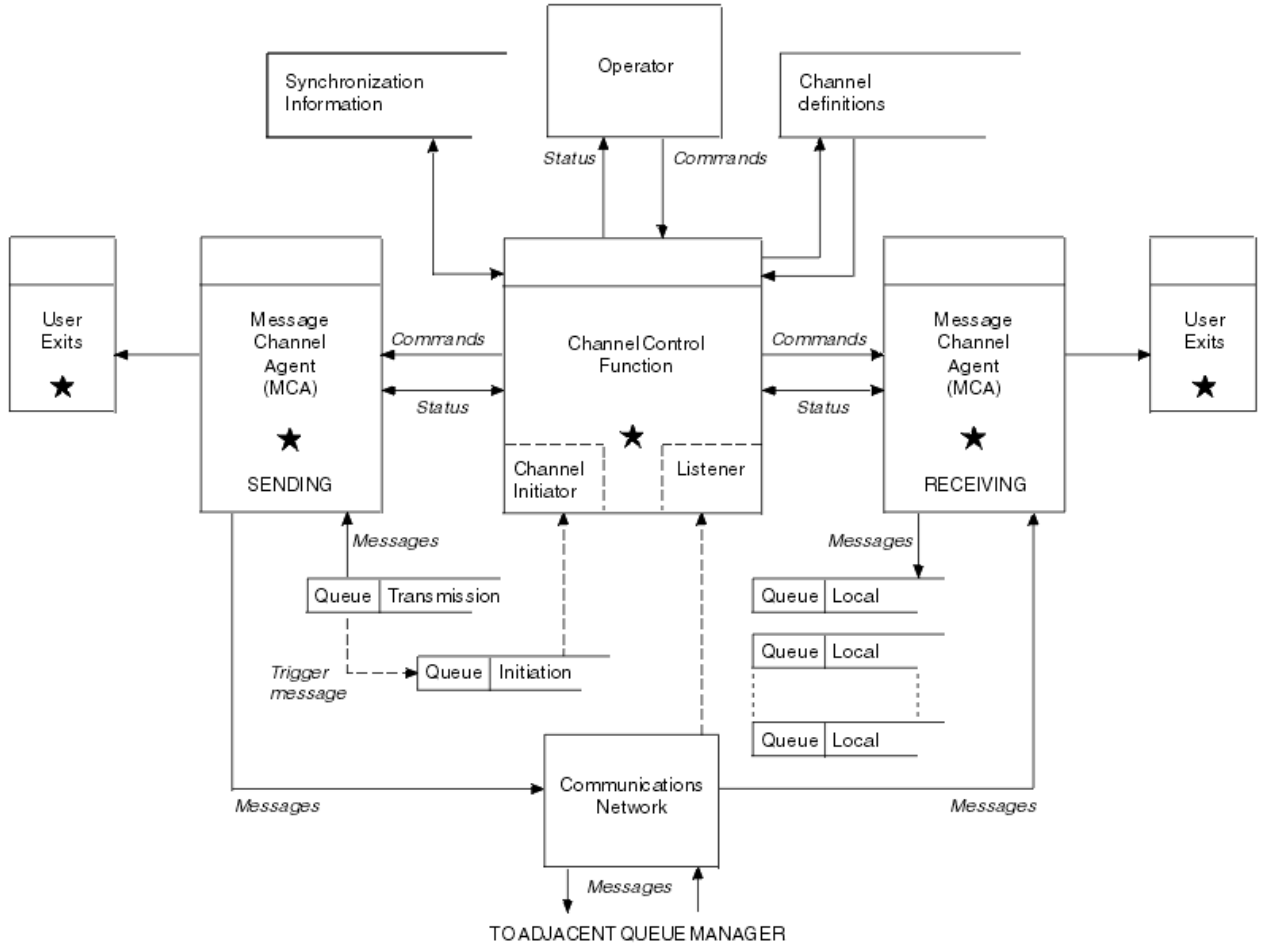
Dağıtılmış kuyruk yönetimi için WebSphere MQ ' u hazırlarken göz önünde bulundurulacak diğer konular. Bu konu, teslim edilemeyen ileti kuyruğunu, Kullanımda kuyrukları, sistem uzantılarını ve kullanıcı çıkışı programlarını ve çalışmakta olan kanalları ve dinleyicileri güvenilir uygulamalar olarak kapsar.

İlgili başvurular

Örnek yapılanış bilgileri

İleti gönderme ve alma

Aşağıdaki şekil, ileti iletildiğinde varlıklar arasındaki ilişkileri ayrıntılandıran dağıtılmış kuyruk yönetimi modelini göstermektedir. Ayrıca, denetim akışının akışını da gösterir.



Şekil 11. Dağıtılmış kuyruk yönetimi modeli

Not:

1. Platforma bağlı olarak kanal başına bir MCA vardır. Belirli bir kuyruk yöneticisi için bir ya da daha çok kanal denetim işlevi olabilir.
2. MCA ' ların ve kanal denetimi işlevlerinin uygulanması, platforma bağlı olarak son derece yüksek bir platforma bağlıdır. Programlar, süreçler ya da iş parçacıkları olabilir ve bunlar tek bir varlık ya da birçok bağımsız ya da bağlantılı parça oluşturan bir çok kişi olabilir.
3. Bir yıldızla işaretlenen tüm bileşenler MQI ' yi kullanabilir.

Kanal parametreleri

MCA, parametrelerinden birini birkaç yoldan alır:

- Bir komut tarafından başlatıldıysa, kanal adı bir veri alanına geçirilir. Daha sonra MCA, doğrudan özniteliklerini almak için kanal tanımını doğrudan okur.
- Gönderen için ve bazı durumlarda sunucu kanallarında MCA, kuyruk yöneticisi tetikleyicisi tarafından otomatik olarak başlatılabilir. Kanal adı, geçerli olduğu yerlerde tetikleyici süreç tanımlamasından alınır ve MCA 'ya geçirilir. Geri kalan işleme, daha önce açıklanan işlemlerle aynıdır. Sunucu kanalları, yalnızca tam olarak nitelenirse tetiklenecek şekilde ayarlanmalıdır; yani, bağlantı kurmak için bir CONNAME belirtiyorlar.
- Bir gönderen, sunucu, istekte bulunan ya da istemci-bağlantı tarafından uzaktan başlatıldıysa, kanal adı, iş ortağı ileti kanalı aracısının ilk verilerinde geçirilir. MCA, bağlantı özelliklerini almak için kanal tanımını doğrudan okur.

Kanal tanımında tanımlı olmayan bazı öznitelikler de pazarlığa açık bir şekilde devam eder:

İletileri böl

Bir uç, bölünmüş iletileri desteklemiyorsa, bölünmüş iletiler gönderilmez.

Dönüştürme yeteneği

Bir uç gerekli kod sayfası dönüştürme işlemini gerçekleştiremezse ya da gerektiğinde sayısal kodlama dönüştürmesi gerçekleştiremezse, diğer ucu bu dönüştürme işlemini gerçekleştirmelidir. İki ucu da destekliyorsa, kanal başlayamaz.

Dağıtım listesi desteği

Bir ucu dağıtım listelerini desteklemiyorsa, iş ortağı MCA iletim kuyruğunda bir işaret ayarlar, böylece birden çok noktaya ilişkin iletilerin kesişmesini bilir.

Kanal durumu ve sıra numaraları

İleti kanalı aracı programları, her bir kanal için geçerli sıra numarası ve mantıksal iş birimi numarasının kayıtlarını ve kanalın genel durumunu kaydeder. Bazı platformlar, kanalları denetlemenize yardımcı olmak için bu durum bilgilerini görüntülemenize olanak sağlar.

Başka bir kuyruk yöneticiye ileti gönderilmesi

Bu bölümde, kuyruk yöneticileri arasında, gerekli önkoşullar ve gereken yetkiler de dahil olmak üzere ileti göndermenin en kolay yolu açıklanır. Başka yöntemler de, iletileri uzak bir kuyruk yöneticisine göndermek için de kullanılabilir.

Bir kuyruk yöneticisinden diğerine ileti göndermeden önce, aşağıdaki adımları yapmanız gerekir:

1. Seçtiğiniz iletişim protokolünüzün kullanılabilir olup olmadığını denetleyin.
2. Kuyruk yöneticilerini başlatın.
3. Kanal başlatıcılarını başlat.
4. Dinleyicileri başlatın.

Gerekli nesnelere yaratmak için doğru WebSphere MQ güvenlik yetkilendirmesine de sahip olmanız.

Bir kuyruk yöneticisinden diğerine ileti göndermek için:

- Kaynak kuyruk yöneticisinde aşağıdaki nesnelere tanımlayın:
 - Gönderen kanalı
 - Uzak kuyruk tanımlaması
 - Başlatma kuyruğu (isteğe bağlı)
 - İletim kuyruğu
 - Ölü-mektup kuyruğu
- Hedef kuyruk yöneticisinde aşağıdaki nesnelere tanımlayın:
 - Alıcı kanalı
 - Hedef kuyruk

- Ölü-mektup kuyruğu

WebSphere MQ platformunuza bağlı olarak, bu nesnelere tanımlamak için birkaç farklı yöntem kullanabilirsiniz:

- On all platforms, you can use the WebSphere MQ script commands (MQSC) described in [MQSC komutları](#) the programmable command format (PCF) commands described in [Yönetim görevlerinin otomatikleştirilmesi](#), or the WebSphere MQ Explorer.

Başka bir kuyruk yöneticisine ileti göndermek için bileşenlerin yaratılmasına ilişkin ek bilgi edinmek için aşağıdaki alt konulara bakın:

İlgili kavramlar

[“Kuyruk yöneticilerinin yaratılması ve yönetilmesi” sayfa 19](#)

İletileri ve kuyrukları kullanabilmeniz için önce en az bir kuyruk yöneticisi ve ilişkili nesnelere yaratmanız ve başlatmanız gerekir.

[“IBM WebSphere MQ dağıtılmış-ileti sistemi teknikleri” sayfa 28](#)

Bu bölümdeki alt konular, kanal planlaması yaparken kullanılan teknikleri açıklar. Bu alt konularda, kuyruk yöneticilerinizin nasıl bağlanacağını planlamanıza yardımcı olacak ve uygulamalarınız arasındaki ileti akışını planlamanıza yardımcı olacak teknikler açıklanmaktadır.

[“Dağıtılmış kuyruk yönetimine giriş” sayfa 47](#)

Distributed queue Management (DQM), kuyruk yöneticileri arasında iletişimi tanımlamak ve denetlemek için kullanılır.

[“Tetikleme kanalları” sayfa 66](#)

WebSphere MQ, kuyruklardaki belirli koşullar karşılandığında otomatik olarak bir uygulama başlatmak için bir olanak sağlar. Bu tesise tetikleme deniyor.

[“İletilerin güvenliği” sayfa 64](#)

Dağıtılmış kuyruk yönetimi, WebSphere MQ' un tipik kurtarma özelliklerine ek olarak, ileti kanalının iki ucu arasında eşgüdümü bir eşitleme noktası yordamı kullanarak iletilerin düzgün bir şekilde teslim edilmesini sağlar. Bu yordam bir hata saptarsa, kanalı kapatır, böylece sorunu araştırabilir ve kanal yeniden başlatılincaya kadar iletileri iletim kuyruğunda güvenli bir şekilde tutar.

[“UNIX, Linux, and Windows üzerindeki kanalların izlenmesi ve denetlenmesi” sayfa 72](#)

DQM için, kanalları uzak kuyruk yöneticilerine yaratmanız, izlemeniz ve denetlemeniz gerekir. Komutları, programları, IBM WebSphere MQ Explorer programlarını, kanal tanımlarına ilişkin dosyaları ve eşitleme bilgileri için bir depolama alanını kullanarak kanalları denetleyebilirsiniz.

[“İstemci ile sunucu arasındaki bağlantıların yapılandırılması” sayfa 96](#)

WebSphere MQ MQI istemcileri ve sunucuları arasındaki iletişim bağlantılarını yapılandırmak için iletişim protokolünüze karar verin, bağlantının her iki ucundaki bağlantıları tanımlayın, bir dinleyici başlatın ve kanalları tanımlayın.

İlgili görevler

[“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156](#)

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlanacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

Kanalların tanımlanması

Bir kuyruk yöneticisinden diğerine ileti göndermek için iki kanal tanımlamanız gerekir. Kaynak kuyruk yöneticisinde bir kanal ve hedef kuyruk yöneticisinde bir kanal tanımlamalısınız.

Kaynak kuyruk yöneticisinde

Kanal tipi SENDER olan bir kanal tanımlayın. Aşağıdaki bilgileri belirtmeniz gerekir:

- Kullanılacak iletim kuyruğunun adı (XMITQ özniteliği).
- İş ortağı sisteminin bağlantı adı (CONNNAME özniteliği).
- Kullanmakta olduğunuz iletişim protokolünün adı (TRPTYPEE özniteliği). On WebSphere MQ for z/OS, the protocol must be TCP or LU6.2. Diğer platformlarda bunu belirtmenize gerek yoktur. Değeri varsayılan kanal tanımınızdan almak için bu programı bırakabilirsiniz.

Tüm kanal özniteliklerine ilişkin ayrıntılar için [Kanal öznitelikleri](#) ' nde yer verilir.

Hedef kuyruk yöneticisinde

Kanal tipi ALICI olan bir kanal ve gönderen kanalıyla aynı adı tanımlayın.

Kullanmakta olduğunuz iletişim protokolünün adını belirtin (TRPTYPEE özniteliği). On WebSphere MQ for z/OS, the protocol must be TCP or LU6.2. Diğer platformlarda bunu belirtmenize gerek yoktur. Değeri varsayılan kanal tanımınızdan almak için bu programı bırakabilirsiniz.

Alıcı kanalı tanımlamaları soysal olabilir. Başka bir deyişle, aynı alıcıyla iletişim kuran birden çok kuyruk yöneticisi varsa, gönderme kanallarının tümü alıcı için aynı adı belirtebilir ve bir günlük nesnesi tanımlaması bunların tümü için geçerli olur.

Not: TRPTYPE parametresinin değeri, yanıt veren ileti kanalı aracısı tarafından yok sayılır. Örneğin, gönderici kanal tanımlamasındaki TCP ' nin TRPTYPE değeri, iş ortağı olarak alıcı kanalı tanımında TRPTYPE LU62 TRPTYPE ile başlar.

Kanalı tanımladığınızda, PING CHANNEL komutunu kullanarak bu kanalı sınavabilirsiniz. Bu komut, gönderen kanaldan alıcı kanalına özel bir ileti gönderir ve döndürülebileni denetler.

Kuyrukların tanımlanması

Bir kuyruk yöneticisinden diğerine ileti göndermek için en çok altı kuyruk tanımlamanız gerekir. Kaynak kuyruk yöneticisinde en çok dört kuyruk tanımlamanız ve hedef kuyruk yöneticisinde en çok iki kuyruk tanımlamanız gerekir.

Kaynak kuyruk yöneticisinde

- Uzak kuyruk tanımlaması

Bu tanımda, aşağıdakileri belirtin:

Uzak kuyruk yöneticisi adı

Hedef kuyruk yöneticisinin adı.

Uzak kuyruk adı

Hedef kuyruk yöneticisindeki hedef kuyruğun adı.

İletim kuyruğu adı

İletim kuyruğunun adı. Bu iletim kuyruğu adını belirtmenize gerek yoktur. Bunu yapmazsanız, hedef kuyruk yöneticisiyle aynı adı taşıyan bir iletim kuyruğu kullanılır. Bu değer yoksa, varsayılan iletim kuyruğu kullanılır. İletim kuyruğuna, hedef kuyruk yöneticisiyle aynı adı vermeniz önerilir; böylece, kuyruğun varsayılan değer olarak kuyruğun bulunması gerekir.

- Başlatma kuyruğu tanımlaması

z/OS' ta gerekli ve diğer platformlarda isteğe bağlı. Kullanıma hazırlama kuyruğunu SYSTEM.CHANNEL.INITQadlandırmasını dikkate alın. diğer platformlarda.

- İletim kuyruğu tanımlaması

USAGE özniteliği XMITQ değerine ayarlanmış yerel bir kuyruk.

- Ölü-mektup kuyruğu tanımlaması

Teslim edilmeyen iletilerin yazılabildiği bir ölü harf kuyruğu tanımlayın.

Hedef kuyruk yöneticisinde

- Yerel kuyruk tanımlaması

Hedef kuyruk. Bu kuyruğun adı, kaynak kuyruk yöneticisinde bulunan uzak kuyruk tanımlamasının uzak kuyruk adı alanında belirtilenlerle aynı olmalıdır.

- Ölü-mektup kuyruğu tanımlaması

Teslim edilmeyen iletilerin yazılabildiği bir ölü harf kuyruğu tanımlayın.

İlgili kavramlar

[“İletim Kuyruğu Yaratılması” sayfa 52](#)

Bir kanaldan (istekte bulunan bir kanaldan başka) başlatılabilmesi için, iletim kuyruğunun bu bölümde anlatıldığı gibi tanımlanması gerekir. İletim kuyruğu, kanal tanımlamasında adlandırılmalıdır.

İletim Kuyruğu Yaratılması

Bir kanaldan (istekte bulunan bir kanaldan başka) başlatılabilmesi için, iletim kuyruğunun bu bölümde anlatıldığı gibi tanımlanması gerekir. İletim kuyruğu, kanal tanımlamasında adlandırılmalıdır.

Her ileti gönderme kanalı için, USAGE özniteliği XMITQ değerine ayarlanmış bir yerel kuyruk tanımlayın. Uzak kuyruk tanımlarınızda belirli bir iletim kuyruğunu kullanmak istiyorsanız, gösterildiği gibi bir uzak kuyruk yaratın.

Bir iletim kuyruğu yaratmak için aşağıdaki örneklerde gösterildiği gibi, WebSphere MQ Commands (MQSC) komutunu kullanın:

İletim kuyruğu örneği yarat

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') USAGE(XMITQ)
```

Uzak kuyruk örneği yarat

```
DEFINE QREMOTE(PAYROLL) DESCR('Remote queue for QM2') +  
XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Örneklerde gösterildiği gibi, uzak sistemdeki kuyruk yöneticisi adının iletim kuyruğunu adlandırıp adlandırmasını dikkate alın.

Kanal başlatılıyor

Kaynak kuyruk yöneticisinde tanımlı uzak kuyruğa ileti yerleştirdiğinizde, kanal başlatılıncaya kadar iletim kuyruğunda saklanır. Kanal başlatıldığında, iletiler uzak kuyruk yöneticilikteki hedef kuyruğa teslim edilir.

START CHANNEL komutunu kullanarak, gönderme kuyruğu yöneticilerinin kanalını başlatın. Gönderme kanalını başlattığınızda, alma kanalı otomatik olarak (dinleyici tarafından) başlatılır ve iletiler hedef kuyruğa gönderilir. İletilerin aktarılacağı iletiler için, ileti kanalının her iki ucu da çalışır durumda olmalıdır.

Kanalın iki ucu farklı kuyruk yöneticilerinde olduğundan farklı özniteliklerle tanımlanabilirler. Farklılıkları çözmek için, kanal başlatıldığında iki uç arasında bir ilk veri anlaşması vardır. Genel olarak, kanalın iki ucu daha az kaynağa ihtiyaç duyan özniteliklerle birlikte çalışır. Bu, daha büyük sistemlerin, ileti kanalının diğer ucundaki küçük sistemlerin daha küçük kaynaklarını barındırabilmesini sağlar.

Gönderme MCA, kanala göndermeden önce büyük iletileri böler. Bunlar uzak kuyruk yöneticisinde yeniden birleştirilir. Bu, kullanıcı için belirgin değildir.

MCA, birden çok iş parçacığını kullanarak ileti aktarabilir. *pipelining* adı verilen bu işlem, MCA 'nın iletileri daha verimli bir şekilde ve daha az bekleme durumu ile aktarmasını sağlar. Pipelining kanal performansını artırır.

Kanal denetimi işlevi

Kanal denetim işlevi, kanalları tanımlamanız, izlemeniz ve denetlemeniz için gerekli olanakları sağlar.

Komutlar, panolar, programlar ya da bir komut satırından kanal denetimi işlevine verilir. Pano arabirimi, kanal durumu ve kanal tanımlama verilerini de görüntüler. Programlanabilir Komut Biçimlerini ya da [“UNIX, Linux, and Windows üzerindeki kanalların izlenmesi ve denetlenmesi” sayfa 72](#)' de ayrıntılı olarak açıklanan WebSphere MQ komutlarını (MQSC) ve denetim komutlarını kullanabilirsiniz.

Komutlar aşağıdaki gruplara ayrılır:

- Kanal yönetimi
- Kanal denetimi
- Kanal durumu izleme

Kanal yönetimi komutları, kanalların tanımlarıyla başa çıkabiliyor. Bu bilgiler aşağıdakileri gösterebilmenizi sağlar:

- Kanal tanımlaması yarat
- Kanal tanımlamasını kopyala
- Kanal tanımlamasını deęiřtirme
- Kanal tanımlamasının silinmesi

Kanal denetimi komutları, kanalların alıřmasını ynetir. Bu bilgiler ařaęıdakileri gsterebilmenizi saęlar:

- Bir kanal bařlatma
- Kanalı durdur
- İř ortaęıyla yeniden eřitile (bazı somutlamalarda)
- İleti sıra numaralarını ilk durumuna getir
- Belirsiz bir ileti kmesini zmlle
- Ping; kanaldan bir test iletiřimi gnder

Kanal izleme, kanalların durumunu grntler; rneęin:

- Geerli kanal ayarları
- Kanalin etkin mi, yoksa etkin deęil mi?
- Kanalin eřitlenmiř durumda sonlandırılıp sonlandırılmayacaęı

Kanalları tanımlama, denetleme ve izleme hakkında daha fazla bilgi iin ařaęıdaki alt bařlıklara bakın:

Kanallar hazırlanıyor

Bir ileti kanalı ya da MQI kanalı bařlatma giriřiminde bulunmadan nce, kanalı hazırlamanız gerekir. Yerel ve uzak kanal tanımlamalarının tm zniteliklerinin doęru ve uyumlu olduęundan emin olmanız gerekir.

[Kanal znitelikleri](#) , kanal tanımlarını ve zniteliklerini aıklar.

Aık kanal tanımlarını ayarlamanıza raęmen, kanal grřmeleri bir kanal bařlatıldıęında yrtldęnde, tanımlanan deęerlerin birini ya da dięerini geersiz kılabilir. Bu davranıř olaęandır ve kullanıcı iin belirgin deęildir ve tersi durumda uyumsuz tanımların birlikte alıřabilmesi iin bu Őekilde dzenlenmiřtir.

Gnlk nesnesi ve sunucu baęlantısı kanallarının otomatik tanımlaması

z/OSdışındaki tm platformlarda WebSphere MQ dıřında, uygun kanal tanımlaması yoksa, otomatik tanımlaması etkin olan bir alıcı ya da sunucu baęlantısı kanalı iin otomatik olarak bir tanımlama yaratılır. Tanım Őu kullanılarak yaratılır:

1. Uygun model kanalı tanımlaması (SYSTEM.AUTO.RECEIVERya da SYSTEM.AUTO.SVRCONN. Otomatik tanımlama iin model kanalı tanımlamaları, sistem varsayılan deęerleri (SYSTEM.DEF.RECEIVERve SYSTEM.DEF.SVRCONN, aıklama alanı dıřında, "Auto-defind by" (Otomatik olarak tanımlanan) ve ardından 49 boř bořluk. Sistem yneticisi, saęlanan model kanalı tanımlamalarının herhangi bir blmn deęiřtirmeyi seebilir.
2. İř ortaęı sisteminden gelen bilgiler. İř ortaęıdaki deęerler kanal adı ve sıra numarası sarma deęeri iin kullanılır.
3. Otomatik tanımlama tarafından yaratılan deęerleri deęiřtirmek iin kullanabileceęiniz bir kanal ıkıř programı. Bkz. [Channel auto-definition exit programı](#).

Daha sonra, bir otomatik tanımlama ıkıřı tarafından deęiřtirilip deęiřtirilmedięini ya da model tanımlamasının deęiřtirilip deęiřtirilmedięini belirlemek iin aıklama imlenmiř olur. İlk 44 karakter hala "Otomatik olarak tanımlanmıřsa", arkasından 29 bořluk karakteri eklenirse, kuyruk yneticisi adı eklenir. Son 20 karakter yine de bořluksa, yerel saat ve tarih eklenir.

Tanım yaratılıp saklandıęında, kanal bařlatma iřlemi her zaman var olsa da devam eder. Toplu iř byklę, iletim byklę ve ileti byklę iř ortaęıyla grřlr.

Diğer nesnelerin tanımlanması

Bir ileti kanalının başlatılabilmesi için, kuyruk yöneticilerinde her iki ucun da tanımlanmış (ya da otomatik tanımlama için etkinleştirilmiş) olması gerekir. Hizmet vermek üzere olduğu iletim kuyruğu, gönderme uçta kuyruk yöneticisinde tanımlanmalıdır. İletişim bağlantısı tanımlanmalıdır ve kullanılabilir olmalıdır. “Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27’inde açıklanan senaryoları uygulamak için, diğer WebSphere MQ nesnelerini (uzak kuyruk tanımları, kuyruk yöneticisi diğer adı tanımlamaları ve yanıt kuyruğu diğer adı tanımları gibi) hazırlamanız gerekebilir.

MQI kanallarını tanımlama hakkında bilgi için bkz. “MQI kanallarının tanımlanması” sayfa 109.

İletim kuyruğu başına birden çok ileti kanalı

İletim kuyruğu başına birden çok kanal tanımlanması mümkündür, ancak bu kanallardan yalnızca biri herhangi bir zamanda etkin olabilir. Trafik dengeleme ve bağlantı hatası düzeltici işlem için kuyruk yöneticileri arasında alternatif rotaların sağlanması için bu seçeneği göz önünde bulundurun. Önceki kanal, gönderme bitiminde bir ileti kümesini belirsiz bir şekilde bırakması durumunda, bir iletim kuyruğu başka bir kanal tarafından kullanılamaz. Daha fazla bilgi için “Belirsiz kanallar” sayfa 63 başlıklı konuya bakın.

Kanal başlatma

Bir kanal, iletileri dört yoldan biriyle iletmeye başlamasına neden olabilir. Bu durumda şunlar olabilir:

- Bir işletmen tarafından başlatıldı (alıcı, küme-alıcı ya da sunucu bağlantısı kanalları).
- İletim kuyruğundan tetiklenir. Bu yöntem, yalnızca gönderen kanalları ve tam olarak nitelenmiş sunucu kanalları (CONNNAME belirten kanallar) için geçerlidir. Kanalları tetiklemek için gerekli nesnelere hazırlamanız gerekir.
- Bir uygulama programından (alıcı, küme-alıcı ya da sunucu bağlantısı kanallarından değil) başlatıldı.
- Bir gönderen, kümeden gönderen, istekçi, sunucu ya da istemci-bağlantı kanalı tarafından ağdan uzaktan başlatıldı. Alıcı, küme alıcılı ve büyük olasılıkla sunucu ve istekte bulunan kanal iletimleri bu şekilde başlatılır; dolayısıyla, sunucu bağlantısı kanalları da vardır. Kanalların kendileri zaten başlatılmalı (yani, etkin).

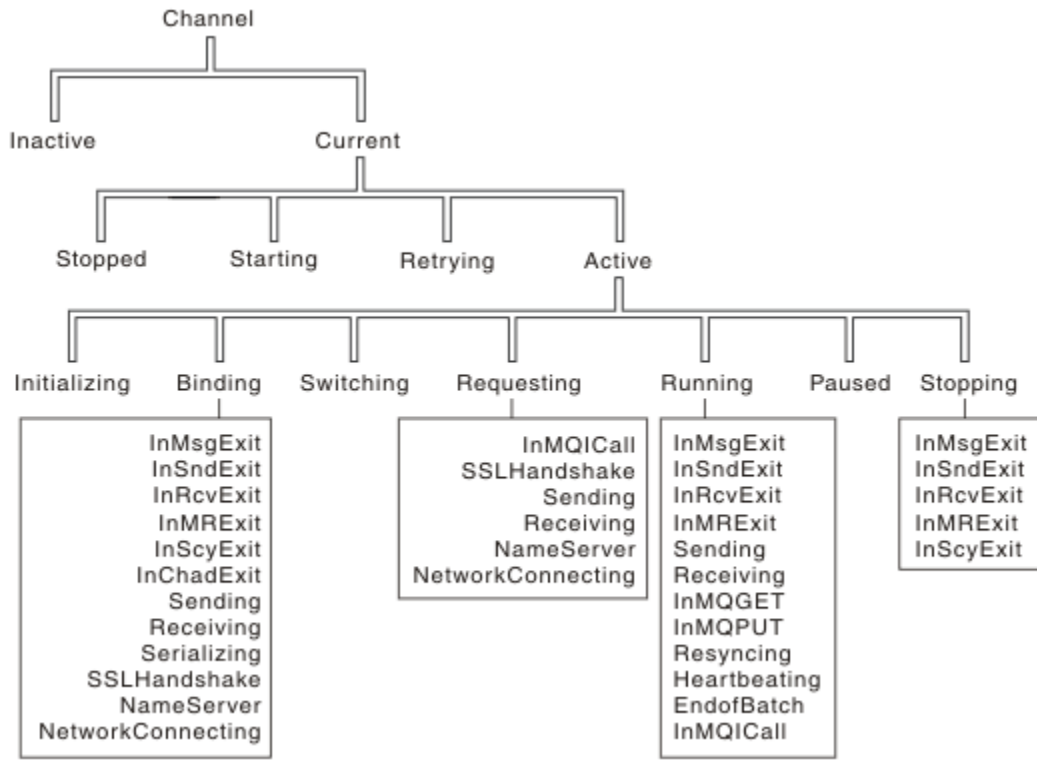
Not: çünkü bir kanal 'başlatılmış' olduğu için mutlaka mesaj iletilmiyor. Bunun yerine, daha önce açıklanan dört olaydan biri gerçekleştiğinde, iletim işlemini başlatmak için 'enabled' (etkin) olabilir. Bir kanala ilişkin etkinleştirme ve devre dışı bırakma işlemi, START ve STOP işleciyle birlikte gerçekleştirilir.

Kanal durumları

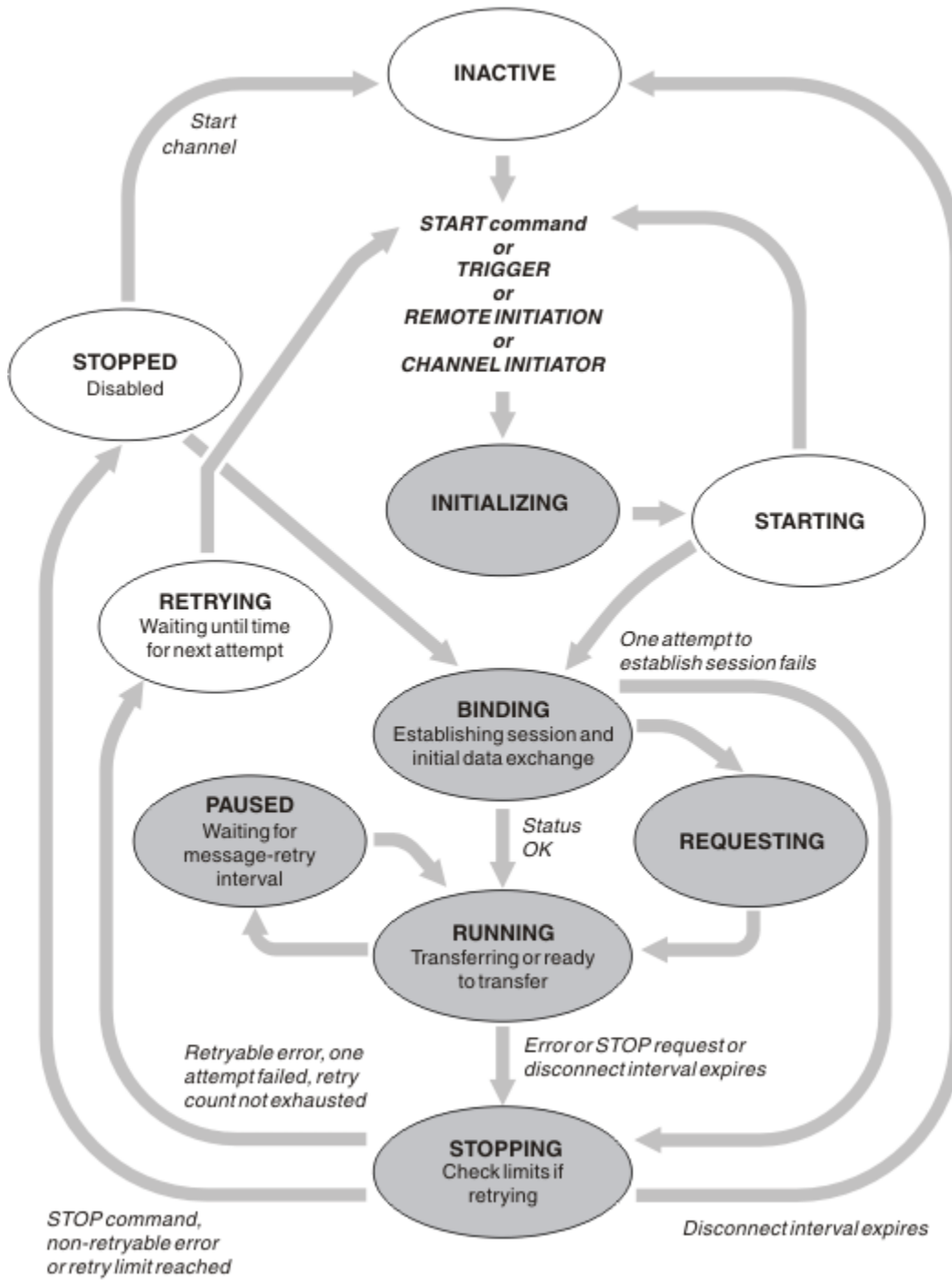
Bir kanal, her an birçok eyaletten birinde olabilir. Bazı eyaletlerde alt devletler de vardır. Belirli bir durumdan bir kanal başka eyaletler arasında geçiş yapabilir.

Şekil 12 sayfa 55 , olası tüm kanal durumlarının sıradüzenini ve kanal durumlarının her biri için geçerli olan alt durumları gösterir.

Şekil 13 sayfa 56 , kanal durumları arasındaki bağlantıları gösterir. Bu bağlantılar, tüm ileti kanalı ve sunucu bağlantısı kanallarına uygulanır.



Şekil 12. Kanal durumları ve alt durumları



Şekil 13. Kanal durumları arasındaki akışlar

Yürürlükteki ve etkin

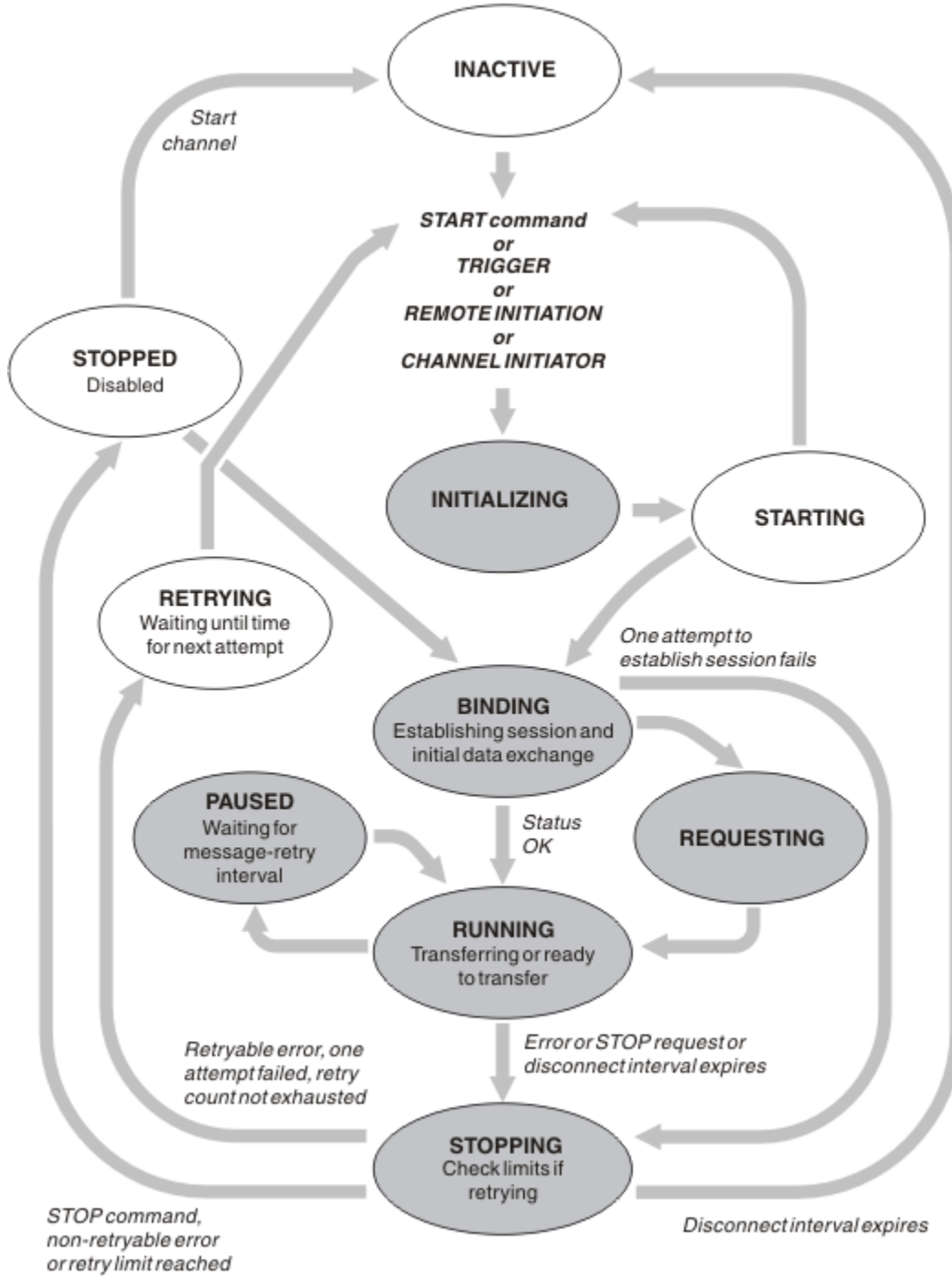
Kanal, etkin olmayan herhangi bir durumdaysa *yürürlükteki* olur. Yürürlükteki kanal, REDICATING, DURING ya da STARTING durumunda değilse *etkin* . Bir kanal etkin olduğunda, kaynak tüketir ve bir işlem ya da iş parçacığı çalışıyor demektir. activeiçinde etkin bir kanalın olası yedi durumu (INITIALING, BINDING, ANAHTARLAMA, ISTEME, ÇALIŞTIRMA, DURAKLAT YA DA DURDURMA), Şekil 13 sayfa 56içinde vurgulanır

Etkin bir kanal, kanalın tam olarak ne yapmakta olduğu konusunda daha fazla ayrıntı veren bir alt durumu gösterebilir. Her bir duruma ilişkin alt durumlar Şekil 12 sayfa 55içinde gösterilir.

Yürürlükteki ve etkin

Kanal, etkin olmayan herhangi bir durumdaysa "yürürlükteki" olur. Yürürlükteki kanal, REDICATING, DURING ya da STARTING durumunda değilse "etkin" .

Bir kanal "etkin" ise, kanalın tam olarak ne yaptığına ilişkin daha fazla ayrıntı veren bir alt durum da gösterilebilir.



Şekil 14. Kanal durumları arasındaki akışlar

Not:

1. Bir kanal, Şekil 14 sayfa 57 içinde vurgulanan altı durumdan birinin (INITIALING, BINDING, ISTEME, RUNING, DURAKLATILAN ya da DURDURMA) olduğunda, kaynak ve bir işlem ya da iş parçacığı çalışıyor; kanal *etkin*.

2. Kanal DURDURULDU durumunda olduğunda, sonraki durum henüz bilinmediği için oturum etkin olabilir.

Yürürlükteki kanal sayısı üst sınırının belirtilmesi

Aynı anda geçerli olabilecek kanal sayısı üst sınırını belirtebilirsiniz. Bu sayı, kanal durumu çizelgesinde, durdurulan kanallar ve yeniden denenmiş kanallar da içinde olmak üzere, giriş yapan kanalların sayısıdır. Specify this using the queue manager configuration file for UNIX and Linux systems, or the WebSphere MQ Explorer. Kullanıma hazırlama ya da yapılanış kütüğü kullanılarak ayarlanan değerler hakkında daha fazla bilgi için [Dağıtılmış kuyruğa alma için yapılandırma dosyası stanzaları](#) başlıklı konuya bakın. For more information about specifying the maximum number of channels, see [IBM WebSphere MQ uygulamasını yönetme](#) for WebSphere MQ for UNIX and Linux systems, and Pencereler systems.

Not:

1. Sunucu bağlantısı kanalları bu sayıya dahil edilir.
2. Bir kanal, etkin duruma gelmeden önce geçerli olmalıdır. Bir kanal başlatıldıysa, ancak yürürlükteki bir kanal başlatılamazsa, başlatma işlemi başarısız olur.

Etkin kanal sayısı üst sınırının belirtilmesi

Ayrıca, sisteminizin birçok başlangıç kanalı tarafından aşırı yüklenmesini önlemek için etkin kanal sayısı üst sınırını da belirleyebilirsiniz. Bu yöntemi kullanırsanız, diğer kanallar sona erdirildikçe, bağlantı kesme aralığı öznitelikliğini, bekleme kanallarının en kısa sürede başlamasını sağlamak için düşük bir değere ayarlayın.

Her kanalda, ortağıyla bağlantı kurma girişiminde bulunmaya çalışılan bir kanal, etkin bir kanal haline gelmelidir. Girişim başarısız olursa, bir sonraki deneme süresi oluncaya kadar etkin olmayan bir geçerli kanal olarak kalır. Bir kanal yeniden deneme sayısının ve sıklığın yeniden deneme sayısı ve yeniden deneme aralığı kanal öznitelikleri tarafından belirlendiği sayı. Bu özniteliklerin her ikisi için kısa ve uzun değerler vardır. Ek bilgi için [Kanal öznitelikleri](#) başlıklı konuya bakın.

Bir kanal etkin bir kanal olmak zorunda olduğunda (bir START komutu verildiğinden ya da tetiklendiği için ya da başka bir yeniden deneme girişimi zamanı olduğu için), ancak etkin kanal sayısı üst sınır değerinde olduğu için, kanal, etkin yuvalardan biri serbest bırakılıncaya kadar, başka bir kanal yönetim ortamı etkinleşmesini durduruncaya kadar bekleme durumuna neden olur. Ancak, bir kanal, uzaktan başlatıldığı için başlatılıyorsa ve o sırada kullanılabilir etkin yuva yoksa, uzaktan başlatma reddedilir.

İsteyen kanalın dışında bir kanal etkin olmaya çalışırsa, STARTING durumuna girer. Bu durum, yalnızca kısa bir süre için STARTING durumunda olmasına rağmen, hemen kullanılabilir bir etkin yuva olsa da ortaya çıkar. Ancak, kanal etkin bir yuvayı beklemek zorunda kalıyorsa, bekleme sırasında STARTING durumunda olur.

İstekte bulunanın kanalları STARTING durumuna girmiyor. Etkin kanal sayısı sınırda olduğu için bir istekte bulunan kanal başlatılamazsa, kanal olağan dışı bir şekilde sona erer.

Bir kanal, bir requester kanalı dışında, etkin bir yuva alamıyor ve bir kanal için bekleme işlemi yapılırsa, günlüğüne bir ileti yazılır ve bir olay oluşturulur. Bir yuva daha sonra serbest bırakıldığında ve kanal bunu elde edebildiğinde, başka bir ileti ve olay oluşturulur. Kanal bir yuvayı düz bir şekilde edinebiliyorsa, bu olaylar ve iletiler de oluşturulmaz.

Kanal etkin duruma gelmesini beklerken STOP CHANNEL komutu verilirse, kanal durdurulmuş duruma geçer. Kanal Durdurulmuş bir olay ortaya çıktı.

Sunucu bağlantısı kanalları, etkin kanal sayısı üst sınırında yer alır.

For more information about specifying the maximum number of active channels, see [IBM WebSphere MQ uygulamasını yönetme](#) for WebSphere MQ for UNIX and Linux systems, and Pencereler systems.

Kanal hataları

Kanallardaki hatalar, kanalın daha fazla iletim durdurmasına neden olur. Kanal bir gönderen ya da sunucuysa, RETRY durumuna geçer; sorunun kendisini temizleyebilir olması mümkündür. RETRY durumuna geçemiyorsa, kanal durdurulmuş durumuna geçer.

İletilerin gönderilmesi için, ilişkili iletim kuyruğu GET (DISABLE) olarak ayarlanır ve tetikleme kapatılır. (STATUS (STOP) olan bir STOP komutu, bu durumu STOP durumuna getiren tarafı alır; yalnızca, bağlantı kesme aralığının süre bitimi ya da STATUS (INACTIVE) olan bir STOP komutu, olağan durumda sona ermesini ve etkinlik dışı duruma gelmesini sağlar.) Durdurulmuş durumda olan kanallar yeniden başlatılabilmesi için işletmen müdahalesini gerekir (bkz. [“Durdurulan kanallar yeniden başlatılıyor” sayfa 62](#)).

Not: UNIX, Linux ve Windows sistemlerinde, yeniden deneme girişiminde bulunulması için bir kanal başlatıcısı çalışır durumda olmalıdır. Kanal başlatıcı yoksa, kanal devre dışı olur ve el ile yeniden başlatılması gerekir. Kanalı başlatmak için bir komut dosyası kullanıyorsanız, komut dosyasını çalıştırmayı denemeden önce kanal başlatıcısının çalıştığından emin olun.

Uzun yeniden deneme sayısı (LONGRTY) , yeniden çalışmanın nasıl yeniden deneneceğini açıklar. Hata temizlenirse, kanal otomatik olarak yeniden başlatılır ve iletim kuyruğu yeniden etkinleştirilir. Hata temizlenmeden yeniden deneme sınırına ulaşırsa, kanal durdurulmuş duruma geçer. Durdurulmuş bir kanal, işletmen tarafından el ile yeniden başlatılmalıdır. Hata hala mevcutsa, yeniden yeniden denenmez. Başarılı bir şekilde başlatılırsa, iletim kuyruğu yeniden etkinleştirilir.

If the queue manager stops while a channel is in RETRYING or STOPPED status, the channel status is remembered when the queue manager is restarted. Ancak, kanal durdurulmuş durumdayken kuyruk yöneticisi durduğunda, SVRCONN kanal tipine ilişkin kanal durumu sıfırlanır.

Bir kanal, kuyruğun dolu olduğu ya da engellendiği için bir iletiyi hedef kuyruğa koyamıyorsa, kanal işlemi bir zaman aralığında (ileti-yeniden deneme sayısı özniteliinde belirtilen) bir zaman sayısını (ileti-yeniden deneme aralığı öznitelide belirtilir) yeniden deneyebilir. Diğer bir seçenek olarak, kendi iletinizi yazabilirsiniz; bu durumda, hangi koşulların yeniden deneneceğini ve yapılan denemelerin sayısını belirleyen yeniden deneme çıkışıdır. Kanal, iletinin yeniden deneme aralığının bitmesini beklerken PAUSED durumuna geçer.

Kanal öznitellikleriyle ilgili bilgi için bkz. [Kanal öznitellikleri](#) ve iletiyle ilgili çıkışa ilişkin bilgi için [ileti alışverişi kanallarına ilişkin kanal çıkışı programları](#) .

Sunucu bağlantısı kanal sınırları

Sunucu bağlantısı kanal sınırlarını, istemci uygulamalarının yorucu kuyruk yöneticisi kanalı kaynaklarının (**MAXINST**) olmasını önlemek ve yorucu bir sunucu bağlantısı kanal kapasiteden tek bir istemci uygulamasını önlemek için **MAXINSTC**' yi ayarlayabilirsiniz.

Tek bir kuyruk yöneticisinde, herhangi bir zamanda en çok kanal sayısı üst sınırı etkin olabilir. Sunucu bağlantısı kanal eşgörünümlerinin toplam sayısı, etkin kanal sayısı üst sınırına eklenir.

Başlatılabilecek bir sunucu bağlantısı kanalının eşzamanlı eşgörünümlerinin üst sınır sayısını belirtmezseniz, tek bir istemci uygulaması için tek bir sunucu bağlantısı kanalına bağlanabilir ve kullanılabilir olan etkin kanal sayısı üst sınırının egzozuna geçilebilir. Etkin kanal sayısı üst sınırına ulaşıldığında, kuyruk yöneticisine diğer kanalların başlatılmalarını önler. Bu durumu önlemek için, hangi istemci tarafından başlatılmış olursa olsun, başlatılabilecek tek bir sunucu bağlantısı kanalının eş zamanlı eşgörünümlerinin sayısını sınırlamanız gerekir.

Sınırın değeri, sunucu bağlantı kanalının şu anda çalışmakta olan eşgörünümlerinin altına indirgenirse, sıfıra bile, çalışmakta olan kanallar etkilenmez. Çalışmakta olan eşgörünümlerinin sayısının sınırdan daha az olması için, var olan eşgörünümlerin çalıştırılması başarısız oluncaya kadar yeni eşgörünümler başlatılamaz.

Ayrıca, birçok farklı istemci-bağlantı kanalı tek bir sunucu bağlantısı kanalına bağlanabilir. Hangi istemci tarafından başlatılmış olursa olsun, başlatılabilecek tek bir sunucu bağlantısı kanalının eşzamanlı eşgörünümlerinin sayısı sınırı, istemcinin kuyruk yöneticisinin etkin kanal sığasını üst sınırını tüketmesini önler. Tek bir istemciden başlatılabilecek tek bir sunucu bağlantısı kanalının eş zamanlı eşgörünümlerinin sayısını sınırlamasanız da, tek bir hatalı istemci uygulamasının bu kadar çok bağlantıyı açması mümkün

olur. Bu nedenle, tek bir sunucu bağlantısı kanalı için ayrılan kanal sığasını boşaltır ve bu nedenle, kanalı kullanması gereken diğer istemcileri de bu kanala bağlanmasını önler. Bu durumu önlemek için, tek bir istemciden başlatılabilecek tek bir sunucu bağlantısı kanalının eş zamanlı eşgörünümlerinin sayısını sınırlamanız gerekir.

Bireysel istemci sınırının değeri, tek tek istemcilerden çalışmakta olan sunucu bağlantısı kanalının eşgörünümlerinin sayısının altında azaldıysa, sıfıra bile, çalışan kanalların etkilenmemesi gerekir. Ancak, sunucu bağlantısı kanalının yeni yönetim ortamları yeni sınırı aşan bir istemciden başlatılamaz; böylece, çalışmakta olan eşgörünümlerinin sayısı bu parametrenin değerinden küçük olacak şekilde, o istemciden gelen yeterli sayıda var olan yönetim ortamları çalıştırılmadıkça yeni sınırı aşamaz.

Kanalın diğer ucunun hala kullanılabilir olup olmadığını kontrol etme

Kanalın diğer ucunun kullanılabilir olup olmadığını denetlemek için sağlıklı işletim bildirim aralığını, canlı tutma aralığını ve alma zaman aşımını kullanabilirsiniz.

Kalp atışları

İletim kuyruğunda hiçbir ileti olmadığına, akışlar gönderen MCA ' dan geçirileceğini belirtmek için sağlıklı işletim bildirim aralığı kanal özneliğini, Heartbeat interval (HBINT) içinde anlatıldığı gibi kullanabilirsiniz.

Canlı tut

In WebSphere MQ for UNIX, Linux, and Pencereler systems, if you are using TCP as your transport protocol, you can set `keepalive=yes`. Bu seçeneği belirlerseniz, TCP belirli aralıklarla bağlantının diğer ucunun hala kullanılabilir olup olmadığını denetler. Öyle değil, kanal yok edildi. Bu seçenek, Keepalive Interval (KAINT) içinde açıklanır.

If you have unreliable channels that report TCP errors, use of the **Keepalive** option means that your channels are more likely to recover.

Keepalive seçeneğinin işleyişini denetlemek için zaman aralıkları belirtebilirsiniz. Zaman aralığını değiştirdiğinizde, değişiklik etkilendikten sonra yalnızca TCP/IP kanalları başlatılır. Zaman aralığı için seçtiğiniz değerin, kanala ilişkin bağlantı kesme aralığı değerinden küçük olduğundan emin olun.

Keepalive seçeneğini kullanma hakkında daha fazla bilgi için DEFINE CHANNEL komutundaki KAINT parametresine bakın.

Alma zaman aşımı

İletim protokolünüz olarak TCP kullanıyorsanız, bir dönem için veri alınmazsa, boşta duran bir MQI dışı kanal bağlantısının alıcı sonu da kapatılır. Bu dönem, *alma zaman aşımı* değeri, HBINT (sağlıklı işletim bildirim aralığı) değerine göre belirlenir.

WebSphere MQ for UNIX, Linuxve Pencereler sistemlerinde, *alma zaman aşımı* değeri aşağıdaki gibi ayarlanır:

1. İlk akış sayısı için, herhangi bir anlaşma gerçekleşmeden önce, *alma zaman aşımı* değeri, kanal tanımından HBINT değerinin iki katı olur.
2. Kanallar bir HBINT değerini kararlaştırdıktan sonra, HBINT 60 saniyeden az bir değere ayarlanırsa, *alma zaman aşımı* değeri bu değerin iki katı olarak ayarlanır. HBINT 60 saniye ya da daha fazla değere ayarlıysa, *alma zaman aşımı* değeri HBINT değerinden 60 saniye daha büyük olarak ayarlanır.

Not:

1. Değerlerden herhangi biri sıfırda, zaman aşımı yoktur.
2. Kalp atışlarını desteklemeyen bağlantılar için, HBINT değeri 2. adımda sıfıra, dolayısıyla zaman aşımı yoktur; bu nedenle TCP/IP KEPENCE değerini kullanmanız gerekir.
3. Paylaşımı paylaşan istemci bağlantıları için, yalnızca bir MQGET olağanüstü olduğunda değil, tüm kanalda kalp atışları kanal genelinde (her iki uçtan da) akabilir.

4. Paylaşım konuşmalarının kullanılmadığı istemci bağlantıları için, yalnızca istemci bir MQGET çağrısıyla ilgili olarak bekleme ile ilgili sorunlar olduğunda kalp atışları sunucudan aklanır. Bu nedenle, sağlıklı işletim bildirim aralığını istemci kanalları için çok küçük olarak ayarlamamız önerilmez. Örneğin, sağlıklı işletim bildirim 10 saniye olarak ayarlandıysa, bu süre boyunca veri akışı yapılmadığı için kesinleştirme işlemi 20 saniyeden uzun sürerse, MQCMIT çağrısı başarısız olur (MQRC_CONNECTION_BROKEN ile). Bu, büyük iş birimleriyle gerçekleşebilir. Ancak, sağlıklı işletim bildirim aralığı için uygun değerler seçildiyse, yalnızca MQGET ile bekleme süresi önemli zaman aralıklarına sahip olduğunda bu durum gerçekleşmez.

SHARECNV değeri sıfır değilse, istemci tam çift yönlü bağlantı kullanır; bu bağlantı, istemcinin tüm MQI çağrıları sırasında sağlıklı işletim bildirim (ve yaptığı) sağlıklı bir şekilde sağlıklı işletim bildirim yapabildiği anlamına gelir.

5. WebSphere MQ Sürüm 7 Client kanallarında, sağlıklı işletim sistemi, istemci tarafında olduğu kadar hem sunucudan da hem de sunucudan akabilir. Her iki uçta da zamanaşımı, 60 saniyeden az olan HBYT 'ler için 2*HBINT ' e ve 60 saniyeden fazla HBYT 'ler için HBINT+60 ' e bağlıdır.
6. İki kez sinyal aralığından sonra bağlantının iptal edilmesi, en az her sinyal aralığında bir veri ya da sağlıklı işletim akışı beklediğinden, sinyal geçerli olur. Sağlıklı işletim bildirim aralığını çok küçük olarak ayarlamak, özellikle kanal çıkışlarını kullanıyorsanız sorunlara neden olabilir. Örneğin, HBINT değeri bir saniye ise, gönderme ya da alma çıkışı kullanılırsa, alma ucu kanalı iptal etmeden önce yalnızca 2 saniye bekler. MCA, iletiyi şifrelemek gibi bir görev gerçekleştiriyorsa, bu değer çok kısa olabilir.

Benimseyen MCA

Benimseyen MCA işlevi, IBM WebSphere MQ Explorer ' in bir alıcı kanalını iptal etmesini ve yerine yeni bir alıcı göndermesini sağlar.

Bir kanal bir iletişim hatasına zarar veriyorsa, alıcı kanalı 'iletişim alma' durumunda bırakılabilir. İletişimler yeniden kurulduktan sonra, gönderen kanalı yeniden bağlanmayı dener. Uzak kuyruk yöneticisi, alıcı kanalının çalışmakta olduğunu bulursa, aynı alıcı kanalının başka bir sürümünün başlatılmasına izin vermez. Bu sorun, sorunu gidermek ya da sistemin canlı tutma (keepalive) kullanımını gidermek için kullanıcı müdahalesi gerektirir.

Benimseme MCA işlevi, sorunu otomatik olarak çözer. IBM WebSphere MQ Explorer ' un bir alıcı kanalını iptal etmesini ve yerine yeni bir tane başlatmasını sağlar.

İşlev çeşitli seçeneklerle ayarlanabilir. **distributed** Dağıtılmış platformlar için bkz. [Yönetme](#).

Kanalların durdurulması ve durdurulması

Bu konuda, bağlantı kesme zaman aralığı sona ermeden önce bir kanalı nasıl durdurabileceğiniz ve susturabileceğiniz açıklanır.

İleti kanalları, kuyruk yöneticileri arasında, yalnızca bağlantı kesme aralığı kanal özneliği tarafından denetlenen sonlandırma yoluyla, kuyruk yöneticileri arasında uzun süre çalışan bağlantılar olacak şekilde tasarlanmıştır. Bu düzenek, işletmenin bağlantı kesme zaman aralığı sona ermeden önce kanalı sona erdirmesi gerekmediği sürece iyi çalışır. Bu gereksinime aşağıdaki durumlarda oluşabilir:

- Sistem susturma
- Kaynak tasarrufu
- Bir kanalın bir ucunda tek taraflı eylem

Bu durumda, kanalı durdurabilirsiniz. Bunu kullanarak şunları yapabilirsiniz:

- STOP CHANNEL MQSC komutu
- Kanal PCF komutunun durdurulması komutu
- IBM WebSphere MQ Gezgini

Bu komutları kullanarak kanalları durdurmak için kullanılacak üç seçenek vardır:

QUIES

QUIESCE seçeneği, kanalı durdurmadan önce yürürlükteki ileti kümesini sona erdirmeyi dener.

FORCE

FORCE seçeneği, kanalı hemen durdurmayı dener ve kanal belirsiz durumda bırakılabileceğinden, kanaldan yeniden eşzamanlanmasını gerektirebilir.

TERMINATE

TERMINATE seçeneği, kanalı hemen durdurmayı dener ve kanala ilişkin iş parçacığı ya da işlemi sonlandırmış olur.

Tüm bu seçenekler, kanalı DURDURULMUŞ durumda bırakır ve işletmen müdahalesini yeniden başlatmasını gerektirir.

Gönderme uçta kanal durduruluyorsa, ancak işletmenin yeniden başlatılması için işletmen müdahalesini gerektiren bir işlem vardır. Kanalin giriş sonunda, MCA gönderen taraftan veri beklediği için işler çok daha zor olur ve kanal *düzenli* 'i alma tarafından sona erdirmeyi başlatmanın bir yolu yoktur; durdurma komutu, MCA' nın veri beklemesinden geri dönünceye kadar beklemektedir.

Sonuç olarak, gereken işletim özelliklerine bağlı olarak, kanal kullanmanın önerilen üç yolu vardır:

- Kanallarınızın uzun süre çalışır durumda olmasını istiyorsanız, yalnızca gönderme bitiminden sonra düzenli olarak sona erdirilebileceğini unutmayın. Kanallar kesildiğinde, yeniden başlatmak için bu işlem durdurulur, işletmen müdahalesi (bir START KANAL komutu) gerekir.
- Kanallarınızın yalnızca iletilmeleri için iletiler olduğunda etkin olmasını istiyorsanız, bağlantı kesme aralığını oldukça düşük bir değere ayarlayın. Varsayılan ayar yüksektir ve bu nedenle bu denetim düzeyinin gerekli olduğu kanallar için önerilmez. alıcı kanalını kesmek zor olduğu için, en ekonomik seçenek, kanalın otomatik olarak kopması ve iş yükü talepleri olarak yeniden bağlanmasını sağlamış olması. Çoğu kanal için, bağlantı kesme aralığının uygun ayarı buluşsal olarak oluşturulabilir.
- Sinyal aralığı özneteliğini kullanarak, gönderme yapan MCA 'nın ileti göndermediği dönemler sırasında alıcı MCA' ya bir sinyal akışı göndermesine neden olur. Bu işlem, bekleme durumundan alınan MCA 'yı serbest bırakır ve bağlantı kesme aralığının süresinin dolması için beklemeden kanalı susturma olanağı sağlar. Sağlıklı işletim bildirim aralığını, bağlantı kesme aralığının değerinden daha düşük bir değer olarak verin.

Not:

1. Sunucu kanalları için, bağlantı kesme aralığını düşük bir değere ayarlamanız ya da kalp atışlarının kullanılmasını önermeniz önerilir. Bu düşük değer, sunucu kanalının göndereceği hiçbir ileti olmadığına, istekte bulunan kanalının olağandışı bittiği (örneğin kanal iptal edildiği için) olduğu durumlarda kullanılabilir. Bağlantı kesme aralığı üst ve kalp atışları kullanımda değilse, sunucu, istekte bulunanın sona erdiğini algılamaz (bu durumda, istekte bulunana bir sonraki ileti gönderme girişiminde bulunsun). Sunucu çalışmaya devam ederken, kuyrukta daha fazla ileti elde etmek için, dışlayıcı giriş için iletim kuyruğunu açık tutar. Kanalı yeniden başlatmak için istekte bulunmuyorsa, sunucu hala dışlayıcı giriş için iletim kuyruğu açık durumda olduğundan, başlatma isteği bir hata alır. Sunucu kanalını durdurmak ve daha sonra, yeniden istekte bulunandan kanalı yeniden başlatmanız gerekir.

Durdurulan kanallar yeniden başlatılıyor

Bir kanal DURDURULAN durumuna geçtiğinde, kanalı el ile yeniden başlatmanız gerekir.

Kanalı yeniden başlatmak için aşağıdaki komutlardan birini yayınlayın:

- START CHANNEL MQSC komutu
- Kanal PCF 'yi Başlat komutu
- IBM WebSphere MQ Gezgini

Gönderen ya da sunucu kanalları için, kanal DURDURULDU durumuna girdiğinde, ilişkili iletim kuyruğu GET (DISABLE) olarak ayarlandı ve tetikleme işlemi kapatıldı. Başlatma isteği alındığında, bu öznetelikler otomatik olarak sıfırlanır.

If the queue manager (on distributed platforms) stops while a channel is in RETRYING or STOPPED status, the channel status is remembered when the queue manager is restarted. Ancak, kanal durdurulmuş durumdayken kuyruk yöneticisi durdurulursa, SVRCONN kanal tipine ilişkin kanal durumu sıfırlanır.

Belirsiz kanallar

Belirsiz kanal kanalı, iletilerin gönderildiği ve alındığı uzak bir kanalla kuşku içinde olan bir kanaldır.

Bu ve bir kuyruk yöneticisi arasındaki farkın, hangi iletilerin bir kuyruğa gönderileceği konusunda kuşku göstermesine dikkat edin.

Bir kanalın, Batch Heartbeat kanal parametresini (BATCHEB) kullanarak kuşku konması için gereken fırsatı azaltabilirsiniz. Bu parametre için bir değer belirtildiğinde, başka bir işlem yapmadan önce, bir gönderen kanalı uzak kanalın hala etkin olduğunu denetler. Yanıt alınmazsa, alıcı kanalın artık etkin olmadığı kabul edilir. İletiler geriye işlenebilir ve yeniden yönlendirilebilir ve gönderen kanalı belirsiz bir şekilde yerleştirilemez. Bu, alıcının, alıcı kanalının hala etkin olduğunu doğrulayan gönderici kanalı arasındaki döneme belirsiz bir şekilde yerleştirilebileceği ve alıcı kanalının gönderilen iletileri aldığından emin olduğu süreyi kısaltır. Toplu iş bildirim parametresiyle ilgili ek bilgi için [Kanal öznitelikleri](#) başlıklı konuya bakın.

Belirsiz kanal sorunları genellikle otomatik olarak çözülür. İletişim kaybedildiğinde bile, bir kanal, teslim alma durumu bilinmediği için göndericide bir ileti grubuyla kuşku halinde yerleştirilirse, iletişim yeniden kurulduğunda durum çözülür. Sıra numarası ve LUWID kayıtları bu amaçla tutulurlar. Kanal, LUWID bilgileri değiş tokuş edilinceye kadar kuşku içinde olur ve kanal için yalnızca tek bir ileti kümesi şüpheye yol gösterilebilir.

Gerektiğinde, kanalı el ile yeniden eşitleyebilirsiniz. *el ile* terimi, WebSphere MQ sistem yönetimi komutlarını içeren işlemlerin ya da programların kullanılmasını içerir. El ile yeniden eşzamanlama işlemi aşağıdaki gibi çalışır. Bu açıklama MQSC komutlarını kullanır, ancak PCF eşdeğerlerini de kullanabilirsiniz.

1. Kanalın **her biri** tarafındaki son kesinleştirilecek mantıksal iş tanıtıcısı (LUWID) bulmak için DISPLAY CHSTATUS komutunu kullanın. Bunu yapmak için aşağıdaki komutları kullanın:

- Kanalın belirsiz tarafı için:

```
DISPLAY CHSTATUS(name) SAVED CURLUWID
```

Kanalı daha ayrıntılı tanımlamak için CONNAME ve XMITQ parametrelerini kullanabilirsiniz.

- Kanalın alıcı tarafı için:

```
DISPLAY CHSTATUS(name) SAVED LSTLUWID
```

Kanalı daha ayrıntılı tanımlamak için CONNAME parametresini kullanabilirsiniz.

Bu komutlar farklı olduğundan, kanalın yalnızca gönderme yapan tarafı belirsiz durumda olabilir. Giriş tarafı hiçbir zaman şüphe yok.

On WebSphere MQ for IBM i, the DISPLAY CHSTATUS command can be executed from a file using the STRMQMMQSC command or the Work with MQM Channel Status CL command, WRKMQMCHST

2. İki LUWID aynıysa, alan taraf, gönderenin şüpheye göre kabul ettiği iş birimini kesinleştirmiştir. Gönderme tarafı, artık gelen belirsiz iletileri iletim kuyruğundan kaldırabilir ve yeniden etkinleştirebilir. Bu, aşağıdaki kanal RESOLVE komutu ile yapılır:

```
RESOLVE CHANNEL(name) ACTION(COMMIT)
```

3. İki LUWID farklıysa, alan taraf, gönderenin şüphe içinde olduğunu düşündüğü iş birimini işlememiş demektir. Gönderme tarafının, iletim kuyruğunda belirsiz olan iletileri alıkoymasına ve yeniden göndermesi gerekir. Bu, aşağıdaki kanal RESOLVE komutu ile yapılır:

```
RESOLVE CHANNEL(name) ACTION(BACKOUT)
```

Bu işlem tamamlandığında, kanal artık belirsiz durumda olur. İletim kuyruğu, gerekliyse, başka bir kanal tarafından kullanılabilir.

Sorunların saptanması

Bir komut sunulduğunda keşfedilen ve kanalların çalışması sırasında ortaya çıkarılan sorunlar ortaya çıktı: sorunların saptanması için iki ayrı yöndür.

Komut geçerlilik denetimi

Komutlar ve pano verileri işlenmek üzere kabul edilmeden önce hatalardan kurtulmalıdır. Geçerlilik denetimi tarafından bulunan tüm hatalar, hata iletileri tarafından hemen kullanıcıya bildirilir.

Sorun tanılama, bu hata iletilerinin yorumlanması ve düzeltici işlem yapılması ile başlar.

Sorunların işlenmesi

Kanalların olağan çalışması sırasında bulunan sorunlar, sistem konsoluna ya da sistem günlüğüne bildirilir. Sorun tanılama, günlüğün ilgili tüm bilgileri derlemeyle başlar ve sorunu belirlemek için çözümlenmeye devam eder.

Olanaklı olduğunda, komutları başlatan uçbirime doğrulama ve hata iletileri döndürülür.

WebSphere MQ , kullanım ve performans alanında eğilimleri belirlemek için kullanabileceğiniz, muhasebe ve istatistiksel veriler üretir. **distributed** Dağıtılmış altyapılarda, bu bilgiler PCF kayıtları olarak üretilir, ayrıntılar için [Yapı veri tipleri](#) konusuna bakın.

İletiler ve kodlar

Sorunun birincil tanımıyla yardımcı olacak iletiler ve kodlar için bkz. [Tanılama iletileri ve neden kodları](#).

İletilerin güvenliği

Dağıtılmış kuyruk yönetimi, WebSphere MQ' un tipik kurtarma özelliklerine ek olarak, ileti kanalının iki ucu arasında eşgüdümü bir eşitleme noktası yordamı kullanarak iletilerin düzgün bir şekilde teslim edilmesini sağlar. Bu yordam bir hata saptarsa, kanalı kapatır, böylece sorunu araştırabilir ve kanal yeniden başlatılincaya kadar iletileri iletim kuyruğunda güvenli bir şekilde tutar.

Eşitleme noktası yordamı, kanal başlatıldığında bir *belirsiz* durumu kurtarmaya yönelik girişimlerde bir avantaja sahip olur. (*Kuş-belirsiz* , bir eşitleme noktasının istendiği, ancak isteğin sonucu henüz bilinmediği için kurtarma biriminin durumu.) Bu olanakla da ilişkili iki işlev de vardır:

1. Kesinleştirme ya da geri alma ile çözümler
2. Sıra numarasını ilk durumuna getir

Bu işlevlerin kullanımı yalnızca olağandışı durumlarda gerçekleşir çünkü kanal çoğu durumda otomatik olarak kurtarır.

Hızlı, kalıcı olmayan iletiler

Kanaldaki kalıcı olmayan iletilerin daha hızlı bir şekilde sağlanabileceğini belirtmek için kalıcı olmayan ileti hızı (NPMSPED) kanal özneteliği kullanılabilir. Bu özneteliğe ilişkin daha fazla bilgi için bkz. [Kalıcı olmayan ileti hızı \(NPMSPED\)](#).

Bir kanal hızlı bir şekilde sona erdirilirse, kalıcı olmayan iletiler geçiş durumunda olduğunda, iletiler kaybedilebilir ve gerekiyorsa, kurtarma işlemi için gerekli olan uygulamaya kadar olan iletiler kaybedilebilir.

Alma kanalı iletiyi hedef kuyruğuna koyamıyorsa, önceden tanımlanmış bir ileti varsa, bu ileti kuyruklanmış olan ileti kuyruğuna yerleştirilir. Yoksa, ileti atılır.

Not: Kanalın diğer ucu seçeneği desteklemiyorsa, kanal normal hızda çalışır.

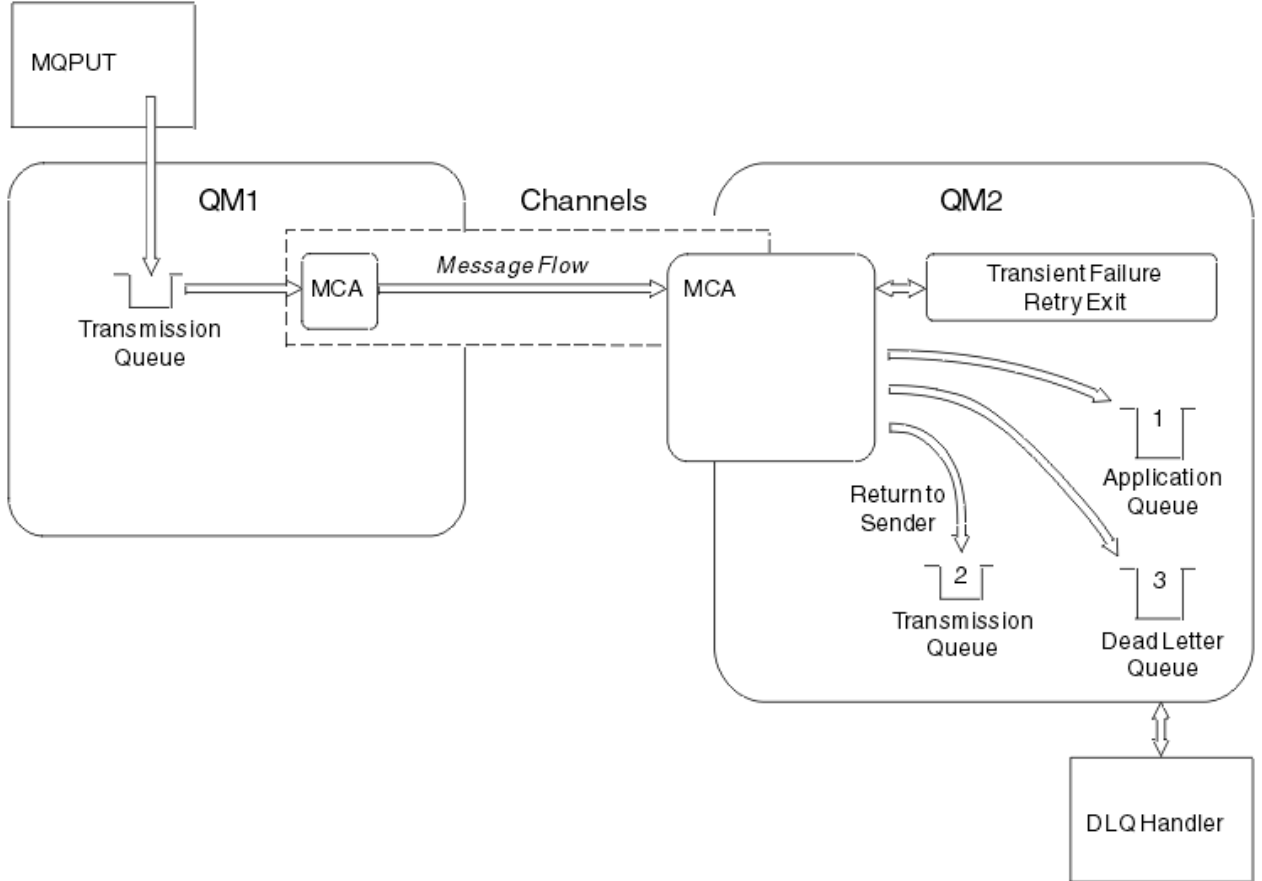
Teslim Edilmeyen İletiler

Bir ileti teslim edilemediğinde ne olacağı hakkında bilgi için bkz. [“Bir ileti teslim edilemezse ne olur?”](#) sayfa 65.

Bir ileti teslim edilemezse ne olur?

Bir ileti teslim edilemediğinde, MCA bunu birkaç şekilde işleyebilir. Yeniden deneyebilir, gönderene geri dönüşebilir ya da onu ölü harf kuyruğuna yerleştirebilir.

Şekil 15 sayfa 65 , MCA 'nın hedef kuyruğa bir ileti koyamadığında ortaya çıkan işleme gösterir. (Gösterilen seçenekler tüm altyapılar üzerinde geçerli değildir.)



Şekil 15. Bir ileti teslim edilemediğinde ne olur

Bu şekilde gösterildiği gibi, MCA 'nın sağlayamadığı bir iletiyle birçok şey yapabilir. Alınan işlem, kanal tanımlandığında ve ileti için MQPUT rapor seçeneklerinde belirtilen seçeneklerle belirlenir.

1. İleti-yeniden deneme

MCA, geçici bir nedenle (örneğin, kuyruk dolu olduğu için) bir nedenden ötürü hedef kuyruğa bir ileti koyamıyorsa, MCA bekleyebilir ve işlemi daha sonra yeniden deneyebilir. MCA 'nın ne kadar süreyle bekleyeceğini ve kaç kez denediğini saptayabilirsiniz.

- Kanalınızı tanımlarken, MQPUT hataları için bir ileti yeniden deneme süresi ve aralığı belirtebilirsiniz. İleti hedef kuyruğa konamıyorsa, kuyruk dolu olduğu için ya da bir yerleştirmeyi engelliyorsa, MCA işlemi, belirtilen zaman aralığında belirtilen zaman sayısını, MCA işlemi dener.
- Kendi iletinizi yazabilirsiniz-yeniden deneme çıkışıdır. Çıkış, MCA 'nın MQPUT ya da MQOPEN işlemini yeniden denemesini istediğiniz koşullar altında belirleyebilmenizi sağlar. Kanalı tanımlarken çıkışa ilişkin adı belirtin.

2. Gönderene dön

İleti yeniden deneme başarısız olduysa ya da farklı tipte bir hata saptandıysa, MCA iletiyi orijinaline geri gönderebilir. Gönderene geri dönmeyi etkinleştirmek için, iletiyi özgün kuyruğa koyduğunuzda, ileti tanımlayıcısında aşağıdaki seçenekleri belirtmeniz gerekir:

- MQRO_EXCEPTION_WITH_FULL_DATA rapor seçeneği
- MQRO_DISCARD_MSG rapor seçeneği
- Yanıtlama kuyruğu ve yanıtlama kuyruğu yöneticisinin adı

MCA iletiyi hedef kuyruğa koyamıyorsa, özgün iletiyi içeren bir kural dışı durum raporu oluşturur ve bunu, özgün iletide belirtilen yanıtlama kuyruğuna gönderilecek bir iletim kuyruğuna koyar. (Yanıtlama kuyruğu, MCA ile aynı kuyruk yöneticisiyse, ileti bir iletim kuyruğuna değil, doğrudan o kuyruğa konadır.)

3. Ölü-mektup kuyruğu

Bir ileti teslim edilemezse ya da döndürülmezse, DLQ (dead-letter queue; ölü-mektup kuyruğuna (DLQ) yerleştirilir). İletiyi işlemek için DLQ işleyicisini kullanabilirsiniz. Bu işleme, Teslim edilmeyen iletilerin WebSphere MQ ölme-harfli kuyruk işleyicisiyle işlenmesi for IBM WebSphere MQ for UNIX, Linux için açıklanmaktadır. Ölü-harf kuyruğu kullanılamıyorsa, MCA gönderen ileti iletim kuyruğunda iletiyi bırakır ve kanal durur. Hızlı bir kanalda, bir ölü-mektup kuyruğuna yazılamayan kalıcı olmayan iletiler kaybedilir.

IBM WebSphere MQ Version 7.0' da, yerel bir ölü-mektup kuyruğu tanımlanmadıysa, uzak kuyruk kullanılabilir ya da tanımlı değildir ve uzak bir ileti kuyruğu yoktur; böylece gönderen kanalı RETRY (yeniden deneme) içine girer ve iletiler otomatik olarak iletim kuyruğuna geri alınır.

İlgili başvurular

[Dead-Letter Queue \(USEDLQ\) kuyruğunu kullan](#)

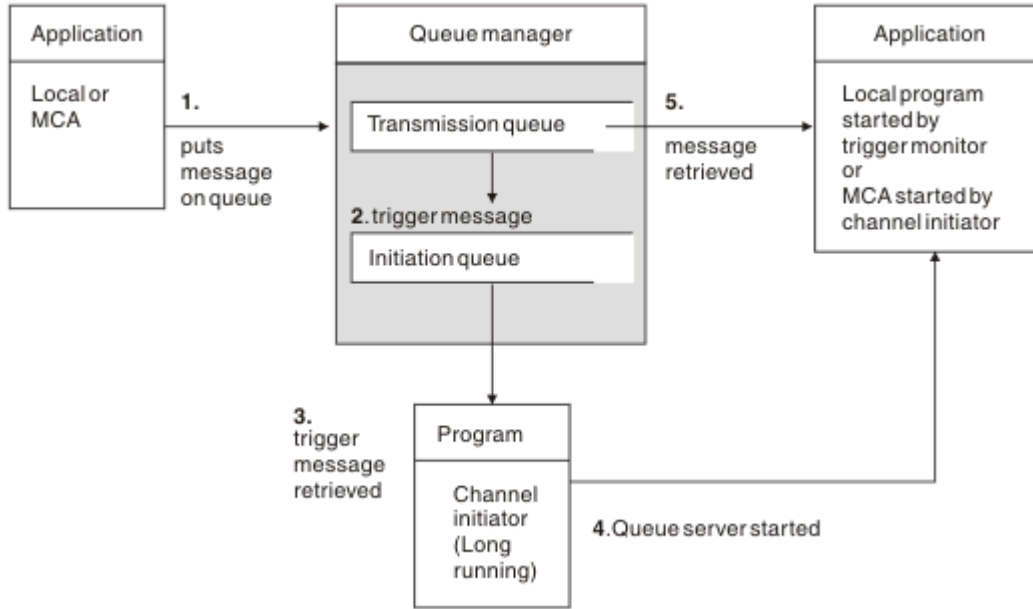
Tetikleme kanalları

WebSphere MQ , kuyruklardaki belirli koşullar karşılandığında otomatik olarak bir uygulama başlatmak için bir olanak sağlar. Bu tesise tetikleme deniyor.

Bu açıklama, tetikleme kavramlarına genel bakış olarak yönelmektedir. Eksiksiz bir açıklama için bkz. [Starting WebSphere MQ applications using triggers](#).

Platforma özgü bilgiler için aşağıdakine bakın:

- Windows için bkz. UNIX and Linux sistemleri, [“UNIX, Linux ve Windows sistemlerindeki tetikleme kanalları.”](#) sayfa 68



Şekil 16. Tetikleme kavramları

Tetikleme için gereken nesnelere Şekil 16 sayfa 67’de gösterilir. Bu, aşağıdaki olay sırasını gösterir:

1. Yerel kuyruk yöneticisi, bir uygulamadaki ya da ileti kanalı aracısından (MCA) iletim kuyruğunda bir ileti yerleştirir.
2. Tetikleme koşulları yerine getirildiğinde, yerel kuyruk yöneticisi başlatma kuyruğuna bir tetikleyici iletisi yerleştirir.
3. Uzun süredir çalışan kanal başlatıcı programı, başlatma kuyruğunu izler ve gelen iletileri alır.
4. Kanal başlatıcı, tetikleme iletilerini, bu iletilerin içerdiği bilgilere göre işler. Bu bilgiler, ilgili MCA 'nın başlatıldığı durumda kanal adını içerebilir.
5. Yerel uygulama ya da MCA tetiklenmiş olarak, ileti iletilerini iletim kuyruğundan alır.

Bu senaryoya ilişkin bilgi için aşağıdaki başlıklara gereksinim duyarsınız:

- İletim kuyruğunu, başlatma kuyruğu adıyla (SYSTEM.CHANNEL.INITQ) ilgili öznitelige karşılık gelir.
- Kullanıma hazırlama kuyruğunun (SYSTEM.CHANNEL.INITQ) bulunur.
- Kanal başlatıcı programının kullanılabilir durumda ve çalışır durumda olduğundan emin olun. Kanal başlatıcı programı, başlatma komutundaki başlatma kuyruğunun adıyla birlikte sağlanmalıdır.
- Optionally, create the process definition for the triggering, if it does not exist, and ensure that the *UserData* field contains the name of the channel it serves. Bir süreç tanımlaması yaratmak yerine, kanal adını iletim kuyruğunun *TriggerData* öznitelide belirtebilirsiniz. WebSphere MQ for UNIX, Linux ve Windows sistemleri, kanal adının boş olarak belirtilmesine izin verir; bu durumda, bu iletim kuyruğuna sahip ilk kullanılabilir kanal tanımlaması kullanılır.
- İletim kuyruğu tanımının, hizmet verecek süreç tanımlamasının adını (varsa), başlatma kuyruğu adını ve hissettirdiğiniz tetikleme özelliklerini en uygun şekilde içerdiğini doğrulayın. Tetikleme denetim özniteliği, gerektiğinde tetiklemeye izin verir ya da etkinleştirilmez.

Not:

1. Kanal başlatıcı programı, kanalları başlatmak için kullanılan başlatma kuyruğunu izleyen bir 'tetikleyici izleme programı' işlevi görür.
2. Bir başlatma kuyruğu ve tetikleme işlemi, kanal sayısını tetiklemek için kullanılabilir.
3. Herhangi bir sayıda kullanıma hazırlama kuyruğu ve tetikleyici işlemi tanımlanabilir.

4. Sistemin kanalla başlatılmasını önlemek için, FIRST tetikleme tipi önerilir.

UNIX, Linux ve Windows sistemlerindeki tetikleme kanalları.

Tetiklenecek süreçleri tanımlayarak WebSphere MQ' ta bir süreç tanımlaması yaratabilirsiniz. İletilerin iletim kuyruğuna vardığında tetiklenecek süreç tanımlamasını yaratmak için MQSC komutu DEFINE PROCESS komutunu kullanın. Süreç tanımlamasının USERDATA özneliği, iletim kuyruğu tarafından hizmet verilen kanalın adını içerir.

Yerel kuyruğu tanımlayın (QM4), tetikleme iletilerinin, kanalı başlatan uygulamayı (QM3.TO.QM4) tetiklemek için initiation queue (IQ) olanağına yazılacağı belirtilmesini sağla:

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(P1) USAGE(XMITQ)
```

Define the application (process P1) to be started:

```
DEFINE PROCESS(P1) USERDATA(QM3.TO.QM4)
```

Diğer bir seçenek olarak, WebSphere MQ for UNIX, Linux ve Pencerele sistemleri için, iletim kuyruğunun TRIGDATA öznelideki kanal adını belirleyerek bir süreç tanımlaması gereksinimini ortadan kaldıracaktır.

Yerel kuyruğu tanımlayın (QM4). Specify that trigger messages are to be written to the default initiation queue SYSTEM.CHANNEL.INITQ, to trigger the application (process P1) that starts channel (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ)  
USAGE(XMITQ) TRIGDATA(QM3.TO.QM4)
```

Bir kanal adı belirtmezseniz, kanal başlatıcısı, kanal tanımlama dosyalarını, adı belirtilen iletim kuyrukla ilişkili bir kanal buluncaya kadar arar.

İlgili kavramlar

[“Kanal başlatıcısının başlatılması ve durdurulması” sayfa 68](#)

Tetikleme, kanal başlatıcı işlemi kullanılarak gerçekleştirilir.

[“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27](#)

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eşitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluşları arasındaki iletişim ile ilgili daha ayrıntılı bilgi sağlanmaktadır.

İlgili başvurular

[UNIX, Linux ve Windows sistemlerinde kanal programları](#)

Kanal başlatıcısının başlatılması ve durdurulması

Tetikleme, kanal başlatıcı işlemi kullanılarak gerçekleştirilir.

Bu kanal başlatıcı işlemi, MQSC komutu START CHINIT ile başlatılır. Varsayılan başlatma kuyruğunu kullanmıyorsanız, komutta yer alma kuyruğunun adını belirtin. Örneğin, varsayılan kuyruk yöneticisi için kuyruk IQ ' u başlatmak üzere START CHINIT komutunu kullanmak için şunu girin:

```
START CHINIT INITQ(IQ)
```

Varsayılan olarak, kanal başlatıcısı varsayılan başlatma kuyruğu olan SYSTEM.CHANNEL.INITQ kullanılarak otomatik olarak başlatılır. Tüm kanal başlatıcılarınızı el ile başlatmak istiyorsanız, aşağıdaki adımları izleyin:

1. Kuyruk yöneticisini yaratın ve başlatın.
2. Kuyruk yöneticisinin SCHINIT özelliğini MANUAL olarak değiştirin.

3. Kuyruk yöneticisini sona erdirin ve yeniden başlatın

Linux ve Pencerelemler sistemlerinde, otomatik olarak bir kanal başlatıcısı başlatılır. Başlatabileceğiniz kanal başlatıcılarının sayısı sınırlıdır. Varsayılan değeri ve üst sınır değeri 3 'tür. Bunu, UNIX and Linux sistemleri için qm.ini dosyasındaki MAXINITIATOR ' ları kullanarak ve Windows sistemleri için kayıt defterinde değiştirebilirsiniz.

See [WebSphere MQ Control komutları](#) for details of the run channel initiator command **runmqchi**, and the other control commands.

Kanal başlatıcı durduruluyor

Varsayılan kanal başlatıcısı, kuyruk yöneticisi başlattığınızda otomatik olarak başlatılır. Bir kuyruk yöneticisi durdurulduğunda, tüm kanal başlatıcıları otomatik olarak durdurulur.

Başlatma ve yapılandırma dosyaları

Kanal kullanıma hazırlama verilerinin işlenmesi, WebSphere MQ platformunuza bağlıdır.

Windows, UNIX and Linux sistemleri

In WebSphere MQ for Windows, UNIX and Linux systems, there are *yapılandırma dosyaları* to hold basic configuration information about the WebSphere MQ installation.

İki yapılandırma dosyası vardır: biri makine için geçerli, diğeri ise tek bir kuyruk yöneticisi için geçerlidir.

WebSphere MQ yapılandırma dosyası

Bu dosya, WebSphere MQ sistemindeki tüm kuyruk yöneticilerine ilişkin bilgileri içerir. Dosyaya mq.ini adı verilir. It is fully described in the [Yönetme](#) for WebSphere MQ for Windows, UNIX and Linux systems.

Kuyruk yöneticisi yapılandırma dosyası

Bu dosya, belirli bir kuyruk yöneticisiyle ilgili yapılandırma bilgilerini içerir. Dosyaya qm.ini adı verilir.

Bu, kuyruk yöneticisi yaratma işlemi sırasında yaratılır ve kuyruk yöneticisinin herhangi bir yönüyle ilgili yapılandırma bilgilerini tutabilir. Dosyada tutulan bilgiler, günlükün yapılandırmasının WebSphere MQ yapılandırma dosyasındaki varsayılan yapılandırmalardan nasıl farklılık göstereceğini içerir.

Kuyruk yöneticisi yapılandırma kütüğü, kuyruk yöneticisi tarafından meşgul edilen izin ağacının kökünde tutulur. Örneğin, DefaultPath özniteliklerinde, QMNAME adlı bir kuyruk yöneticisine ilişkin kuyruk yöneticisi yapılandırma kütükleri aşağıdaki gibi olur:

UNIX and Linux sistemleri için:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

An excerpt of a qm.ini file follows. Bu, TCP/IP dinleyicisinin 2500 numaralı bağlantı noktasını dinlemek olduğunu, yürürlükteki kanal sayısı üst sınırının 200 olduğunu ve etkin kanal sayısı üst sınırının 100 olduğunu belirtir.

```
TCP:
  Port=2500
CHANNELS:
  MaxChannels=200
  MaxActiveChannels=100
```

Giden kanal tarafından kullanılacak bir TCP/IP kapısı aralığı belirleyebilirsiniz. Bir yöntem, bir kapı değerleri aralığının başlangıcını ve bitmesini belirtmek için qm.ini dosyasını kullanmaktadır. Aşağıdaki örnekte, kanal aralığını belirten bir qm.ini dosyası gösterilmektedir:

```
TCP:
  StrPort=2500
```

```
EndPort=3000
CHANNELS:
MaxChannels=200
MaxActiveChannels=100
```

StrPort ya da EndPort için bir değer belirtirseniz, her ikisi için de bir değer belirlemeniz gerekir. EndPort değerinin her zaman StrPortdeğerinden büyük olması gerekir.

Kanal, belirtilen aralıktaki kapı değerlerinin her birini kullanmaya çalışır. Bağlantı başarılı olduğunda, kapı değeri, kanalın kullandığı kapıdır.

Windows sistemleri için:

```
C:\Program Files\IBM\WebSphere MQ\qmgrs\QMNAME\qm.ini
```

qm.ini kütükleriyle ilgili ek bilgi için bkz. [Dağıtılmış kuyruğa alma için yapılandırma dosyası dayanakları.](#)

İletiler için veri dönüştürme

WebSphere MQ iletileri, farklı kuyruk yöneticilerindeki kuyruklar arasında gönderildiklerinde veri dönüştürmesi gerektirebilir.

Bir WebSphere MQ iletisi iki bölümden oluşur:

- Bir ileti tanımlayıcısında bilgileri denetleme
- Uygulama Verileri

İki parçadan biri, farklı kuyruk yöneticilerindeki kuyruklar arasında gönderildiğinde veri dönüştürme gerektirebilir. Uygulama verileri dönüştürme hakkında bilgi için bkz. [Uygulama verileri dönüşümü.](#)

Kendi ileti kanalı aracılarınızı yazma

WebSphere MQ , kendi Message Channel Agent (MCA) programlarınızı yazmanızı ya da bir bağımsız yazılım üreticisinden birini kurmanızı sağlar.

WebSphere MQ 'u kendi özel iletişim protokolünüz üzerinde birlikte kullanmak ya da WebSphere MQ 'un desteklemediği bir protokol üzerinden ileti göndermek için kendi MCA programlarınızı yazmak isteyebilirsiniz. (Bir WebSphere MQ ile birlikte sağlanan MCA 'yı diğer uçta kullanarak, kendi MCA 'yı yazmanız için yazamazsınız.)

WebSphere MQ tarafından sağlanmamış bir MCA 'yı kullanmaya karar verirseniz, aşağıdaki noktaları göz önünde bulundurmanız gerekir.

İleti gönderme ve alma

Uygulamanızın onları nereye yerleştirdiği, örneğin bir iletim kuyruğundan ileti alan ve iletişim kurmak istediğiniz bir iletişim kuralına gönderme yapan bir gönderme uygulaması yazmanız gerekir. Ayrıca, bu protokolden ileti alan ve hedef kuyruklara yerleştiren bir alma uygulaması yazmanız gerekir. Gönderme ve alma uygulamaları, herhangi bir özel arabirim değil, ileti kuyruğu arabirimini (MQI) çağırılıyor.

İletilerin yalnızca bir kez teslim edildiğinden emin olmanız gerekir. Bu teslimata yardımcı olmak için eşitleme noktası eşgüdümü kullanılabilir.

Kanal denetimi işlevi

Kanalları denetlemek için kendi denetim işlevlerinizi sağlamalısınız. You cannot use WebSphere MQ channel administration functions either for configuring (for example, the DEFINE CHANNEL command) or monitoring (for example, DISPLAY CHSTATUS) your channels.

kullanıma hazırlama dosyası

Gerekliyse, kendi kullanıma hazırlama dosyanızı belirtmeniz gerekir.

Uygulama verileri dönüştürme

Büyük olasılıkla farklı bir sisteme gönderdiğiniz iletiler için veri dönüştürme işlemi için izin vermek isteyebilirsiniz. Bu durumda, uygulamanızın bulunduğu her yerden ileti alınırken MQGET çağrısındaki MQGMO_CONVERT seçeneğini kullanın; örneğin, iletim kuyruğu.

Kullanıcı çıkışları

Kullanıcı çıkışlarına gerek olup olmadığını göz önünde bulundurun. Böyle bir durumda, WebSphere MQ tarafından kullanılan arabirim tanımlarını kullanabilirsiniz.

Tetikleme

Uygulamanız bir iletim kuyruğuna ileti koyarsa, iletim kuyruğu özniteliklerini ayarlayabilirsiniz; böylece iletiler kuyruğa ulaştığında gönderdiğiniz MCA 'nın tetiklenmesi tetiklenir.

Kanal başlatıcı

Kendi kanal başlatıcınızı sağlamanız gerekebilir.

Dağıtılmış kuyruk yönetimi için dikkate alınması gereken diğer konular

Dağıtılmış kuyruk yönetimi için WebSphere MQ 'u hazırlarken göz önünde bulundurulacak diğer konular. Bu konu, teslim edilemeyen ileti kuyruğunu, Kullanımda kuyrukları, sistem uzantılarını ve kullanıcı çıkışı programlarını ve çalışmakta olan kanalları ve dinleyicileri güvenilir uygulamalar olarak kapsar.

Teslim edilemeyen ileti kuyruğu

Teslim edilmemiş ileti kuyruğuna (ölü harf kuyruğu ya da DLQ olarak da bilinir) gelen iletilerin işlenmesini sağlamak için, bu iletileri işlemek üzere tetiklenebilecek ya da düzenli aralıklarla çalıştırılabilecek bir program yaratın. UNIX and Linux sistemlerinde WebSphere MQ ile bir DLQ işleyicisi sağlanır; daha fazla bilgi için [Örnek DLQ işleyicisi, amqsdllqbaşlıklı konuya](#) bakın.

Kullanımda olan kuyruklar

Alıcı kanallarına ilişkin MCA 'lar, iletiler iletilmediklerinde bile hedef kuyrukları açık tutabilirler. This results in the queues appearing to be "kullanımda".

Kanal sayısı üst sınırı

Bkz. [Dağıtılmış kuyruğa alma için yapılandırma dosyası stanzaları](#) .

Sistem uzantıları ve kullanıcı çıkışı programları

Kanal tanımlamasında, iletilerin işlenmesi sırasında ek programların tanımlanmış zamanlarda çalıştırılabilmesini sağlamak için bir tesis sağlanır. Bu programlar WebSphere MQ ile sağlanmaz, ancak her kuruluş yerel gereksinimlere göre sağlanabilir.

Çalıştırmak için, bu kullanıcı çıkışı programları önceden tanımlanmış adlara sahip olmalı ve kanal programlarına çağrı için kullanılabilir olmalıdır. İleti kanalı tanımlamalarında, kullanıcı çıkışı programlarının adları yer alır.

Bu programlara denetimi devretmek ve bu programlardan denetim döndürülmesini işlemek için tanımlanmış bir denetim öbeği arabirimi vardır.

Bu programların çağrıldığı kesin yerler ve denetim öbeklerinin ve adlarının ayrıntıları, [ileti alışverişi kanallarına ilişkin kanal çıkışı programları](#) 'nda bulunur.

Kanalların ve dinleyicilerin güvenilir uygulamalar olarak çalıştırılması

Ortamınızda performans önemli bir önem gösteriyorsa ve ortamınız sabit ise, FACTPATH bağıni kullanarak kanallarınızı ve dinleyicilerinizi güvenilir olarak çalıştırabilirsiniz. Kanalların ve dinleyicilerin güvenilir olarak çalıştırılıp çalıştırılmayacağını etkileyen iki faktör vardır:

- Ortam değişkeni MQ_CONNECT_TYPE=FASTPATH ya da MQ_CONNECT_TYPE = STANDARD. Bu büyük ve küçük harfe duyarlıdır. Geçerli olmayan bir değer belirtirseniz, bu değer yoksayılr.
- MQIBindType in the Channels stanza of the qm. ini or registry file. Bunu FASTPATH ya da STANDARD olarak ayarlayabilirsiniz ve büyük/küçük harfe duyarlı değildir. Varsayılan değer STANDARD 'tır.

Gerekli etkiyi gerçekleştirmek için ortam değişkeniyle ilişkili olarak MQIBindType ögesini kullanarak aşağıdaki gibi kullanılabilir:

MQIBindType	Ortam deęiřkeni	Sonu
Standart	Tanımlı deęil	Standart
FastPath	Tanımlı deęil	FastPath
Standart	Standart	Standart
FastPath	Standart	Standart
Standart	FastPath	Standart
FastPath	FastPath	FastPath
Standart	CLIENT	CLIENT
FastPath	CLIENT	Standart
Standart	LOCAL	Standart
FastPath	LOCAL	Standart

Özet olarak, kanalları ve dinleyicileri güvenilir olarak alıřtırmanın yalnızca iki yolu vardır:

1. qm.ini ya da kayıt defterinde MQIBindType= FASTPATH belirtilerek ve ortam deęiřkenini belirtmeyerek.
2. qm.ini ya da kayıt dosyasına MQIBindType= FASTPATH belirterek ve ortam deęiřkenini FASTPATH deęerine ayarlayın.

İletifimciler duraęan sreler olduęundan, alıřan dinleyicileri gvenilen olarak kullanmayı dřnn. Kararsız kanal ıkıřları ya da STOP CHANNEL MODE (TERMINATE) komutunu kullanmadıęınız srece, alıřan kanalları güvenilir olarak kullanmayı dřnn.

UNIX, Linux, and Windowszerindeki kanalların izlenmesi ve denetlenmesi

DQM iin, kanalları uzak kuyruk yneticilerine yaratmanız, izlemeniz ve denetlemeniz gerekir. Komutları, programları, IBM WebSphere MQ Explorerprogramlarını, kanal tanımlarına iliřkin dosyaları ve eřitleme bilgileri iin bir depolama alanını kullanarak kanalları denetleyebilirsiniz.

Ařaęıdaki komut tiplerini kullanabilirsiniz:

IBM WebSphere MQ komutları (MQSC)

MQSC 'yi, Windows, UNIX and Linux sistemlerinde bir MQSC oturumunda tek komut olarak kullanabilirsiniz. Daha karmařık ya da birden ok komut vermek iin, MQSC komutu, komut satırından alıřtırdıęınız bir dosyaya oluřturulabilir. Ayrıntılar iin bkz. [MQSC komutları](#). Bu blmde, daęıtılmıř kuyruklama iin MQSC 'nin kullanılmasına iliřkin bazı basit rnekler verilmektedir.

Kanal komutları, IBM WebSphere MQ Komutlarının (MQSC) bir alt kmeleridir. MQSC 'yi ve denetim komutlarını ařaęıdaki řekilde kullanıyorsunuz:

- Kanal Tanımlamaları Yarat, Kopyala, Grntle, Deęiřtir ve Sil
- Kanalları bařlatma ve durdurma, ping, kanal sıra numaralarını ilk durumuna getirme ve baęlantılar yeniden kurulamadıęında belirsiz iletiler zmler
- Kanallarla ilgili durum bilgilerini grntle

Denetim komutları

Bu iřlevlerden bazıları iin komut satırında *denetim komutları* de yayınlayabilirsiniz. Ayrıntılar iin bkz. [denetim komutları](#).

Programlanabilir komut biimi komutları

Ayrıntılar iin bkz. [PCF komutları](#).

IBM WebSphere MQ Explorer

UNIX, Linux ve Windows sistemlerinde, IBM WebSphere MQ Explorer' ı kullanabilirsiniz. Bu, denetim komutlarını ya da MQSC komutlarını kullanmaya alternatif olarak yönetim görevlerini yerine getirmek için bir grafik denetim arabirimi sağlar. Kanal tanımlamaları kuyruk yöneticisi nesnelere olarak tutulur.

Her kuyruk yöneticisinin, birbiriyle uyumlu uzak kuyruk yöneticilerine yönelik bağlantıları denetlemek için bir DQM bileşeni vardır. Bir depolama alanı sıra numaralarını ve *mantıksal iş birimi (LUW)* tanıtıcılarını içerir. Bunlar, kanal eşitleme amaçları için kullanılır.

Farklı komut tiplerini kullanarak ileti kanallarını kurarken ve denetlenirken kullanabileceğiniz işlevlerin bir listesi için [Çizelge 8 sayfa 74'](#) e bakın.

İlgili kavramlar

[“Nesnelerle çalışmaya başlama” sayfa 75](#)

Kanallar tanımlanmalıdır ve bir kanal başlatılmadan önce, bunların ilişkili nesnelere var olmalı ve kullanılabilir durumda olmalıdır. Bu bölüm size nasıl olduğunu gösterir.

[“Windows için iletişimi ayarlama” sayfa 82](#)

Dağıtılmış bir kuyruğa alma yönetimi kanalı başlatıldığında, kanal tanımında belirlenen bağlantıyı kullanmaya çalışır. Bunun başarılı olması için, bağlantının tanımlanabilmesini ve kullanılabilir olmasını sağlamak gerekir. Bu bölümde, WebSphere systemsfor Pencerelemler sistemleri için communicationMQ iletişim formlarından birini kullanarak bu işlem nasıl yapılır açıklanır.

[“UNIX and Linux sistemleri üzerinde iletişim kurma” sayfa 91](#)

DQM, IBM WebSphere MQ için uzak bir kuyruğa alma tesisidir. Sistem işletmeni tarafından denetlenir taban nesne iletişim bağlantılarına arabirim oluşturan kuyruk yöneticisi için kanal denetim programları sağlar. Dağıtılmış kuyrukla yönetimi tarafından tutulan kanal tanımları bu bağlantıları kullanır.

İlgili başvurular

[UNIX, Linux ve Windows sistemlerinde kanal programları](#)

[Dağıtılmış platformlar için ileti kanalı planlama örneği](#)

[Örnek yapılandırma bilgileri](#)

[Kanal öznelikleri](#)

Kanalları kurmak ve denetlemek için gereken işlevler

Kanalların ayarlanması ve denetim kanallarının denetlenmesi için IBM WebSphere MQ işlevlerine gerek vardır. Kanal işlevleri bu konuda açıklanabilir.

You can create a channel definition using the default values supplied by IBM WebSphere MQ, specifying the name of the channel, the type of channel you are creating, the communication method to be used, the transmission queue name and the connection name.

Kanal adı, kanalın her iki ucunda da aynı olmalı ve ağ içinde benzersiz olmalıdır. Ancak, kullanılan karakterleri IBM WebSphere MQ nesne adları için geçerli olan karakterler için sınırlamanız gerekir.

Diğer kanalla ilgili işlevler için aşağıdaki konulara bakın:

- [“Nesnelerle çalışmaya başlama” sayfa 75](#)
- [“İlişkili nesnelere yaratılıyor” sayfa 76](#)
- [“Varsayılan nesnelere yaratılıyor” sayfa 76](#)
- [“Kanal yaratılması” sayfa 76](#)
- [“Kanal görüntüleme” sayfa 77](#)
- [“Kanal durumunun görüntülenmesi” sayfa 77](#)
- [“Ping Komutu Kullanılarak Bağlantıların Denetlenmesi” sayfa 78](#)
- [“Kanal başlatma” sayfa 78](#)
- [“Kanalı durdurma” sayfa 79](#)
- [“Kanalı yeniden adlandırma” sayfa 80](#)
- [“Bir kanalı ilk durumuna getirme” sayfa 81](#)

- “Bir kanaldaki belirsiz iletiler çözümleniyor” sayfa 81

Çizelge 8 sayfa 74 , gereksinim duyabilirsiniz IBM WebSphere MQ işlevlerinin tam listesini gösterir.

Çizelge 8. UNIX, Linux, and Windows sistemlerinde gereken işlevler			
İşlev	Denetim komutları	MQSC	WebSphere MQ Explorer eşdeğeri mi?
Kuyruk yöneticisi işlevleri			
Değişiklik kuyruğu yöneticisi		ALTER QMGR	Evet
Kuyruk yöneticisi yarat	crtmqm		Evet
Kuyruk yöneticisini sil	dlmqm		Evet
Görüntü kuyruğu yöneticisi		QMGR GÖRÜNTÜLE	Evet
Kuyruk yöneticisini sona erdir	endmqm		Evet
Ping kuyruğu yöneticisi		QMGR PING.	Hayır
Kuyruk yöneticisini başlat	strmqm		Evet
Komut sunucusu işlevleri			
Komut sunucusunu görüntüle	dspmqcsv		Hayır
Komut sunucusunu sona erdir	endmqcsv		Hayır
Komut sunucusunu başlat	strmqcsv		Hayır
Kuyruk işlevleri			
Değişiklik kuyruğu		ALTER QALIAS ALTER QLOCAL ALTER QMODEL ALTER QREMOTE Bkz. ALTER kuyrukları .	Evet
Kuyruğu temizle		QLOCAL ' I TEMIZLE	Evet
Kuyruk yarat		QALIAS TANIMLANMASI QLOCAL TANIMLAMA Bkz. DEFINE kuyrukları .	Evet
Kuyruğu sil		SIL QALIAS DELETE QLOCAL DELETE QMODEL DELETE QREMOTE Bkz. DELETE kuyrukları .	Evet
Görüntü kuyruğu		GÖRÜNTÜLE KUYRUK	Evet
işlem işlevleri			

Çizelge 8. UNIX, Linux, and Windows sistemlerinde gereken işlevler (devamı var)

İşlev	Denetim komutları	MQSC	WebSphere MQ Explorer eşdeğeri mi?
Süreci değiştir		<u>ALTER PROCESS</u>	Evet
Süreç yarat		<u>SÜREÇ TANIMLA</u>	Evet
Süreci Sil		<u>Süreci Sil</u>	Evet
İşlemi görüntüle		<u>GÖRÜNEN SÜREÇ</u>	Evet
Kanal işlevleri			
Kanalı değiştir		<u>ALTER KANALI</u>	Evet
Kanal yarat		<u>KANAL TANIMI</u>	Evet
Kanalı sil		<u>KANALI SIL</u>	Evet
Görüntü kanalı		<u>KANAL GÖRÜNTÜLE</u>	Evet
Kanal durumunu görüntüle		<u>CHSTATU GÖRÜNTÜLE</u>	Evet
Bitiş kanalı		<u>KANALI DURDUR</u>	Evet
Ping kanalı		<u>PING KANALI</u>	Evet
Kanalı sıfırla		<u>KANAL ILK DURUMUNA GETİR</u>	Evet
Kanalı Çözümle		<u>KANAL çöZüMü</u>	Evet
Kanalı çalıştır	<u>runmqchl</u>	<u>KANAL BAŞLAT</u>	Evet
Kanal başlatıcıyı çalıştır	<u>runmqchi</u>	<u>CHINIT</u>	Hayır
Çalıştırma dinleyicisi ¹	<u>runmqlsr</u>	<u>DINLEYICI BAŞLAT</u>	Hayır
Bitiş dinleyicisi	<u>endmqlsr</u> (Windows sistemleri, AIX, HP-UX ve Solaris only)		Hayır
Not:			
1. Kuyruk yöneticisi başlatıldığında bir dinleyici otomatik olarak başlatılabilir.			

Nesnelerle çalışmaya başlama

Kanallar tanımlanmalıdır ve bir kanal başlatılmadan önce, bunların ilişkili nesnelere var olmalı ve kullanılabilir durumda olmalıdır. Bu bölüm size nasıl olduğunu gösterir.

Use the WebSphere MQ commands (MQSC) or the IBM WebSphere MQ Explorer to:

1. İleti kanallarının ve ilişkili nesnelere tanımlanması
2. İleti kanallarını izleme ve denetleme

Tanımlamanız gereken ilişkili nesnelere şunlardır:

- İletim kuyrukları
- Uzak kuyruk tanımlamaları
- Kuyruk yöneticisi diğer ad tanımlamaları
- Yanıtlama kuyruğu diğer ad tanımlamaları

- Yanıt-yerel kuyruklar
- Tetikleme işlemleri (MCA ' lar)
- İleti kanalı tanımlamaları

Her kanal için belirli bir iletişim bağlantısı tanımlanmalıdır ve kanal çalıştırılmadan önce kullanılabilir. LU 6.2, TCP/IP, NetBIOS, SPX ve DECnet bağlantılarının tanımlarına ilişkin açıklamalar için, kuruluşunuza ilişkin özel iletişim kılavuzuna bakın. Ayrıca bkz. [Örnek yapılandırma bilgileri](#).

Nesne yaratma ve nesnelere çalışma hakkında daha fazla bilgi için aşağıdaki alt başlıklara bakın:

İlişkili nesnelere yaratılıyor

MQSC, ilişkili nesnelere yaratmak için kullanılır.

Kuyruğu ve diğer ad nesnelere yaratmak için MQSC ' yi kullanın: iletim kuyrukları, uzak kuyruk tanımlamaları, kuyruk yöneticisi diğer ad tanımlamaları, yanıt kuyruğu diğer ad tanımlamaları ve yanıtlanma-yerel kuyruklar.

Ayrıca, benzer bir şekilde tetikleme (MCA ' lar) işlemlerinin tanımlamalarını da yaratın.

Tüm gerekli nesnelere nasıl oluşturulacağını gösteren bir örnek için bkz. [Dağıtılmış altyapılar için ileti kanalı planlama örneği](#).

Varsayılan nesnelere yaratılıyor

Varsayılan nesnelere, bir kuyruk yöneticisi yaratıldığında otomatik olarak yaratılır. Bu nesnelere, kuyruklar, kanallar, bir süreç tanımlaması ve denetim kuyruklarıdır. Varsayılan nesnelere yaratıldıktan sonra, strmqm komutunu -c seçeneğiyle çalıştırarak istediğiniz zaman bunları değiştirebilirsiniz.

crtmqm komutunu bir kuyruk yöneticisi yaratmak için kullandığınızda, komut bir varsayılan nesne kümesi yaratmak için de bir program başlatır.

1. Her bir varsayılan nesne sırayla oluşturulur. Program, kaç nesnenin başarıyla tanımlandığını, kaç nesnenin var olduğunu ve değiştirildiğini ve kaç başarısız girişimde bulunmayı sürdüreceğini sürekli olarak saklar.
2. Program, sonuçları size görüntüler ve herhangi bir hata oluştuysa, ayrıntılar için sizi uygun hata günlüğüne yönlendirir.

Program çalışmayı bitirdiğinde, kuyruk yöneticisini başlatmak için strmqm komutunu kullanabilirsiniz.

Crtmqm ve strmqm komutlarına ilişkin ek bilgi için [Denetim komutları](#) konusuna bakın.

Varsayılan nesnelere değiştirilmesi

-c seçeneğini belirttiğinizde, nesnelere oluşturulurken kuyruk yöneticisi geçici olarak başlatılır ve daha sonra, yeniden kapanır. -c seçeneğiyle strmqm komutu verilirse, varsayılan değerler var olan sistem nesnelere yeniler (örneğin, bir kanal tanımlamasının MCAUSER özniteliği boşluk olarak ayarlanmış olur). Kuyruk yöneticisini başlatmak istiyorsanız, strmqm komutunu -c seçeneği olmadan yeniden kullanmanız gerekir.

Varsayılan nesnelere değiştirmek isterseniz, eski amqscoma.tst dosyasının kendi sürümünü yaratabilir ve dosyayı düzenleyebilirsiniz.

Kanal yaratılması

Bağlantının her bir ucunda bir tane olmak üzere **iki** kanal tanımlaması yaratın. İlk kanal tanımlamasını ilk kuyruk yöneticisinde yaratıyorsunuz. Daha sonra, ikinci kuyruk yöneticisinde ikinci kanal tanımlamasını da, bağlantının diğer ucunda oluşturursun.

Her iki uç da **aynı** kanal adı kullanılarak tanımlanmalıdır. İki ucun **uyumlu** kanal tipleri olması gerekir; örneğin: Gönderen ve Alıcı.

Bağlantının bir ucu için bir kanal tanımlaması yaratmak üzere MQSC komutu TANILAT KANALINI KULLANIN. Kanalın adını, bağlantının bu ucuna ilişkin kanal tipini, bağlantı adını, tanımlamasını (gerekliyse), iletim kuyruğunun adını (gerekliyse) ve iletim protokolünü de ekleyin. Ayrıca, gerekli kanal

tipi için sistem varsayılan değerlerinden farklı olmasını istediğiniz diğer öznitelikleri de ekleyin; daha önce topladığınız bilgileri kullanın.

Kanal öznitelikleri' taki kanal özniteliklerinin değerlerine karar verme konusunda size yardımcı olarak sağlandınız.

Not: Ağınızdaki tüm kanalların benzersiz olarak adlanmanız önerilir. Kanal adında kaynak ve hedef kuyruk yöneticisi adları da içinde olmak üzere bunu yapmak için iyi bir yöntemdir.

Kanal örneği oluştur

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) +  
DESCR('Sender channel to QM2') +  
CONNNAME(QM2) TRPTYPE(TCP) XMITQ(QM2) CONVERT(YES)
```

MQSC ' nin tüm örneklerinde, komut bir komut dosyasında görüldüğü şekilde gösterilir ve Windows ya da UNIX ya da Linux sistemlerinde yazıldığı gibi. Etkileşimli olarak bir komut yayınlaması dışında, iki yöntem aynı görünür; önce bir MQSC oturumu başlatmalısınız. Type `runmqsc, for the default queue manager, or runmqsc qmname` where `qmname` is the name of the required queue manager. Daha sonra, örneklerde gösterildiği gibi, herhangi bir sayıda komut yazın.

Taşınabilirlik için, komutlarınızın satır uzunluğunu 72 karaktere sınırlayın. Use the concatenation character, +, as shown to continue over more than one line. Pencereler üzerinde, komut satırındaki girişi sonlamak için Ctrl-z tuşlarını kullanın. UNIX and Linux sistemlerinde, Ctrl-d tuşlarını kullanın. Diğer bir seçenek olarak, UNIX, Linux ya da Windows sistemlerinde **end** komutunu kullanın.

Kanal görüntüleme

Bir kanala ilişkin öznitelikleri görüntülemek için MQSC komutu `DISPLAY CHANNEL ' '` ı kullanın.

Belirli bir öznitelik istenmediyse ve belirtilen kanal adı soysal değilse, varsayılan olarak `DISPLAY CHANNEL` komutunun ALL deęiştirgesi varsayılan olarak kabul edilir.

Öznitelikler, Kanal özniteliklerinde açıklanmıştır.

Kanal örneklerini görüntüle

```
DISPLAY CHANNEL(QM1.TO.QM2) TRPTYPE,CONVERT  
DISPLAY CHANNEL(QM1.TO.*) TRPTYPE,CONVERT  
DISPLAY CHANNEL(*) TRPTYPE,CONVERT  
DISPLAY CHANNEL(QM1.TO.QMR34) ALL
```

Kanal durumunun görüntülenmesi

Kanal adını ve kanalların geçerli durumunu ya da kaydedilmiş bilgilerin durumunu belirterek, MQSC komutu `DISPLAY CHSTATUS` deęerini kullanın.

`DISPLAY CHSTATUS`, tüm ileti kanalları için geçerlidir. Sunucu bağlantısı kanalları dışındaki MQI kanalları için geçerli deęildir.

Görüntülenen bilgiler şunları içerir:

- Kanal adı
- İletişim bağlantısı adı
- Kanal durumunun belirsiz durumu (uygun olduğunda)
- Son sıra numarası
- İletim kuyruğu adı (uygun olduğunda)
- Belirsiz tanıtıcı (uygun olduğunda)
- Son kesinleştirilen sıra numarası

- Mantıksal iş birimi tanıtıcısı
- Süreç Tanıtıcısı
- İş Parçacığı Tanıtıcısı (yalnızca Windows)

Kanal durumu örneklerini görüntüle

```
DISPLAY CHSTATUS(*) CURRENT
DISPLAY CHSTATUS(QM1.TO.*) SAVED
```

Kanalda en az bir ileti kümesi iletilinceye kadar, kaydedilen durum geçerli değildir. Bir kanal durdurulduğunda (STOP CHL komutu kullanılarak) ve kuyruk yöneticisi sona erdirildiğinde durum da kaydedilir.

Ping Komutu Kullanılarak Bağlantıların Denetlenmesi

Uzak uçla birlikte sabit bir veri iletilmesi için MQSC komutu PING KANALINI kullanın.

Ping, bağlantının kullanılabilir olduğunu ve işleyeceğini sistem denetmenine güven sağlar.

Ping, iletim kuyruklarının ve hedef kuyrukların kullanılmasını içermez. Kanal tanımlarını, ilgili iletişim bağlantısını ve ağ ayarlarını kullanır. Yalnızca kanal etkin durumda değilse kullanılabilir.

Yalnızca göndericiden ve sunucu kanallarından kullanılabilir. İlgili kanal bağlantının uzak tarafında başlatılır ve başlatma parametresi kararlaşımı gerçekleştirir. Hatalar olağan bir şekilde bildirilir.

İleti değiş tokasının sonucu Ping complete ya da bir hata iletilmesi olarak sunulur.

LU 6.2 ile ping komutu gönder

Ping komutu çağrıldığında, varsayılan olarak hiçbir kullanıcı kimliği ya da parola alma uçlarına akar. Kullanıcı kimliği ve parola gerekiyorsa, bunlar kanal tanımlamasındaki başlangıç ucunda yaratılabilir. Kanal tanımına bir parola girilirse, bu parola kaydedilmeden önce WebSphere MQ tarafından şifrelenir. Daha sonra, sohbetin karşısına geçmeden önce şifresi çözülüyor.

Kanal başlatma

Gönderen, sunucu ve istekte bulunan kanallarına ilişkin MQSC komutu START kanalını kullanın. Uygulamalar için ileti alışverişi yapabilmemiz için, gelen bağlantılar için bir dinleyici programı başlatmanız gerekir.

START CHANNEL, kuyruk yöneticisi tetiklemesi ile bir kanalın ayarlandığı yerde gerekli değildir.

Başlatma sırasında, gönderen MCA kanal tanımlamalarını okur ve iletim kuyruğunu açar. Alıcı ya da sunucu kanalının karşılık gelen MCA ' larını uzaktan başlatan kanal başlatma sırası yayınlandı. Bunlar başlatıldığında, gönderici ve sunucu işlemleri, iletim kuyruğuna gelen iletileri bekler ve bunları ulaştıkları şekilde iletir.

Tetikleme ya da çalıştırma kanallarını iş parçacıkları olarak kullandığınızda, başlatma kuyruğunu izlemek için kanal başlatıcının kullanılabilir olduğundan emin olun. Kanal başlatıcı, kuyruk yöneticisinin bir parçası olarak varsayılan olarak başlatılır.

Ancak, TCP ve LU 6.2 diğer yetenekleri sağlar:

- UNIX and Linux sistemlerinde TCP için, inetd bir kanal başlatmak üzere yapılandırılabilir. inetd ayrı bir süreç olarak başladı.
- UNIX and Linux sistemlerinde LU 6.2 için, SNA ürününüzü LU 6.2 yanıtlayıcı işlemini başlatmak üzere yapılandırın.
- Windows sistemlerinde LU 6.2 için, SNA Server ' ı kullanarak kanal başlatmak için TpStart komutunu (SNA Server ile birlikte sağlanan bir yardımcı program) kullanabilirsiniz. TpStart , ayrı bir işlem olarak başlatıldı.

Başlatma seçeneğinin kullanılması, her zaman kanalın yeniden eşzamanlanmasına neden olur; bu durumda, gerekli yerlerde kanal yeniden eşzamanlanması gerekir.

Başarılı olmak için başlangıç için:

- Kanal tanımları, yerel ve uzak, var olmalıdır. Bir alıcı ya da sunucu bağlantısı kanalı için uygun kanal tanımlaması yoksa, kanal otomatik olarak tanımlıysa, varsayılan bir varsayılan değer otomatik olarak yaratılır. Bkz. [Channel auto-definition exit programı](#).
- İletim kuyruğu var olmalı ve bu kuyruğu kullanan başka kanalların olmaması gerekir.
- MCA ' lar, yerel ve uzak, var olmalıdır.
- İletişim bağlantısı kullanılabilir olmalıdır.
- Kuyruk yöneticilerinin çalışıyor, yerel ve uzak olması gerekir.
- İleti kanalı zaten çalışır durumda olmamalıdır.

Ekrana bir kanal başlatma isteğinin kabul edildiğini onaylayan bir ileti yayınlanır. Başlatma komutunun başarılı olduğunu doğrulamak için, hata günlüğünü denetleyin ya da DISPLAY CHSTATUS seçeneğini kullanın. Hata günlükleri şunlardır:

Windows

`MQ_INSTALLATION_PATH\qmgrs\qmname\errors\AMQERR01.LOG` (qmnameadlı her kuyruk yöneticisi için)

`MQ_INSTALLATION_PATH\qmgrs\@SYSTEM\errors\AMQERR01.LOG` (genel hatalar için)

`MQ_INSTALLATION_PATH` , WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder.

Not: Windows sistemlerinde, yine de Windows sistem uygulaması olay günlüğüne bir ileti de alabilirsiniz.

UNIX and Linux sistemleri

`/var/mqm/qmgrs/qmname/errors/AMQERR01.LOG` (qmnameadlı her kuyruk yöneticisi için)

`/var/mqm/qmgrs/@SYSTEM/errors/AMQERR01.LOG` (genel hatalar için)

On Pencereler, UNIX and Linux systems, use the runmqslsr command to start the WebSphere MQ listener process. Varsayılan olarak, kanal bağlantısı için gelen istekler, dinleyici işleminin MCA ' ları amqrmppa sürecinin iş parçacıkları olarak başlatmasına neden olur.

```
runmqslsr -t tcp -m QM2
```

Giden bağlantılar için, kanalı aşağıdaki üç yoldan biriyle başlatmalısınız:

1. Kanalı, MCATYPE parametresine bağlı olarak bir işlem ya da iş parçacığı olarak başlatmak için, kanal adını belirterek MQSC komutu START kanalını kullanın. (Kanallar iş parçacığı olarak başlatılırsa, bunlar kanal başlatıcısından oluşan iş parçacıklarıdır.)

```
START CHANNEL(QM1.TO.QM2)
```

2. Kanalı bir işlem olarak başlatmak için runmqchl denetim komutunu kullanın.

```
runmqchl -c QM1.TO.QM2 -m QM1
```

3. Kanalı tetiklemek için kanal başlatıcıyı kullanın.

Kanalı durdurma

Kanalı, etkinliği durdurmasını istemek için MQSC komutu STOP kanalını kullanın. Kanal, kanalı yeniden başlayıncaya kadar kanal yeni bir ileti kümesi başlatmaz.

Durdurulan kanalların yeniden başlatılmasına ilişkin bilgi için bkz. [“Durdurulan kanallar yeniden başlatılıyor” sayfa 62.](#)

Bu komut, MQCHT_CLNTCONN dışında herhangi bir kanala yayınlanabilir.

Gereken durdurma tipini seçebilirsiniz:

Susturma örneğini durdur

```
STOP CHANNEL(QM1.TO.QM2) MODE(QUIESCE)
```

Bu komut, kanaldan düzgün bir şekilde kapanmasını ister. Yürürlükteki ileti kümesi tamamlanır ve eşitleme noktası yordamı, kanalın diğer ucu ile gerçekleştirilir. Kanal boşta olursa, bu komut giriş kanalını sonlandırmaz.

Zorlamalı durdurma örneği

```
STOP CHANNEL(QM1.TO.QM2) MODE(FORCE)
```

Bu seçenek, kanalı hemen durdurur, ancak kanalın iş parçacıklarını ya da sürecini sonlandırmaz. Kanal, yürürlükteki ileti kümesini işlemeyi tamamlamaz ve bu nedenle, kanalı belirsiz olarak bırakabilir. Genel olarak, susturma durdurma seçeneğini kullanmayı göz önünde bulundurun.

Sonlandırma örneğini durdur

```
STOP CHANNEL(QM1.TO.QM2) MODE(TERMINATE)
```

Bu seçenek, kanalı hemen durdurur ve kanalın iş parçacıklarını ya da sürecini sona erdirir.

Stop (quiesce) durdu-örnek

```
STOP CHANNEL(QM1.TO.QM2) STATUS(STOPPED)
```

Bu komut bir MODE belirtmiyor, bu nedenle MODE (QUIESCE) değeri varsayılan olarak ayarlanır. Otomatik olarak yeniden başlatılamaması, ancak el ile başlatılması gerektiği için kanalın durdurulduğunu kabul eder.

Etkin olmayan örnek durdur (quiesce)

```
STOP CHANNEL(QM1.TO.QM2) STATUS(INACTIVE)
```

Bu komut bir MODE belirtmiyor, bu nedenle MODE (QUIESCE) değeri varsayılan olarak ayarlanır. Kanal, gerektiğinde otomatik olarak yeniden başlatılması için devre dışı kılınmasını ister.

Kanalı yeniden adlandırma

İleti kanalını yeniden adlandırmak için MQSC ' yi kullanın.

Aşağıdaki adımları gerçekleştirmek için MQSC ' yi kullanın:

1. Kanalı durdurmak için STOP kanalını kullanın.
2. Yeni adla yinelenen bir kanal tanımlaması yaratmak için KANAL TANIMLA komutunu kullanın.
3. Doğru yaratıldığını denetlemek için DISPLAY CHANNEL ' ı kullanın.
4. Özgün kanal tanımlamasını silmek için DELETE CHANNEL seçeneğini kullanın.

Bir ileti kanalını yeniden adlandırmaya karar verirsiniz, bir kanalda **iki** kanal tanımlaması varsa, her bir uçta bir kanal tanımının olduğunu unutmayın. Kanalı yeniden adlandırdığınızdan emin olun, her iki ucu da aynı anda değiştirin.

Bir kanalı ilk durumuna getirme

İleti sıra numarasını değiştirmek için MQSC komutu RESET CHANNEL 'ı kullanın.

RESET CHANNEL komutu, herhangi bir ileti kanalı için kullanılabilir, ancak MQI kanalları (istemci bağlantısı ya da sunucu bağlantısı) için kullanılamaz. İlk ileti, kanal sonraki başlatıldığı anda yeni sırayı başlatır.

Komut, gönderici ya da sunucu kanalında yayınlanırsa, kanal yeniden başlatıldığında, değişikliğin diğer tarafını bilgilendirir.

İlgili kavramlar

[“Nesnelerle çalışmaya başlama” sayfa 75](#)

Kanallar tanımlanmalıdır ve bir kanal başlatılmadan önce, bunların ilişkili nesnelere var olmalı ve kullanılabilir durumda olmalıdır. Bu bölüm size nasıl olduğunu gösterir.

[“Kanal denetimi işlevi” sayfa 52](#)

Kanal denetim işlevi, kanalları tanımlamanız, izlemeniz ve denetlemeniz için gerekli olanakları sağlar.

[“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27](#)

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eşitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluşları arasındaki iletişim ile ilgili daha ayrıntılı bilgi sağlanmaktadır.

İlgili başvurular

[KANALI](#)

Bir kanaldaki belirsiz iletiler çözülüyor

İletiler gönderici ya da sunucu tarafından belirsiz olarak tutulduğunda MQSC komutu RESOLVE CHANNEL 'ı kullanın. Örneğin, bağlantının bir ucu sona erdirildi ve bunun kurtarılmasına ilişkin bir olasılık da yok.

RESOLVE CHANNEL komutu iki parametreden birini kabul eder: BACKUT ya da COMMIT. Commit, iletileri iletim kuyruğuna geri yüklerken, bunları atar.

Kanal programı bir ortakla oturum kurmayı denemiyor. Bunun yerine, belirsiz iletileri temsil eden mantıksal iş tanıtıcısı (LUWID) birimini belirler. İstendiği gibi, aşağıdaki sorunlar da ortaya çıktı:

- İletilerin iletim kuyruğuna geri yüklenmesini sağlamak için BACKUT ya da
- İletilerin iletim kuyruğundan silinmesi için COMMIT işlemi.

Çözümün başarılı olması için:

- Kanal etkin değil olmalıdır
- Kanal şüphe içinde olmalı.
- Kanal tipi gönderici ya da sunucu olmalıdır
- Yerel kanal tanımlaması var olmalıdır
- Yerel kuyruk yöneticisinin çalışır durumda olması gerekir

İlgili kavramlar

[“Nesnelerle çalışmaya başlama” sayfa 75](#)

Kanallar tanımlanmalıdır ve bir kanal başlatılmadan önce, bunların ilişkili nesnelere var olmalı ve kullanılabilir durumda olmalıdır. Bu bölüm size nasıl olduğunu gösterir.

[“Kanal denetimi işlevi” sayfa 52](#)

Kanal denetim işlevi, kanalları tanımlamanız, izlemeniz ve denetlemeniz için gerekli olanakları sağlar.

[“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27](#)

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eşitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluşları arasındaki iletişim ile ilgili daha ayrıntılı bilgi sağlanmaktadır.

İlgili başvurular

[KANALIN](#)

Windows için iletişimi ayarlama

Dağıtılmış bir kuyruğa alma yönetimi kanalı başlatıldığında, kanal tanımında belirlenen bağlantıyı kullanmaya çalışır. Bunun başarılı olması için, bağlantının tanımlanabilmesini ve kullanılabilir olmasını sağlamak gerekir. Bu bölümde, WebSphere systemsfor Pencerele sistemleri için communicationMQ iletişim formlarından birini kullanarak bu işlem nasıl yapılır açıklanır.

Örnek yapılandırma- IBM WebSphere MQ for Windows' a başvurmanız yararlı olabilir.

UNIX and Linux sistemleri için bkz. [“UNIX and Linux sistemleri üzerinde iletişim kurma” sayfa 91.](#)

Bağlantı üzerinde karar verme

WebSphere MQ for Windows sistemleri için aşağıdaki dört iletişim biçiminden birini seçin:

- [“Windows üzerinde TCP bağlantısı tanımlama” sayfa 82](#)
- [“ Windows üzerinde LU 6.2 bağlantısı tanımlanması” sayfa 84](#)
- [“Windows üzerinde NetBIOS bağlantısının tanımlanması” sayfa 85](#)
- [“Windows üzerinde SPX bağlantısı tanımlama” sayfa 88](#) (yalnızca Windows XP ve Windows 2003 Server)

Her kanal tanımlaması, İletim protokolü (Transport Type) özneliği olarak yalnızca bir protokol belirtmelidir. Bir ya da daha çok protokol kuyruk yöneticisi tarafından kullanılabilir.

WebSphere MQ istemcileri için, farklı iletim protokolleri kullanan alternatif kanalların olması yararlı olabilir. WebSphere MQ istemcileriyle ilgili daha fazla bilgi için bkz. [Müşterilere genel bakış.](#)

İlgili kavramlar

[“Dağıtılmış kuyruklama kullanılarak uygulamaların bağlanması” sayfa 27](#)

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eşitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluşları arasındaki iletişim ile ilgili daha ayrıntılı bilgi sağlanmaktadır.

[“UNIX, Linux, and Windows üzerindeki kanalların izlenmesi ve denetlenmesi” sayfa 72](#)

DQM için, kanalları uzak kuyruk yöneticilerine yaratmanız, izlemeniz ve denetlemeniz gerekir. Komutları, programları, IBM WebSphere MQ Explorer programlarını, kanal tanımlarına ilişkin dosyaları ve eşitleme bilgileri için bir depolama alanını kullanarak kanalları denetleyebilirsiniz.

[“İstemci ile sunucu arasındaki bağlantıların yapılandırılması” sayfa 96](#)

WebSphere MQ MQI istemcileri ve sunucuları arasındaki iletişim bağlantılarını yapılandırmak için iletişim protokolünüze karar verin, bağlantının her iki ucundaki bağlantıları tanımlayın, bir dinleyici başlatın ve kanalları tanımlayın.

Windows üzerinde TCP bağlantısı tanımlama

Gönderilen uçta bir kanal yapılandırarak hedefin adresini belirtmek için ve alıcı uçta bir dinleyici programını çalıştırarak bir TCP bağlantısı tanımlayın.

Gönderme bitişi

Kanal tanımının Bağlantı adı alanında, hedef makinenin anasistem adını ya da TCP adresini belirtin.

Varsayılan değer olarak 1414 'e bağlanmak için kullanılan kapı. Kapı numarası 1414, Internet Tarafından Atanan Numaralar Yetkilisi tarafından IBM WebSphere MQ olarak atanır.

Varsayılan değer dışında bir kapı numarası kullanmak için, kanal nesnesi tanımlamasının bağlantı adı alanında bu değeri belirtin.

```
DEFINE CHANNEL('channel name') CHLTYPE(SDR) +
  TRPTYPE(TCP) +
  CONNAME('OS2ROG3(1822)') +
  XMITQ('XMITQ name') +
  REPLACE
```

Burada OS2R0G3 , uzak kuyruk yöneticisinin DNS adıdır ve 1822 , gereken kapıdır. (Bu, alıcı uçtaki dinleyicinin dinlediği kapı olmalıdır.)

Kanal nesnesi tanımlamasında herhangi bir değişikliği almak için çalışmakta olan bir kanal durdurulmalı ve yeniden başlatılmalıdır.

Varsayılan kapı numarasını, IBM WebSphere MQ for Windows için .ini dosyasında belirterek varsayılan kapı numarasını değiştirebilirsiniz:

```
TCP:  
Port=1822
```

Not: Kullanılacak TCP/IP kapı numarasını seçmek için IBM WebSphere MQ , bulduğu ilk kapı numarasını aşağıdaki sırada kullanır:

1. Kanal tanımlamasında ya da komut satırında açık olarak belirtilen kapı numarası. Bu sayı, bir kanal için varsayılan kapı numarasının geçersiz kılınmasına izin verir.
2. .ini dosyasının TCP stanzasında belirtilen kapı özniteliği. Bu sayı, bir kuyruk yöneticisi için varsayılan kapı numarasının geçersiz kılınmasına izin verir.
3. Varsayılan değer 1414 'tür. Bu, hem gelen hem de giden bağlantılar için Internet Assigny Numbers Yetkisi tarafından IBM WebSphere MQ ' ye atanan numaradır.

qm.inikomutunu kullanarak ayarladığınız değerlerle ilgili daha fazla bilgi için bakınız: [Configuration file stanzas for distributed queuing](#).

TCP ' de Alma

Bir alma kanalı programını başlatmak için, gelen ağ isteklerini algılamaya ve ilişkili kanalı başlatmaya yönelik bir dinleyici programı başlatılmalıdır. IBM WebSphere MQ dinleyicisini kullanabilirsiniz.

Kanal programlarının alınması, gönderme kanalından başlatılan başlatma isteğine yanıt olarak başlatılır.

Bir alma kanalı programını başlatmak için, gelen ağ isteklerini algılamaya ve ilişkili kanalı başlatmaya yönelik bir dinleyici programı başlatılmalıdır. IBM WebSphere MQ dinleyicisini kullanabilirsiniz.

IBM WebSphere MQ ile birlikte verilen Dinleyiciyi çalıştırmak için, iş parçacığı olarak yeni kanalları başlatır, [runmq1sr](#) komutunu kullanın.

runmq1sr komutunu kullanmaya ilişkin temel bir örnek:

```
runmq1sr -t tcp [-m QMNAME] [-p 1822]
```

Köşeli ayraçlar isteğe bağlı deęiştirgeleri belirtir; QMNAME varsayılan kuyruk yöneticisi için gerekli deęildir ve varsayılan deęer (1414) kullanıyorsanız kapı numarası gerekmez. Kapı numarası 65535 'i aşmamalıdır.

Not: Kullanılacak TCP/IP kapı numarasını seçmek için IBM WebSphere MQ , bulduğu ilk kapı numarasını aşağıdaki sırada kullanır:

1. Kanal tanımlamasında ya da komut satırında açık olarak belirtilen kapı numarası. Bu sayı, bir kanal için varsayılan kapı numarasının geçersiz kılınmasına izin verir.
2. .ini dosyasının TCP stanzasında belirtilen kapı özniteliği. Bu sayı, bir kuyruk yöneticisi için varsayılan kapı numarasının geçersiz kılınmasına izin verir.
3. Varsayılan deęer 1414 'tür. Bu, hem gelen hem de giden bağlantılar için Internet Assigny Numbers Yetkisi tarafından IBM WebSphere MQ ' ye atanan numaradır.

For the best performance, run the IBM WebSphere MQ listener as a trusted application as described in “Kanalların ve dinleyicilerin güvenilir uygulamalar olarak çalıştırılması” sayfa 71. Güvenilen uygulamalarla ilgili bilgi için [Güvenilen uygulamalara ilişkin kısıtlamalar](#) başlıklı konuya bakın.

TCP/IP SO_KEEPALIVE seçeneğinin kullanılması

Windows SO_KEEPALIVE seçeneğini kullanmak istiyorsanız, kayıt defterinize aşağıdaki girişi eklemelisiniz:

```
TCP:
KeepAlive=yes
```

SO_KEEPALIVE seçeneğiyle ilgili daha fazla bilgi için bkz. [“Kanalın diğer ucunun hala kullanılabilir olup olmadığını kontrol etme” sayfa 60.](#)

Windows üzerinde, Windows KeepAliveTime seçeneğine ilişkin HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters kayıt değeri, bağlantı denetlenmeden önce geçen aralığı denetler. Varsayılan değer iki saattir.

Windows üzerinde LU 6.2 bağlantısı tanımlanması

SNA, iki makine arasında bir LU 6.2 etkileşimi kurulabilecek şekilde yapılandırılmalıdır.

SNA yapılandırıldıktan sonra aşağıdaki adımları izleyin.

Bilgi için aşağıdaki tabloya bakın.

Çizelge 9. Uzak kuyruk yöneticisi altyapısına ilişkin yerel Windows sistemindeki ayarlar		
Uzak altyapı	TADı	TPPATH
CICSolmadanz/OS ya da MVS/ESA	Uzak kuyruk yöneticisiyle ilgili yan bilgilerdeki gibi.	-
CICSkullanılarakz/OS ya da MVS/ESA	CKRC (gönderen) CKSV (requester) CKRC (sunucu)	-
IBM i	IBM i sistemindeki yöneltme girişindeki karşılaştırma değeriyle aynı.	-
UNIX and Linux sistemleri	Uzak kuyruk yöneticisiyle ilgili yan bilgilerdeki gibi.	MQ_INSTALLATION_PATH/bin/amqcrs6a
Pencereler	As specified in the Pencereler Run Listener command, or the invocable Transaction Program that was defined using TpSetup on Pencereler.	MQ_INSTALLATION_PATH\bin\amqcrs6a

MQ_INSTALLATION_PATH , WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder.

Aynı makinede birden çok kuyruk yöneticisi varsa, kanal tanımlamalarındaki TPN ' lerin benzersiz olduğundan emin olun.

TCP/IP üzerinde AnyNet SNA ' yı yapılandırma hakkında en son bilgiler için, aşağıdaki çevrimiçi IBM belgelerine bakın: [AnyNet SNA over TCP/IP](#) ve [SNA Düğümü İşlemleri](#).

İlgili kavramlar

[“Sending end on LU 6.2” sayfa 84](#)

Kullanmakta olduğunuz LU 6.2 ürününün yönetim uygulamasından bir CPI-C yan nesnesi (simgesel hedef) yaratın. Kanal tanımlamasındaki Bağlantı adı alanına bu adı girin. İş ortağına bir LU 6.2 bağlantısı da yaratın.

[“LU 6.2' ta alınıyor” sayfa 85](#)

Kanal programlarının alınması, gönderme kanalından başlatılan başlatma isteğine yanıt olarak başlatılır.

Sending end on LU 6.2

Kullanmakta olduğunuz LU 6.2 ürününün yönetim uygulamasından bir CPI-C yan nesnesi (simgesel hedef) yaratın. Kanal tanımlamasındaki Bağlantı adı alanına bu adı girin. İş ortağına bir LU 6.2 bağlantısı da yaratın.

CPI-C nesnesi nesnesinde, alıcı makinesinde ortak LU adını, TP adını ve kip adını girin. Örneğin:

```
Partner LU Name      OS2R0G2
Partner TP Name     recv
Mode Name           #INTER
```

LU 6.2' ta alınıyor

Kanal programlarının alınması, gönderme kanalından başlatılan başlatma isteğine yanıt olarak başlatılır.

Bir alma kanalı programını başlatmak için, gelen ağ isteklerini algılamaya ve ilişkili kanalı başlatmaya yönelik bir dinleyici programı başlatılmalıdır. Bu dinleyici programını RUNMQLSR komutuyla başlatıp TpName ' ı dinlemek için bu programı vermenizi sağlar. Alternatively, you can use TpStart under SNA Server for Pencereleler.

RUNMQLSR komutunu kullanma

Dinleyiciyi başlatmak için kullanılan komut örneği:

```
RUNMQLSR -t LU62 -n RECV [-m QMNAME]
```

Burada RECV , diğer (gönderme) sonda belirtilen TpName , uzak tarafta başlatmak için "TpName " olarak sona erdirilir. Köşeli ayraç içindeki son kısım isteğe bağlıdır ve varsayılan kuyruk yöneticisi için gerekli değildir.

Tek bir makinede çalışan birden çok kuyruk yöneticisi olması olanaklıdır. Her kuyruk yöneticisine farklı bir TpName atamanız ve sonra her bir kuyruk yöneticisine bir dinleyici programı başlatmalısınız. Örneğin:

```
RUNMQLSR -t LU62 -m QM1 -n TpName1
RUNMQLSR -t LU62 -m QM2 -n TpName2
```

En iyi performans için, WebSphere MQ dinleyicisini güvenilir bir uygulama olarak çalıştırın ([Kanalların ve dinleyicilerin güvenilir uygulamalar olarak çalıştırılması](#) içinde açıklandığı gibi). Güvenilen uygulamalarla ilgili ek bilgi için [Güvenilen uygulamalara ilişkin kısıtlamalar](#) başlıklı konuya bakın.

Şu komutu kullanarak, etkin olmayan bir kuyruk yöneticisi üzerinde çalışan tüm WebSphere MQ dinleyicilerini durdurabilirsiniz:

```
ENDMQLSR [-m QMNAME]
```

Pencereleler üzerinde Microsoft SNA Server 'ın kullanılması

You can use TpSetup (from the SNA Server SDK) to define an invocable TP that then drives amqcrs6a.exe, or you can set various registry values manually. amqcrs6a.exe ' ye geçilmesi gereken parametreler şunlardır:

```
-m QM -n TpName
```

Burada QM , Kuyruk Yöneticisi adı ve TpName TP adıdır. Ek bilgi için *Microsoft SNA Server APPC Programmers Guide* ya da *Microsoft SNA Server CPI-C Programmers Guide* adlı kılavuza bakın.

Bir kuyruk yöneticisi adı belirtmezseniz, varsayılan kuyruk yöneticisi varsayılan olarak kabul edilir.

Windows üzerinde NetBIOS bağlantısının tanımlanması

WebSphere MQ , başka bir WebSphere MQ ürünü için NetBIOS bağlantısı kurarken üç tip NetBIOS kaynağı kullanır: oturumlar, komutlar ve adlar. Bu kaynakların her birinin bir sınırı vardır; bu sınır, varsayılan olarak ya da NetBIOS kurulumu sırasında seçim yoluyla kurulur.

Her bir çalışan kanal, tipten bağımsız olarak, bir NetBIOS oturumu ve bir NetBIOS komutu kullanır. IBM NetBIOS uygulaması, birden çok işlemin aynı yerel NetBIOS adını kullanmasına olanak tanır. Bu nedenle,

WebSphere MQ tarafından kullanılmak üzere yalnızca bir NetBIOS adının kullanılabilir olması gerekir. Örneğin, Novell'in NetBIOS öykünmesi gibi diğer satıcı firmaların uygulamaları, her işlem için farklı bir yerel ad gerektirir. Kullanmakta olduğunuz NetBIOS ürününe ilişkin belgelerden gereksinimlerinizi doğrulayın.

Tüm durumlarda, her tipte yeterli kaynağın kullanılabilir olduğundan emin olun ya da yapılandırmada belirtilen maksimumları artırın. Değerlerinde yapılan herhangi bir değişiklik sistemin yeniden başlatılmasını gerektirir.

Sistem başlatma sırasında, NetBIOS aygıt sürücüsü, uygulamalar tarafından kullanılmak üzere kullanılabilir oturum, komut ve ad sayısını görüntüler. Bu kaynaklar, aynı sistem üzerinde çalışan herhangi bir NetBIOS tabanlı uygulama için kullanılabilir. Bu nedenle, WebSphere MQ'un bunları edinmesi gerekmeden önce diğer uygulamaların bu kaynakları tüketmesi mümkündür. LAN ağ denetimciniz bunu sizin için açıklığa kavuşturabilmelidir.

İlgili kavramlar

[“IBM WebSphere MQ yerel NetBIOS adının tanımlanması” sayfa 86](#)

IBM WebSphere MQ kanal işlemleri tarafından kullanılan yerel NetBIOS adı üç şekilde belirtilebilir.

[“Kuyruk yöneticisi NetBIOS oturum, komut ve ad sınırları oluşturulması” sayfa 87](#)

NetBIOS oturumları, komutları ve adlarına ilişkin kuyruk yöneticisi sınırları iki şekilde belirtilebilir.

[“LAN bağdaştırıcısı numarasının oluşturulması” sayfa 87](#)

Kanalların NetBIOS genelinde başarılı bir şekilde çalışması için, her uçta bağdaştırıcı desteğinin uyumlu olması gerekir. IBM WebSphere MQ qm.ini kütüğünüzün NETBIOS kısmında AdapterNum değerini kullanarak ve runmqslr komutunda -a değiştirgesini belirterek LAN bağdaştırıcısı (LANA) sayısını denetlemenizi sağlar.

[“NetBIOS bağlantısının başlatılması” sayfa 87](#)

Bağlantı başlatmak için gereken adımları tanımlama.

[“NetBIOS bağlantısı için hedef dinleyici” sayfa 88](#)

NetBIOS bağlantısının alıcı uçta üstlenilecek adımları tanımlama.

IBM WebSphere MQ yerel NetBIOS adının tanımlanması

IBM WebSphere MQ kanal işlemleri tarafından kullanılan yerel NetBIOS adı üç şekilde belirtilebilir.

Öncelik sırasına göre üç yol aşağıda yer alıyor:

1. RUNMQLSR komutunun -l değiştirgesinde belirtilen değer, örneğin:

```
RUNMQLSR -t NETBIOS -l my_station
```

2. MQNAME ortam değişkeni, komut tarafından oluşturulan bir değere sahip:

```
SET MQNAME=my_station
```

Her işlem için MQNAME değerini ayarlayabilirsiniz. Diğer bir seçenek olarak, bunu Windows kayıt defterindeki bir sistem düzeyinde de ayarlayabilirsiniz.

Benzersiz adlar gerektiren bir NetBIOS somutlaması kullanıyorsanız, bir IBM WebSphere MQ işleminin başlatıldığı her bir pencerede bir SET MQNAME komutu yayınlamanız gerekir. MQNAME değeri rasgele bir değer, ancak her işlem için benzersiz olmalıdır.

3. The NETBIOS stanza in the queue manager configuration file qm.ini. Örneğin:

```
NETBIOS:  
LocalName=my_station
```

Not:

1. Desteklenen NetBIOS ürünlerinin uygulanmasındaki varyasyonlar nedeniyle, ağdaki her NetBIOS adını benzersiz bir ad olarak yapmanız önerilir. Bunu yapmazsanız, beklenmedik sonuçlar ortaya çıkabilir.

If you have problems establishing a NetBIOS channel and there are error messages in the queue-manager error log showing a NetBIOS return code of X'15', review your use of NetBIOS names.

2. On Windows, you cannot use your machine name as the NetBIOS name because Windows already uses it.
3. Gönderen kanalı başlatma işlemi, bir NetBIOS adının MQNAME ortam değişkeni kullanılarak ya da qm.ini dosyasındaki LocalName kullanılarak belirtilmesini gerektirir.

Kuyruk yöneticisi NetBIOS oturum, komut ve ad sınırları oluşturulması

NetBIOS oturumları, komutları ve adlarına ilişkin kuyruk yöneticisi sınırları iki şekilde belirtilebilir.

Öncelik sırasına göre aşağıdaki şekillere dikkat edin:

1. RUNMQLSR komutunda belirtilen değerler:

```
-s Sessions  
-e Names  
-o Commands
```

Komutta -m işleneni belirlenmezse, değerler yalnızca varsayılan kuyruk yöneticisine uygulanır.

2. The NETBIOS stanza in the queue manager configuration file qm.ini. Örneğin:

```
NETBIOS:  
  
NumSess=Qmgr_max_sess  
NumCmds=Qmgr_max_cmds  
NumNames=Qmgr_max_names
```

LAN bağdaştırıcısı numarasının oluşturulması

Kanalların NetBIOSgenelinde başarılı bir şekilde çalışması için, her uçta bağdaştırıcı desteğinin uyumlu olması gerekir. IBM WebSphere MQ qm.ini kütüğünüzün NETBIOS kısmında AdapterNum değerini kullanarak ve runmqslsr komutunda -a değiştirgesini belirterek LAN bağdaştırıcısı (LANA) sayısını denetlemenizi sağlar.

The default LAN adapter number used by IBM WebSphere MQ for NetBIOS connections is 0. Sisteminizde kullanılmakta olan numaranız aşağıdaki gibi doğrulanıyor:

Windowsişletim sisteminde, LAN bağdaştırıcısı numarasını doğrudan işletim sistemi aracılığıyla sorgulamak mümkün değildir. Bunun yerine, LANACFG.EXE komut satırı yardımcı programı (Microsoft) tarafından kullanılabilir. Aracın çıkışı, sanal LAN bağdaştırıcısı numaralarını ve bunların etkin bağ tanımlarını gösterir. LAN bağdaştırıcısı numaraları hakkında daha fazla bilgi için bkz. Microsoft Knowledge Base yazısı 138037 *HOWTO: 32 bitlik bir Ortamda LANA Numaralarını kullanın.*

queuekuyruk yöneticisi yapılanış kütüğünün NETBIOS kısmında doğru değeri belirtin. qm.ini:

```
NETBIOS:  
AdapterNum=n
```

Burada n, bu sisteme ilişkin doğru LAN bağdaştırıcısı numarasıdır.

NetBIOS bağlantısının başlatılması

Bağlantı başlatmak için gereken adımları tanımlama.

Bağlantıyı başlatmak için, gönderme bitişindeki şu adımları izleyin:

1. MQNAME ya da LocalName değerini kullanarak NetBIOS istasyonu adını tanımlayın.
2. Verify the LAN adapter number being used on your system and specify the correct file using the AdapterNum.

3. Kanal tanımının ConnectionName alanında, hedef dinleyici programı tarafından kullanılmakta olan NetBIOS adını belirtin. On Pencereleler, NetBIOS channels **gerekir** be run as threads. Bunu, kanal tanımlamasında MCATYPE (THREAD) belirtilerek yapın.

```
DEFINE CHANNEL (chname) CHLTYPE(SDR) +  
  TRPTYPE(NETBIOS) +  
  CONNAME(your_station) +  
  XMITQ(xmitq) +  
  MCATYPE(THREAD) +  
  REPLACE
```

NetBIOS bağlantısı için hedef dinleyici

NetBIOS bağlantısının alıcı uçunda üstlenilecek adımları tanımlama.

Alıcı uçta aşağıdaki adımları izleyin:

1. MQNAME ya da LocalName değerini kullanarak NetBIOS istasyonu adını tanımlayın.
2. Verify the LAN adapter number being used on your system and specify the correct file using the AdapterNum.
3. Alıcı kanalını tanımlayın:

```
DEFINE CHANNEL (chname) CHLTYPE(RCVR) +  
  TRPTYPE(NETBIOS) +  
  REPLACE
```

4. İstasyon oluşturmak için WebSphere MQ dinleyici programını başlatın ve iletişim kurmayı mümkün kılabilir. Örneğin:

```
RUNMQLSR -t NETBIOS -l your_station [-m qmgr]
```

Bu komut, iletişim kurulmasını bekleyen bir NetBIOS istasyonu olarak your_station 'yi oluşturur. NetBIOS istasyonu adı, NetBIOS ağıncı boyunca benzersiz olmalıdır.

En iyi performans için, WebSphere MQ dinleyicisini güvenilir bir uygulama olarak çalıştırın (“Kanalların ve dinleyicilerin güvenilir uygulamalar olarak çalıştırılması” sayfa 71’inde açıklandığı gibi). Güvenilen uygulamalarla ilgili ek bilgi için [Güvenilen uygulamalara ilişkin kısıtlamalar](#) başlıklı konuya bakın.

Şu komutu kullanarak, etkin olmayan bir kuyruk yöneticisi üzerinde çalışan tüm WebSphere MQ dinleyicilerini durdurabilirsiniz:

```
ENDMQLSR [-m QMNAME]
```

Bir kuyruk yöneticisi adı belirtmezseniz, varsayılan kuyruk yöneticisi varsayılan olarak kabul edilir.

Windows üzerinde SPX bağlantısı tanımlama

Bir SPX bağlantısı yalnızca, Windows XP ve Windows 2003 Server çalıştıran bir istemci ve sunucu için geçerlidir.

Gönderme bitişindeki kanal tanımlaması hedefin adresini belirtir. Bir dinleyici programının alıcı uçta çalıştırılması gerekir.

İlgili kavramlar

“SSPX üzerine son gönderiliyor” sayfa 89

Hedef makine uzaksa, kanal tanımının Bağlantı adı alanında hedef makinenin SPX adresini belirtin.

“SPX üzerinde alınıyor” sayfa 89

Kanal programlarının alınması, gönderme kanalından başlatılan başlatma isteğine yanıt olarak başlatılır.

“IPX/SPX parametreleri” sayfa 90

Çoğu durumda, IPX/SPX parametrelerinin varsayılan ayarları gereksinimlerinize uygun olacaktır. Ancak, ortamınızda WebSphere MQ için kullanımını ayarlamak üzere bu bazı ortamlardan bazılarını değiştirmeniz gerekebilir.

SSPX üzerine son gönderiliyor

Hedef makine uzaksa, kanal tanımının Bağlantı adı alanında hedef makinenin SPX adresini belirtin.

SPX adresi aşağıdaki biçimde belirtilir:

```
network.node(socket)
```

Burada:

network

Uzak makinenin bulunduğu ağın 4 baytlık ağ adresidir.

node

Uzak makineden LAN bağdaştırıcısının LAN adresi olan 6 baytlık düğüm adresidir.

socket

Uzak makinenin dinlediği 2 baytlık yuva numarasıdır.

Yerel ve uzak makineler aynı ağdaysa, ağ adresi belirtilmemesi gerekir. Uzak uç varsayılan yuvayı (5E86) dinliyorsa, yuvanın belirtilmemesi gerekir.

Bir MQSC komutunun CONNAME parametresinde belirtilen tam olarak belirtilmiş bir SPX adresine ilişkin bir örnek:

```
CONNNAME('00000001.08005A7161E5(5E87)')
```

Makinelerin her ikisi de aynı ağ üzerinde olduğu varsayılan durumda, bu durum şöyle olur:

```
CONNNAME(08005A7161E5)
```

Varsayılan yuva numarası, kuyruk yöneticisi yapılanış kütüğünde (qm.ini) tanımlanarak değiştirilebilir:

```
SPX:  
Socket=5E87
```

qm.inikomutunu kullanarak ayarladığınız değerlerle ilgili daha fazla bilgi için bakınız: [Configuration file stanzas for distributed queuing](#).

SPX üzerinde alınıyor

Kanal programlarının alınması, gönderme kanalından başlatılan başlatma isteğine yanıt olarak başlatılır.

Bir alma kanalı programını başlatmak için, gelen ağ isteklerini algılamaya ve ilişkili kanalı başlatmaya yönelik bir dinleyici programı başlatılmalıdır.

WebSphere MQ dinleyicisini kullanın.

SPX dinleyici birikim arka günlüğü seçeneğinin kullanılması

SPX 'te alınırken, bekleyen bağlantı isteği sayısı üst sınırı belirlendi. Bu, dinleyicinin isteği kabul etmesi için SPX kapısında bekleyen isteklerin *birikim günlüğü* olarak düşünülebilmektedir. Varsayılan dinleyici birikim günlüğü değerleri [Çizelge 10 sayfa 89](#) içinde gösterilir.

Çizelge 10. Pencereler' ta varsayılan bekleyen bağlantı istekleri	
Altyapı	Varsayılan dinleyici birikim günlüğü değeri
Windows Server	5

Çizelge 10. Pencerele' ta varsayılan bekleyen bağlantı istekleri (devamı var)	
Altyapı	Varsayılan dinleyici birikim günlüğü değeri
Windows İş İstasyonu	5

Birikim Çizelge 10 sayfa 89' daki değerlere ulaşırsa, neden kodu, MQCONN ya da MQCONNX kullanılarak kuyruk yöneticisine bağlanma girişimi sırasında MQRC_Q_MGR_NOT_AVAM değeri alınır. Bu gerçekleşirse, yeniden bağlanmayı denemek mümkündür.

However, to avoid this error, you can add an entry in the qm.ini file or in the registry for Pencerele:

```
SPX:
ListenerBacklog = n
```

Bu, SPX dinleyicisi için varsayılan üst sınır olan istek sayısı üst sınırını geçersiz kılar (bkz. Çizelge 10 sayfa 89).

Not: Bazı işletim sistemleri, varsayılan değerden daha büyük bir değeri destekler. Gerekirse, bağlantı sınırına ulaşmaktan kaçınmak için bu kullanılabilir.

Dinleyiciyi birikim seçeneğiyle çalıştırmak için aşağıdakilerden birini kullanın:

- RUNMQLSR -b komutunu kullanın ya da
- BACKLOG öznitelikle **DEFINE LISTENER** MQSC komutunu gerekli değere ayarlayın.

RUNMQLSR komutuna ilişkin bilgi için bkz. [runmqlsr](#). **DEFINE LISTENER** komutuna ilişkin bilgi edinmek için [DEFINE LISTENER](#) başlıklı konuya bakın.

WebSphere MQ dinleyicisinin kullanılması

To run the Listener supplied with WebSphere MQ, that starts new channels as threads, use the RUNMQLSR command. Örneğin:

```
RUNMQLSR -t spx [-m QMNAME] [-x 5E87]
```

Köşeli ayraçlar isteğe bağlı değiştirgeleri belirtir; QMNAME varsayılan kuyruk yöneticisi için gerekli değildir ve varsayılan değer (5E86) kullanıyorsanız yuva numarası gerekmez.

En iyi performans için, WebSphere MQ dinleyicisini güvenilir bir uygulama olarak çalıştırın (“Kanalların ve dinleyicilerin güvenilir uygulamalar olarak çalıştırılması” sayfa 71’inde açıklandığı gibi). Güvenilen uygulamalarla ilgili ek bilgi için [Güvenilen uygulamalara ilişkin kısıtlamalar](#) başlıklı konuya bakın.

Şu komutu kullanarak, etkin olmayan bir kuyruk yöneticisi üzerinde çalışan tüm WebSphere MQ dinleyicilerini durdurabilirsiniz:

```
ENDMQLSR [-m QMNAME]
```

Bir kuyruk yöneticisi adı belirtmezseniz, varsayılan kuyruk yöneticisi varsayılan olarak kabul edilir.

IPX/SPX parametreleri

Çoğu durumda, IPX/SPX parametrelerinin varsayılan ayarları gereksinimlerinize uygun olacaktır. Ancak, ortamınızda WebSphere MQ için kullanımını ayarlamak üzere bu bazı ortamlardan bazılarını değiştirmeniz gerekebilir.

Gerçek parametreler ve bunların değiştirilmesi yöntemi, SPX iletişim desteğinin platformuna ve sağlayıcısına göre değişiklik gösterir. Örnek kısım, özellikle WebSphere MQ kanallarının ve istemci bağlantılarının çalışmasını etkileyebilecek olan bu parametrelerden bazılarını açıklar.

Windows sistemleri

NWLink IPX ve SPX parametrelerinin kullanımına ve ayarına ilişkin tam ayrıntılar için Microsoft belgelerine bakın. IPX/SPX deęiřtirgeleri kayıt defterinde ařaęıdaki yollarda bulunur:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Service\NWLinkSPX\Parameters  
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Service\NWLinkIPX\Parameters
```

UNIX and Linux sistemleri üzerinde iletiřim kurma

DQM, IBM WebSphere MQ için uzak bir kuyruęa alma tesisidir. Sistem iřletmeni tarafından denetlenir taban nesne iletiřim baęlantılarına arabirim oluřturan kuyruk yöneticisi için kanal denetim programları saęlar. Daęıtılmıř kuyruklama yönetimi tarafından tutulan kanal tanımları bu baęlantıları kullanır.

Daęıtılmıř bir kuyruęa alma yönetimi kanalı bařlatıldıęında, kanal tanımında belirlenen baęlantıyı kullanmaya çalıřır. Bařarılı olmak için, baęlantının tanımlanabilmesini ve kullanılabilir olmasını saęlamak gerekir. Bu bölümde, bu iřlem nasıl yapılır açıklanır. Ayrıca, ařaęıdaki kısımlara bařvurmanız yararlı olabilir:

- [Örnek yapılandırma- IBM WebSphere MQ for AIX](#)
- [Örnek yapılandırma- IBM WebSphere MQ for HP-UX](#)
- [Örnek yapılandırma- IBM WebSphere MQ for Solaris](#)
- [Örnek yapılandırma- Linux için IBM WebSphere MQ](#)

Windows için bkz. “ Windows için iletiřimi ayarlama” sayfa 82.

UNIX and Linux sistemlerinde WebSphere MQ için iki iletiřim biçimi arasında seçim yapabilirsiniz:

- [“UNIX and Linux üzerinde TCP baęlantısı tanımlama” sayfa 91](#)
- [“UNIX and Linux üzerinde LU 6.2 baęlantısının tanımlanması” sayfa 95](#)

Her kanal tanımlaması yalnızca iletim protokolü (Transport Type) öznitelięi olarak bir kanal tanımlaması belirtmelidir. Bir ya da daha çok protokol kuyruk yöneticisi tarafından kullanılabilir.

IBM WebSphere MQ Explorer MQI istemcileri için, farklı iletim protokollerini kullanan alternatif kanalların olması yararlı olabilir. IBM WebSphere MQ Explorer MQI istemcileriyle ilgili daha fazla bilgi için bkz. [IBM WebSphere MQ MQI istemcilerine genel bakıř](#) .

İlgili kavramlar

[“Daęıtılmıř kuyruklama kullanılarak uygulamaların baęlanması” sayfa 27](#)

Bu bölümde, kuyruk tanımlaması, kanal tanımlaması, tetikleme ve eřitleme noktası yordamları da içinde olmak üzere WebSphere MQ kuruluřları arasındaki iletiřim ile ilgili daha ayrıntılı bilgi saęlanmaktadır.

[“UNIX, Linux, and Windows üzerindeki kanalların izlenmesi ve denetlenmesi” sayfa 72](#)

DQM için, kanalları uzak kuyruk yöneticilerine yaratmanız, izlemeniz ve denetlemeniz gerekir. Komutları, programları, IBM WebSphere MQ Explorer programlarını, kanal tanımlarına iliřkin dosyaları ve eřitleme bilgileri için bir depolama alanını kullanarak kanalları denetleyebilirsiniz.

[“İstemci ile sunucu arasındaki baęlantıların yapılandırılması” sayfa 96](#)

WebSphere MQ MQI istemcileri ve sunucuları arasındaki iletiřim baęlantılarını yapılandırmak için iletiřim protokolünüze karar verin, baęlantının her iki ucundaki baęlantıları tanımlayın, bir dinleyici bařlatın ve kanalları tanımlayın.

UNIX and Linux üzerinde TCP baęlantısı tanımlama

Gönderme bitiřindeki kanal tanımlaması hedefin adresini belirtir. İletiřimci ya da inet cini, alma uçta baęlantı için yapılandırıldı.

Gönderme bitiři

Kanal tanımının Baęlantı Adı alanında, hedef makinenin anasistem adını ya da TCP adresini belirtin. Varsayılan deęer olarak 1414 'e baęlanmak için kullanılan kapı. Kapı numarası 1414, Internet Assigny Numbers Authority tarafından WebSphere MQ olarak atanır.

Varsayılan değer dışında bir kapı numarası kullanmak için, bağlantı adı alanını bu şekilde değiştirin:

```
Connection Name REMHOST(1822)
```

Burada REMHOST , uzak makinenin anasistem adı ve 1822 , gereken kapı numarasıdır. (Bu, alıcı uçtaki dinleyicinin dinlediği kapı olmalıdır.)

Diğer bir seçenek olarak, kapı numarasını kuyruk yöneticisi yapılandırma kütüğünde (qm.ini) belirterek değiştirebilirsiniz.

```
TCP:  
Port=1822
```

qm.inikomutunu kullanarak ayarladığınız değerlerle ilgili daha fazla bilgi için bakınız: [Configuration file stanzas for distributed queuing](#).

TCP ' de Alma

Bu inet cini (inetd) ya da WebSphere MQ dinleyicisi olan TCP/IP iletişimcisini kullanabilirsiniz.

Bazı Linux dağıtımları artık inet yardımcı programı yerine genişletilmiş inet cinini (xinetd) kullanır. Bir Linux sisteminde genişletilmiş inet cininin nasıl kullanılacağı hakkında bilgi için bkz. [Linuxüzünde TCP bağlantısı oluşturma](#).

İlgili kavramlar

[“TCP/IP dinleyicisinin kullanılması” sayfa 92](#)

To start channels on UNIX and Linux, the /etc/services file and the inetd.conf file must be edited

[“TCP dinleyici arka günlüğü seçeneğinin kullanılması” sayfa 93](#)

TCP ' de, sunucu ile istemci arasında üç yönlü tokalaşma gerçekleşmedikçe, bağlantılar eksik değerlendirilir. Bu bağlantılara, bekleyen bağlantı istekleri denir. Bu bekleyen bağlantı istekleri için bir üst sınır değeri belirlenir ve dinleyicinin isteği kabul etmesi için TCP kapısında bekleyen isteklerin arka günlüğü olarak düşünülebilirler.

[“ WebSphere MQ dinleyicisinin kullanılması” sayfa 94](#)

İş parçacığı olarak yeni kanalları başlatan WebSphere MQ ile sağlanan dinleyiciyi çalıştırmak için `runmqtsr` komutunu kullanın.

[“TCP/IP SO_KEEPALIVE seçeneğinin kullanılması” sayfa 95](#)

Bazı UNIX and Linux sistemlerinde, bağlantının hala kullanılabilir olup olmadığını denetlemeden önce, TCP ' nin ne kadar bekleyeceğini ve ilk denetinin başarısız olması durumunda bağlantıyı yeniden ne sıklıkta denediğini tanımlayabilirsiniz. Bu bir çekirdek ayarlanabilen bir değiştirgeci ya da komut satırına girilebilir.

TCP/IP dinleyicisinin kullanılması

To start channels on UNIX and Linux, the /etc/services file and the inetd.conf file must be edited

Aşağıdaki yönergeleri izleyin:

1. /etc/services kütüğünü düzenleyin:

Not: /etc/services kütüğünü düzenlemek için, bir ayrıcalıklı kullanıcı ya da kök olarak oturum açmış olmanız gerekir. Bunu değiştirebilirsiniz, ancak gönderenin sonunda belirtilen kapı numarasıyla eşleşmesi gerekir.

Şu satırı dosyaya ekleyin:

```
MQSeries 1414/tcp
```

Burada 1414, WebSphere MQ tarafından gerekli olan kapı numarasıdır. Kapı numarası 65535 'i aşmamalıdır.

2. Programı amqcrta olarak çağırarak için inetd.conf dosyasına bir satır ekleyin; burada `MQ_INSTALLATION_PATH`, WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder:

```
MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta  
[-m Queue_Man_Name]
```

Güncellemeler, inetd yapılandırma kütüklerini yeniden okuduktan sonra etkindir. Bunu yapmak için, kök kullanıcı kimliğinden aşağıdaki komutları verin:

- AIX üzerinde:

```
refresh -s inetd
```

- HP-UX üzerinde, mqm kullanıcı kimliğinden:

```
inetd -c
```

- Solaris 10 ya da sonraki yayın düzeylerinde:

```
inetconv
```

- Diğer UNIX and Linux sistemlerinde (Solaris 9 da içinde olmak üzere):

```
kill -1 <process number>
```

inetd tarafından başlatılan dinleyici programı, inetd yerel ayarını devraldığına, MQMDE ' nin yerine getirilmemesi (birleştirilmemiş) ve kuyruğa ileti verisi olarak yerleştirilmek mümkündür. MQMDE ' nin yerine getirildiğinden emin olmak için, yerel ayarı doğru olarak ayarlamanız gerekir. inetd tarafından ayarlanan yerel ayar, WebSphere MQ işlemleri tarafından kullanılan diğer ülke değerleri için seçilen eşleşmeyebilir. Yerel ayarı ayarlamak için:

1. LANG, LC_COLLATE, LC_CTYPE, LC_PARA, LC_NUMERIC, LC_TIME ve LC_MESSAGES yerel ortam değişkenlerini, diğer WebSphere MQ işlemi için kullanılan yerel ayara ayarlayan bir kabuk komut dosyası yaratır.
2. Aynı kabuk komut dosyasında, dinleyici programını çağırın.
3. Dinleyici programı yerine kabuk komut dosyanızı çağırarak için inetd.conf dosyasını değiştirin.

Sunucuda birden çok kuyruk yöneticisi olması olanaklıdır. Her kuyruk yöneticisi için, her iki dosyanın her birine bir satır eklemelisiniz. Örneğin:

```
MQSeries1    1414/tcp  
MQSeries2    1822/tcp
```

```
MQSeries2 stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta -m QM2
```

Burada `MQ_INSTALLATION_PATH`, WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder.

Bu, tek bir TCP kapısında kuyruğa alınan bekleyen bağlantı isteği sayısı sınırlaması varsa, bu hata iletilerinin oluşturulmasını önler. Bekleyen bağlantı isteklerinin sayısına ilişkin bilgi için bkz. [“TCP dinleyici arka günlüğü seçeneğinin kullanılması” sayfa 93.](#)

TCP dinleyici arka günlüğü seçeneğinin kullanılması

TCP ' de, sunucu ile istemci arasında üç yönlü tokalaşma gerçekleşmedikçe, bağlantılar eksik değerlendirilir. Bu bağlantılara, bekleyen bağlantı istekleri denir. Bu bekleyen bağlantı istekleri için bir üst sınır değeri belirlenir ve dinleyicinin isteği kabul etmesi için TCP kapısında bekleyen isteklerin arka günlüğü olarak düşünülebilirler.

Varsayılan dinleyici birikim günlüğü değerleri [Çizelge 11 sayfa 94](#) içinde gösterilir.

Çizelge 11. TCP/IP kapısında kuyruğa alınan bekleyen bağlantı isteği sayısı üst sınırı	
Sunucu platformu	Bağlantı isteği sayısı üst sınırı
AIX	100
HP-UX	20
Linux	100
IBM i	255
Solaris	100
Windows Server	100
Windows İş İstasyonu	100
z/OS	255

Birikim, Çizelge 11 sayfa 94’inde gösterilen değerlere ulaşırsa, TCP/IP bağlantısı reddedilir ve kanal başlatılamaz.

MCA kanalları için, kanaldaki bu sonuçlar bir RETRY durumuna geçiyor ve daha sonra bağlantıyı yeniden deneyiyor.

Ancak, bu hatayı önlemek için, qm.ini dosyasına bir giriş ekleyebilirsiniz:

```
TCP:
ListenerBacklog = n
```

Bu işlem, TCP/IP dinleyicisi için varsayılan değer olan bekleyen istek sayısı üst sınırını (bkz. Çizelge 11 sayfa 94) geçersiz kılar.

Not: Bazı işletim sistemleri, varsayılan değerden daha büyük bir değeri destekler. Gerekirse, bağlantı sınırına ulaşmamak için bu değer kullanılabilir.

Dinleyiciyi birikim seçeneğiyle çalıştırmak için aşağıdakilerden birini kullanın:

- `runmq1sr -b` komutunu kullanın ya da
- BACKLOG öznitelikle **DEFINE LISTENER MQSC** komutunu gerekli değere ayarlayın.

runmq1sr komutuna ilişkin bilgi için bkz. [runmq1sr](#). **DEFINE LISTENER** komutuna ilişkin bilgi edinmek için **DEFINE LISTENER** başlıklı konuya bakın.

WebSphere MQ dinleyicisinin kullanılması

İş parçacığı olarak yeni kanalları başlatan WebSphere MQ ile sağlanan dinleyiciyi çalıştırmak için `runmq1sr` komutunu kullanın.

Örneğin:

```
runmq1sr -t tcp [-m QMNAME] [-p 1822]
```

Köşeli ayraçlar isteğe bağlı değiştirgeleri belirtir; QMNAME varsayılan kuyruk yöneticisi için gerekli değildir ve varsayılan değer (1414) kullanıyorsanız kapı numarası gerekmez. Kapı numarası 65535 'i aşmamalıdır.

En iyi performans için, WebSphere MQ dinleyicisini güvenilir bir uygulama olarak çalıştırın (“Kanalların ve dinleyicilerin güvenilir uygulamalar olarak çalıştırılması” sayfa 71’inde açıklandığı gibi). Güvenilen uygulamalarla ilgili ek bilgi için [Güvenilen uygulamalara ilişkin kısıtlamalar](#) başlıklı konuya bakın.

Şu komutu kullanarak, etkin olmayan bir kuyruk yöneticisi üzerinde çalışan tüm WebSphere MQ dinleyicilerini durdurabilirsiniz:

```
endmq1sr [-m QMNAME]
```

Bir kuyruk yöneticisi adı belirtmezseniz, varsayılan kuyruk yöneticisi varsayılan olarak kabul edilir.

TCP/IP SO_KEEPALIVE seçeneğinin kullanılması

Bazı UNIX and Linux sistemlerinde, bağlantının hala kullanılabilir olup olmadığını denetlemeden önce, TCP ' nin ne kadar bekleyeceğini ve ilk denetimin başarısız olması durumunda bağlantıyı yeniden ne sıklıkta denediğini tanımlayabilirsiniz. Bu bir çekirdek ayarlanabilen bir deęiştirgeci ya da komut satırına girilebilir.

SO_KEEPALIVE seçeneğini kullanmak istiyorsanız (daha fazla bilgi için bkz. “Kanalın dięer ucunun hala kullanılabilir olup olmadığını kontrol etme” sayfa 60) kuyruk yöneticisi yapılanış kütüğünüze (qm.ini) aşağıdaki girişı eklemelisiniz:

```
TCP:
  KeepAlive=yes
```

Ek bilgi için UNIX and Linux sisteminize ilişkin belgelere bakın.

UNIX and Linux üzerinde LU 6.2 bağlantısının tanımlanması

SNA, iki makine arasında bir LU 6.2 etkileşimi kurulabilecek şekilde yapılandırılmalıdır.

TCP/IP üzerinde SNA ' yı yapılandırmaya ilişkin en son bilgiler için, aşağıdaki çevrimiçi IBM belgesine bakın: [Communications Server](#).

SNA, iki sistem arasında bir LU 6.2 etkileşimi kurulabilecek şekilde yapılandırılmalıdır.

Bilgi için *Multiplatform APPC Configuration Guide* adlı belgeye ve aşağıdaki çizelgeye bakın.

Uzak altyapı	TADı	TPPATH
CICS olmadan z/OS	Uzak kuyruk yöneticisine ilişkin yan bilgilerdeki karşılık gelen TPName ile aynıdır.	-
z/OS using CICS	CKRC (gönderen) CKSV (requester) CKRC (sunucu)	-
IBM i	IBM i sistemindeki yöneltme girişindeki karşılaştırma deęeriyle aynı.	-
UNIX and Linux sistemleri	Uzak kuyruk yöneticisine ilişkin yan bilgilerdeki karşılık gelen TPName ile aynıdır.	<i>MQ_INSTALLATION_PATH/bin/amqcrs6a</i>
Pencereler	As specified in the Pencereler Run Listener command, or the invocable Transaction Program that was defined using TpSetup on Pencereler.	<i>MQ_INSTALLATION_PATH\bin\amqcrs6a</i>

MQ_INSTALLATION_PATH , WebSphere MQ ' un kurulu olduęu üst düzey dizini temsil eder.

Aynı makinede birden çok kuyruk yöneticisi varsa, kanal tanımlamalarındaki TPN ' lerin benzersiz olduęundan emin olun.

İlgili kavramlar

[“Gönderme bitişı” sayfa 96](#)

UNIX and Linux sistemlerinde, bir CPI-C yan nesnesi (simgesel hedef) yaratın ve kanal tanımlamasındaki Bağlantı adı alanına bu adı girin. İş ortağına bir LU 6.2 bağlantısı da yaratın.

[“LU 6.2' ta alınıyor” sayfa 96](#)

UNIX and Linux sistemlerinde, alıcı uçta bir dinleme bağlantısı, bir LU 6.2 mantıksal bağlantı tanıtımı ve bir TPN tanıtımı yaratın.

Gönderme bitişi

UNIX and Linux sistemlerinde, bir CPI-C yan nesnesi (simgesel hedef) yaratın ve kanal tanımlamasındaki Bağlantı adı alanına bu adı girin. İş ortağına bir LU 6.2 bağlantısı da yaratın.

CPI-C nesnesi nesnesi, alıcı makinesinde ortak LU adını, hareket programı adını ve kip adını girin. Örneğin:

```
Partner LU Name      REMHOST
Remote TP Name       recv
Service Transaction Program no
Mode Name            #INTER
```

HP-UX üzerinde, gönderenin kullanması gereken yerel LU 'ya adlamak için APPCLLU ortam değişkenini kullanın. Solaris üzerinde, APPC_LOCAL_LU ortam değişkenini yerel LU adı olacak şekilde ayarlayın.

WebSphere MQ bir SNA oturumu kurma girişiminde bulunduğunda CPI-C tarafından desteklenen SECURITY PROGRAMı KULLANILIR.

LU 6.2' ta alınıyor

UNIX and Linux sistemlerinde, alıcı uçta bir dinleme bağlantısı, bir LU 6.2 mantıksal bağlantı tanıtımı ve bir TPN tanıtımı yaratın.

TPN profilinde, yürütülebilir dosyanın tam yolunu ve İşlem Programı adını girin:

```
Full path to TPN executable MQ_INSTALLATION_PATH/bin/amqcrs6a
Transaction Program name    recv
User ID                      0
```

MQ_INSTALLATION_PATH , WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder.

Kullanıcı kimliğini ayarlayabileceğiniz sistemlerde, mqm grubunun üyesi olan bir kullanıcı belirtin. AIX, Solaris ve HP-UX üzerinde APPCTPN (hareket adı) ve APPCLLU (yerel LU adı) ortam değişkenlerini ayarlayın (çağrılan hareket programı için yapılandırma panolarını kullanabilirsiniz).

Varsayılan kuyruk yöneticisi dışında bir kuyruk yöneticisi kullanmanız gerekebilir. Bu durumda, çağrılan bir komut dosyası tanımlayın:

```
amqcrs6a -m Queue_Man_Name
```

daha sonra komut dosyasını çağırın.

İstemci ile sunucu arasındaki bağlantıların yapılandırılması

WebSphere MQ MQI istemcileri ve sunucuları arasındaki iletişim bağlantılarını yapılandırmak için iletişim protokolünüze karar verin, bağlantının her iki ucundaki bağlantıları tanımlayın, bir dinleyici başlatın ve kanalları tanımlayın.

WebSphere MQ' da nesnelere arasındaki mantıksal iletişim bağlantılarına *kanal* denir. WebSphere MQ MQI istemcilerini sunuculara bağlamak için kullanılan kanallara MQI kanalları adı verilir. WebSphere MQ MQI istemcisindeki WebSphere MQ uygulamanızın sunucudaki kuyruk yöneticisiyle iletişim kurulabilmesi için, bağlantınızın her bir ucunda kanal tanımlamaları ayarlayabilirsiniz. Bunun nasıl gerçekleştirileceğine ilişkin ayrıntılı açıklamalar için [Kullanıcı tanımlı kanallarkonusuna](#) bakın.

MQI kanallarınızı tanımlamadan önce aşağıdakileri yapmanız gerekir:

1. Kullanacağınız iletişim biçimine karar verin. Bkz. “Kullanılacak iletişim tipi” sayfa 97.
2. Kanalın her bir ucunda bağlantıyı tanımlayın:

Bağlantıyı tanımlamak için aşağıdakileri yapmanız gerekir:

- Bağlantıyı yapılandırın.
- Kanal tanımlamaları için gerek duyduğunuz parametrelerin değerlerini kaydedin.
- Sunucunun, bir *dinleyici* başlatarak WebSphere MQ MQI istemcinizden gelen ağ isteklerini saptamasını sağlayın.

Kullanılacak iletişim tipi

Farklı platformlar farklı iletim protokollerini destekler. İletim protokolü seçiminiz, WebSphere MQ MQI istemcisi ve sunucu altyapılarının birleşimine bağlıdır.

İstemci ve sunucu platformlarınıza bağlı olarak MQI kanalları için en çok dört tip iletim protokolü vardır:

- LU 6.2
- NetBIOS
- SPX.
- TCP/IP

MQI kanallarınızı tanımladığınızda, her kanal tanımlaması bir iletim protokolü (iletim tipi) özneliği belirtmelidir. Bir sunucu tek bir protokolle sınırlı değildir, bu nedenle farklı kanal tanımları farklı protokoller belirtebilir. WebSphere MQ MQI istemcileri için, farklı iletim protokolleri kullanan alternatif MQI kanallarının olması yararlı olabilir.

İletim protokolü seçiminiz, WebSphere MQ MQI istemcisi ve sunucu altyapılarının belirli bir birleşimiyle kısıtlanmış olabilir. Olası birleşimler aşağıdaki çizelgede gösterilir.

İletim protokolü	WebSphere MQ MQI istemcisi	WebSphere MQ sunucusu
TCP/IP	UNIX sistemleri Pencereler	UNIX sistemleri Pencereler z/OS
LU 6.2	UNIX sistemleri ¹ Pencereler	UNIX sistemleri ¹ Pencereler
NetBIOS	Pencereler	Pencereler
SPX.	Pencereler	Pencereler

Not:
1. Linux Power Systems hariç

Farklı bağlantı tipleri ayarlama hakkında daha fazla bilgi için aşağıdaki bağlantılara bakın:

- [“Windowsüzerinde TCP bağlantısı tanımlama” sayfa 82](#)
- [“UNIX and Linuxüzerinde TCP bağlantısı tanımlama” sayfa 91](#)
- [“TCP/IP bağlantı sınırları” sayfa 99](#)
- [“Windowsüzerinde LU 6.2 bağlantısı tanımlanması” sayfa 84](#)
- [“UNIX and Linuxüzerinde LU 6.2 bağlantısının tanımlanması” sayfa 95](#)
- [“Windowsüzerinde NetBIOS bağlantısının tanımlanması” sayfa 85](#)
- [“Windowsüzerinde SPX bağlantısı tanımlama” sayfa 88](#)

İlgili kavramlar

[“Genişletilmiş bir işlemsel istemci yapılandırılması” sayfa 100](#)

Bu konu grubunda, işlem yöneticisi her bir kategori için genişletilmiş işlemsel işlevin nasıl yapılandırılacağı açıklanmaktadır.

[“MQI kanallarının tanımlanması” sayfa 109](#)

Yeni bir kanal yaratmak için, aynı kanal adını ve uyumlu kanal tiplerini kullanarak, bağlantının her bir ucu için bir tane olmak üzere **iki** kanal tanımlaması yaratmanız gerekir. Bu durumda, kanal tipleri *sunucu-bağlantısı* ve *istemci-bağlantı* tipleridir.

“Farklı altyapılarda sunucu bağlantısı ve istemci-bağlantı tanımlamaları yaratılıyor” sayfa 111

Geçerli olduğu bilgisayarda her bir kanal tanımlaması oluşturabilirsiniz. Bir istemci bilgisayarında kanal tanımlarının nasıl yaratılabileceğiyle ilgili kısıtlamalar vardır.

“Sunucuda sunucu bağlantısı ve istemci-bağlantı tanımları yaratılması” sayfa 114

Sunucuda her iki tanımlamayı da yaratabilir ve istemci bağlantısı tanımlamasını istemci tarafından kullanılabilir duruma getirebilirsiniz.

“MQI kanallarına ilişkin kanal-çıkış programları” sayfa 118

UNIX, Linux ve Pencere sistemlerinde, WebSphere MQ MQI istemci ortamına üç tip kanal çıkışı vardır.

“İstemcinin kuyruk paylaşım grubuna bağlanması” sayfa 122

Bir istemci ile kuyruk paylaşım grubunun üyesi olan bir sunucuda kuyruk yöneticisi arasında bir MQI kanalı yaratarak, bir istemciyi kuyruk paylaşım grubuna bağlayabilirsiniz.

“Yapılandırma dosyası kullanarak istemci yapılandırılması” sayfa 124

Bir metin dosyasındaki öznitelikleri kullanarak istemcilerinizi yapılandırın. Bu öznitelikler, ortam değişkenleriyle ya da altyapıya özgü diğer yollarla geçersiz kılınabilir.

İlgili görevler

IBM MQ MQI istemci uygulamalarının kuyruk yöneticilerine bağlanması

İlgili başvurular

CHLAUTH 'U GÖRÜNTÜLE

CHLAUTH AYARLA

Kullanılacak iletişim tipi

Farklı platformlar farklı iletişim protokollerini destekler. Aktarım iletişim kuralı seçiminiz, WebSphere MQ MQI istemcisi ve sunucu platformlarının birleşimine bağlıdır.

Farklı platformlarda MQI kanalları için dört tip iletişim vardır:

- LU 6.2
- NetBIOS
- SPX
- TCP/IP

MQI kanallarınızı tanımladığınızda, her kanal tanımının bir iletim protokolü (iletim tipi) özneliği belirtmesi gerekir. Bir sunucu tek bir protokolle sınırlı değildir, bu nedenle farklı kanal tanımları farklı protokoller belirleyebilir. WebSphere MQ MQI istemcileri için, farklı iletim protokolleri kullanan diğer MQI kanallarına sahip olmanız yararlı olabilir.

İletim protokolü seçiminiz, WebSphere MQ istemcisi ve sunucu altyapılarının belirli bir bileşimine de bağlıdır. Olası birleşimler aşağıdaki çizelgede gösterilir.

İletim protokolü	WebSphere MQ MQI istemcisi	WebSphere MQ sunucusu
TCP/IP	UNIX sistemleri Pencereleler	UNIX sistemleri Pencereleler
LU 6.2	UNIX sistemleri ¹ Pencereleler	UNIX sistemleri ¹ Pencereleler
NetBIOS	Pencereleler	Pencereleler

Çizelge 14. İletim protokolleri- WebSphere MQ istemcisi ve sunucu platformlarının birleşimi (devamı var)

İletim protokolü	WebSphere MQ MQI istemcisi	WebSphere MQ sunucusu
SPX	Pencereler	Pencereler
Not: 1. Linux dışında (POWER altyapısı)		

İlgili kavramlar

“Windowsüzerinde TCP bağlantısı tanımlama” sayfa 82

Gönderilen uçta bir kanal yapılandırarak hedefin adresini belirtmek için ve alıcı uçta bir dinleyici programını çalıştırarak bir TCP bağlantısı tanımlayın.

“UNIX and Linuxüzerinde TCP bağlantısı tanımlama” sayfa 91

Gönderme bitişindeki kanal tanımlaması hedefin adresini belirtir. İletişimci ya da inet cini, alma uçta bağlantı için yapılandırıldı.

“ Windowsüzerinde LU 6.2 bağlantısı tanımlanması” sayfa 84

SNA, iki makine arasında bir LU 6.2 etkileşimi kurulabilecek şekilde yapılandırılmalıdır.

“UNIX and Linuxüzerinde LU 6.2 bağlantısının tanımlanması” sayfa 95

SNA, iki makine arasında bir LU 6.2 etkileşimi kurulabilecek şekilde yapılandırılmalıdır.

“Windowsüzerinde NetBIOS bağlantısının tanımlanması” sayfa 85

WebSphere MQ , başka bir WebSphere MQ ürünü için NetBIOS bağlantısı kurarken üç tip NetBIOS kaynağı kullanır: oturumlar, komutlar ve adlar. Bu kaynakların her birinin bir sınırı vardır; bu sınır, varsayılan olarak ya da NetBIOSkuruluşu sırasında seçim yoluyla kurulur.

“Windowsüzerinde SPX bağlantısı tanımlama” sayfa 88

Bir SPX bağlantısı yalnızca, Windows XP ve Windows 2003 Server çalıştıran bir istemci ve sunucu için geçerlidir.

İlgili başvurular

“TCP/IP bağlantı sınırları” sayfa 99

Tek bir TCP/IP kapısında kuyruğa yollanabilen, bekleyen bağlantı isteklerinin sayısı altyapıya bağlıdır. Sınırı ulaşıldığında bir hata oluşur.

TCP/IP Bağlantısının Tanımlanması

İstemci uçtaki kanal tanımlamasındaki TCP iletim tipinin belirtilmesi Sunucuda bir dinleyici programı başlatın.

Kanal tanımında TCP iletim tipini belirterek istemcide bir TCP/IP bağlantısı belirtin.

Kanal programlarının alınması, gönderme kanalından başlatılan başlatma isteğine yanıt olarak başlatılır. Bunu yapmak için, gelen ağ isteklerini algılamaya ve ilişkili kanalı başlatmaya yönelik bir dinleyici programı başlatılmalıdır. Bir dinleyici programını başlatma yordamı sunucu platformuna bağlıdır.

İstemci ve sunucu platformlarınız için ilgili konulara bakın.

TCP/IP bağlantı sınırları

Tek bir TCP/IP kapısında kuyruğa yollanabilen, bekleyen bağlantı isteklerinin sayısı altyapıya bağlıdır. Sınırı ulaşıldığında bir hata oluşur.

Bu bağlantı sınırı, bir IBM WebSphere MQ sunucusuna bağlanabileceğiniz istemci sayısı üst sınığıyla aynı değildir. Sunucu sistem kaynakları tarafından belirlenen düzeye kadar, daha fazla istemci sunucuya bağlanabilirsiniz. Bağlantı isteklerine ilişkin birikim değerleri aşağıdaki çizelgede gösterilir:

Çizelge 15. TCP/IP kapısında kuyruğa alınan bekleyen bağlantı isteği sayısı üst sınırı	
Sunucu platformu	Bağlantı isteği sayısı üst sınırı
AIX	100
HP-UX	20
Linux	100
IBM	255
Solaris	100
Windows Sunucusu	100
Windows İş İstasyonu	100
z/OS	255

If the connection limit is reached, the client receives a return code of MQRC_HOST_NOT_AVAILABLE from the MQCONN call, and an AMQ9202 error in the client error log (/var/mqm/errors/AMQERR0n.LOG on UNIX and Linux systems or amqerr0n.log in the errors subdirectory of the IBM WebSphere MQ client installation on Windows). İstemci MQCONN isteğini yeniden deneme girişiminde bulduysa, başarılı olabilir.

Yapabildiğiniz bağlantı isteği sayısını artırmak ve bu sınırlama tarafından oluşturulan hata iletilerini önlemek için, her biri farklı bir kapıda dinlerken birden çok dinleyiciye sahip olabilir ya da birden çok kuyruk yöneticisi olabilir.

NetBIOS ya da SPX bağlantısı tanımlanması

NetBIOS ve SPX bağlantıları yalnızca Windows sistemleri için geçerlidir.

Bir NetBIOS bağlantısı yalnızca Pencereleçalıştıran bir istemci ve sunucu için geçerlidir. Bkz. [NetBIOS bağlantısının tanımlanması](#).

Bir SPX bağlantısı yalnızca, Windows XP ya da Windows 2003 Server çalıştıran bir istemci ve sunucu için geçerlidir. Bkz. [SPX bağlantısının tanımlanması](#).

Genişletilmiş bir işlemsel istemci yapılandırılması

Bu konu grubunda, işlem yöneticisi her bir kategori için genişletilmiş işlemsel işlevin nasıl yapılandırılacağı açıklanmaktadır.

Genişletilmiş işlemsel istemci, her platform için aşağıdaki dış hareket yöneticilerine destek sağlar:

XA uyumlu hareket yöneticileri

Genişletilmiş işlemsel istemci, XA kaynak yöneticisi arabirimini, CICS ve Tuxedo gibi XA uyumlu hareket yöneticilerini desteklemek için sağlar.

Microsoft Transaction Server (yalnızca Windows sistemleri)

Yalnızca Windows sistemlerinde, XA kaynak yöneticisi arabirimi Microsoft Transaction Server (MTS) olanağını da destekler. Genişletilmiş işlemsel istemciyle sağlanan WebSphere MQ MTS desteği, MTS ile XA kaynak yöneticisi arabirimi arasındaki köprüyü sağlar.

WebSphere Application Server

Earlier versions of WebSphere MQ supported WebSphere Application Server Version 4 or Version 5, and required you to carry out certain configuration tasks to use the extended transactional client. WebSphere Application Server Sürüm 6 ve üstü, bir WebSphere MQ ileti alışverişi sağlayıcısını içerir; bu nedenle, genişletilmiş işlemsel istemciyi kullanmanız gerekmez.

İlgili kavramlar

[“XA uyumlu hareket yöneticilerinin yapılandırılması” sayfa 101](#)

Önce, WebSphere MQ temel istemcisini yapılandırın ve daha sonra, bu konulardaki bilgileri kullanarak genişletilmiş işlemsel işlevi yapılandırın.

“ Microsoft Transaction Server” sayfa 108

MTS ' yi hareket yöneticisi olarak kullanabilmeniz için ek yapılandırma gerekmez. Ancak, notların bazı noktalar vardır.

XA uyumlu hareket yöneticilerinin yapılandırılması

Önce, WebSphere MQ temel istemcisini yapılandırın ve daha sonra, bu konulardaki bilgileri kullanarak genişletilmiş işlemsel işlevi yapılandırın.

Not: Bu kısımda, *Distributed Transaction Processing: The XA Specification*(Dağıtılmış İşlem İşleme: XA Belirtimi) altında, Open Group (Açık Grup) tarafından yayınlanan XA arabirimi hakkında temel bir kavrayı

To configure an extended transactional client, you must first configure the WebSphere MQ base client as described in [IBM WebSphere MQ istemcisi kurulması](#) . Using the information in this section, you can then configure the extended transactional function for an XA-compliant transaction manager such as CICS and Tuxedo.

Bir hareket yöneticisi, kuyruk yöneticisine bağlı olan istemci uygulaması tarafından kullanılan MQI kanalını kullanarak, bir kuyruk yöneticisiyle kaynak yöneticisi olarak iletişim kurar. Hareket yöneticisi bir kaynak yöneticisi (xa_) işlev çağrısını gönderdiğinde, çağrıyı kuyruk yöneticisine iletmek ve çıkışı kuyruk yöneticisinden geri almak için MQI kanalı kullanılır.

Kuyruk yöneticisini kaynak yöneticisi olarak açmak için xa_open çağrısı yayınlayarak ya da istemci uygulaması bir MQCONN ya da MQCONNX çağrısı yayınlayarak, MQI kanalını başlatabilir ya da istemci uygulaması MQI kanalını başlatabilir.

- Hareket yöneticisi MQI kanalını başladiysa ve istemci uygulaması daha sonra aynı iş parçacığıdaki MQCONN ya da MQCONNX çağrılarını çağırıyorsa, MQCONN ya da MQCONNX çağrısı başarıyla tamamlanır ve uygulamaya bir bağlantı tanıtıcısı döndürülür. Uygulama, MQRC_ALREADY_CONNECTED neden koduyla bir MQCC_UYARI tamamlanma kodu almamaktadır.
- İstemci uygulaması MQI kanalını başladiysa ve hareket yöneticisi daha sonra aynı iş parçacığıdaki xa_open çağrılarını çağırıyorsa, xa_open çağrısı MQI kanalını kullanarak kuyruk yöneticisine iletilir.

Bir hatanın ardından kurtarma durumunda, hiçbir istemci uygulaması çalışmadığında, hareket yöneticisi, hata sırasında kuyruk yöneticisinin katıldığı tamamlanmamış iş birimlerini kurtarmak için özel olarak ayrılmış bir MQI kanalı kullanabilir.

XA uyumlu bir hareket yöneticisiyle genişletilmiş bir işlemsel istemci kullanırken aşağıdaki koşulları göz önünde bulundurun:

- Tek bir iş parçacığıda, bir istemci uygulaması aynı anda yalnızca bir kuyruk yöneticisine bağlanabilir. Bu kısıtlama yalnızca genişletilmiş bir işlemsel istemci kullanılırken geçerlidir; bir WebSphere MQ temel istemcisi kullanan bir istemci uygulaması, tek bir iş parçacığı içinde koşut zamanlı olarak birden çok kuyruk yöneticisine bağlanabilir.
- İstemci uygulamasının her bir iş parçacığı, farklı bir kuyruk yöneticisine bağlanabilir.
- Bir istemci uygulaması paylaşılan bağlantı tanıtıcılarını kullanamaz.

Genişletilmiş işlemsel işlevi yapılandırmak için, kaynak yöneticisi olarak işlev gören her kuyruk yöneticisi için işlem yöneticisine aşağıdaki bilgileri sağlamanız gerekir:

- Bir xa_open dizgisi
- Bir XA anahtar yapısına ilişkin gösterge

When the transaction manager calls xa_open to open the queue manager as a resource manager, it passes the xa_open string to the extended transactional client as the argument, xa_bilgisi, on the call. Genişletilmiş işlemsel istemci, xa_open dizgisindeki bilgileri aşağıdaki şekillerde kullanır:

- Bir MQI kanalını sunucu kuyruk yöneticisine başlatmak için, istemci uygulaması önceden başlatılmamış olmalıdır.
- İşlem yöneticisinin kaynak yöneticisi olarak açıldığı kuyruk yöneticisinin istemci uygulamasının bağlandığı kuyruk yöneticisiyle aynı olduğunu denetlemek için

- Kuyruk yöneticisi dinamik kayıt kullanıyorsa, hareket yöneticisinin ax_reg ve ax_unreg işlevlerini bulmak için

Xa_open dizesinin biçimi ve xa_open dizesindeki bilgilerin genişletilmiş bir işlemsel istemci tarafından nasıl kullanıldığı hakkında daha fazla ayrıntı için bkz. [“xa_open dizgisinin biçimi” sayfa 102.](#)

XA anahtar yapısı, hareket yöneticisinin genişletilmiş işlemsel istemci tarafından sağlanan xa_ işlevlerini bulmasını sağlar ve kuyruk yöneticisinin dinamik kayıt kullanıp kullanmadığını belirler. Genişletilmiş işlemsel istemciyle sağlanan XA anahtar yapılarına ilişkin bilgi için bkz. [“XA anahtar yapıları” sayfa 105.](#)

Belirli bir hareket yöneticisi için genişletilmiş işlemsel işlevin nasıl yapılandırılacağı ve hareket yöneticisinin genişletilmiş bir işlemsel istemciyle kullanılmasına ilişkin diğer bilgiler için aşağıdaki kısımlara bakın:

- [“Configuring an extended transactional client for CICS” sayfa 107](#)
- [“Tuxedo için genişletilmiş bir işlemsel istemci yapılandırılması” sayfa 108](#)

İlgili kavramlar

[“xa_open dizesinin CHANNEL, TRPTYPE, CONNAME ve QMNAME parametreleri” sayfa 104](#)
Genişletilmiş işlemsel istemcinin, bağlantı kurmak üzere kuyruk yöneticisini belirlemek için bu parametreleri nasıl kullandığını anlamak için bu bilgileri kullanın.

[“xa_open için ek hata işleme” sayfa 105](#)
xa_open çağrısı bazı durumlarda başarısız olur.

İlgili görevler

[“Genişletilmiş işlemsel istemciyi SSL kanallarıyla kullanma” sayfa 106](#)
xa_open dizesini kullanarak bir SSL kanalı ayarlayamazsınız. İstemci kanal tanımlama çizelgesini (ccdt) kullanmak için bu yönergeleri izleyin.

İlgili başvurular

[“TPM ve AXLIB parametreleri” sayfa 105](#)

Genişletilmiş bir işlemsel istemci, hareket yöneticisinin ax_reg ve ax_unreg işlevlerini bulmak için TPM ve AXLIB parametrelerini kullanır. Bu işlevler, yalnızca kuyruk yöneticisi dinamik kayıt kullanıyorsa kullanılır.

[“Genişletilmiş işlemsel işleme hatasının ardından kurtarma” sayfa 105](#)

Bir hata sonrasında, işlem yöneticisi eksik iş birimlerini kurtarabilmelidir. Bunu yapmak için, hareket yöneticisinin, hata sırasında tamamlanmamış bir iş birimine katılan herhangi bir kuyruk yöneticisi kaynak yöneticisi olarak açılabilmesi gerekir.

xa_open dizgisinin biçimi

Bir xa_open dizgisi, tanımlı parametre adları ve değerleri çiftleri içerir.

Bir xa_open dizgisi şu biçimde olur:

```
parm_name1=parm_value1,parm_name2=parm_value2, ...
```

Burada *parm_name* bir değiştirgenin adı ve *parm_value* bir parametrenin değeridir. Değiştirgelerin adları büyük ve küçük harfe duyarlı değildir; ancak, tersi belirtilmedikçe, parametrelerin değerleri büyük ve küçük harfe duyarlıdır. Parametreleri herhangi bir sırada belirleyebilirsiniz.

Parametrelerin adları, anlamları ve geçerli değerleri aşağıda verilmiştir:

Ad

Anlam ve geçerli değerler

Kanal

Bir MQI kanalının adı.

İsteğe bağlı bir değiştirgendir. Bu parametre sağlanırsa, CONNAME parametresi de sağlanmalıdır.

TRPTYPE

MQI kanalına ilişkin iletişim protokolü. Aşağıdakiler geçerli değerlerdir:

LU62

SNA LU 6.2

NETBIOS

NetBIOS

SPX

IPX/SPX

TCP

TCP/IP

İsteğe bağlı bir deęiřtirgedir. Atlanırsa, varsayılan deęer olan TCP ' nin varsayılan deęeri kullanılır. Parametrenin deęerleri büyük ve küçük harfe duyarlı deęildir.

ADı

MQI kanalının sunucu ucunda yer alan kuyruk yöneticisinin aę adresi. Bu parametrenin geęerli deęerleri, TRPTYPE parametresinin deęerine baęlıdır:

LU62

CPI-C yan bilgi giriřini tanıtan simgesel hedef adı.

Bir ortak LU ' nun aę nitelenmiř adı geęerli bir deęer deęil, ortak LU dięeradı da deęil. Bunun nedeni, hareket programı (TP) adını ve kip adını belirtmek için ek parametre olmamasıdır.

NETBIOS

NetBIOS adı.

SPX

4 baytlık bir aę adresi, 6 baytlık düęüm adresi ve isteğe baęlı 2 baytlık yuva numarası. Bu deęerlerin onaltılı gösterimde belirtilmesi gerekir. Bir nokta, aę ve düęüm adreslerini ayırmalıdır ve yuva numarası (belirtildiyse) parantez içine alınmalıdır. Örneęin:

```
0a0b0c0d.804abcde23a1(5e86)
```

Yuva numarası atlanırsa, varsayılan deęer olan 5e86 ' nın deęeri kullanılır.

TCP

Bir anasistem adı ya da bir IP adresi, isteğe baęlı olarak parantez içinde bir kapı numarası izler. Kapı numarası atlanırsa, varsayılan deęer olarak 1414 varsayılan deęeri kullanılır.

İsteğe baęlı bir deęiřtirgedir. Bu parametre saęlanırsa, KANAL parametresi de saęlanmalıdır.

QMNAME

MQI kanalının sunucu ucunda bulunan kuyruk yöneticisinin adı. Ad boş bırakılamaz ya da tek bir yıldız işareti (*) ya da bir yıldız işaretiyle bařlayan ad da olamaz. Bu, parametrenin, ada göre belirli bir kuyruk yöneticisini tanımlaması gerektięi anlamına gelir.

Bu zorunlu bir parametredir.

Bir istemci uygulaması belirli bir kuyruk yöneticisine baęlandığında, herhangi bir işlem kurtarma işlemi aynı kuyruk yöneticisi tarafından işlenmelidir.

Uygulama bir z/OS kuyruk yöneticisine baęlanıyorsa, uygulama belirli bir kuyruk yöneticisinin adını ya da bir kuyruk paylařım grubu (QSG) adını belirtebilir. Kuyruk yöneticisi adı ya da QSG adı kullanılarak, uygulama, bir QMGR kurtarma birimiyle ya da bir GROUP birimi kurtarma atma birimiyle bir harekette olup olmadıęını denetler. GROUP birimi kurtarma yok etme birimi, işlemin kurtarılmasına, QSG ' nin herhangi bir üyesinde işlenmesini saęlar. Kurtarma grubu birimlerini kullanmak için **GROUPUR** kuyruk yöneticisi öznitelięinin etkinleřtirilmiř olması gerekir.

TPM

Kullanılmakta olan hareket yöneticisi. Geęerli deęerler CICS ve SMOKIN ' dir.

Geniřletilmiř bir işlemsel istemci, aynı amaç için bu parametreyi ve AXLIB parametresini kullanır. Bu parametrelere iliřkin daha fazla bilgi için bkz. [TPM ve AXLIB parametreleri](#).

İsteğe baęlı bir deęiřtirgedir. Parametrenin deęerleri büyük ve küçük harfe duyarlı deęildir.

AXLIB

Hareket yöneticisinin ax_reg ve ax_unreg işlevlerini içeren kitaplığın adı.

İsteğe bağlı bir deęiřtirgedir.

Burada bir xa_open dizgisi örneęi yer alıyor:

```
channel=MARS.SVR, trptype=tcp, conname=MARS(1415), qmname=MARS, tpm=cics
```

xa_open dizesinin CHANNEL, TRPTYPE, CONNAME ve QMNAME parametreleri

Geniřletilmiş işlemsel istemcinin, baęlantı kurmak üzere kuyruk yöneticisini belirlemek için bu parametreleri nasıl kullandığını anlamak için bu bilgileri kullanın.

Kanal ve CONNAME parametreleri xa_open dizgisinde belirtilirse, genişletilmiş işlemsel istemci bu parametreleri ve bir MQI kanalını sunucu kuyruk yöneticisine başlatmak için TRPTYPEE parametresini kullanır.

Kanal ve CONNAME deęiřtirgeleri xa_open dizgisinde saęlanmazsa, genişletilmiş işlemsel istemci bir MQI kanalını başlatmak için MQSERVER ortam deęiřkeninin deęerini kullanır. MQSERVER ortam deęiřkeni tanımlı deęilse, genişletilmiş işlemsel istemci QMNAME parametresiyle tanıtılan istemci kanalı tanımlamasındaki giriři kullanır.

Bu durumların her birinde, genişletilmiş işlemsel istemci, QMNAME parametresinin deęerinin MQI kanalının sunucu ucundaki kuyruk yöneticisinin adı olup olmadığını denetler. Deęilse, xa_open çağrısı başarısız olur ve hareket yöneticisi hatayı uygulama için bildirir.

Uygulama istemcisi V7.0.1 ya da sonraki bir yayın düzeyinde z/OS kuyruk yöneticisine baęlanıyorsa, QMNAME parametresi için bir kuyruk paylaşım grubu (QSG) adı belirleyebilirler. Bu, uygulama istemcisinin bir GROUP birimi kurtarma düzeni ile bir harekete katılmasını saęlar.

Uygulama QMNAME parametresi alanında bir QSG adı kullanıyorsa ve GROUPUR özellięi, baęlandıęı kuyruk yöneticisinde devre dıřı bırakıldıysa, xa_open çağrısını başarısız olur.

Uygulama, V7.0.1' dan önceki bir sürümdeki bir kuyruk yöneticisine baęlanırsa, xa_open çağrısı başarılı olur, ancak hareketin bir QMGR kurtarma düzeni birimi vardır. GROUP birimini kurtarma yok etme birimi gerektiren uygulamaların yalnızca V7.0.1 ya da sonraki bir yayın düzeyinde kuyruk yöneticilerine baęlanmasını saęlayın.

İstemci uygulaması daha sonra, hareket yöneticisinin xa_open çağrısını vermek için kullandığı iş parçacığında MQCONN ya da MQCONNX 'i çağırıldığında, uygulama, xa_open çağrısıyla başlatılan MQI kanalı için bir baęlantı tanıtıcısı alır. İkinci bir MQI kanalı başlatılmamaktadır. Geniřletilmiş işlemsel istemci, MQCONN ya da MQCONNX çağrısındaki QMgrName deęiřtirgesinin deęerinin, MQI kanalının sunucu ucundaki kuyruk yöneticisinin adı olup olmadığını denetler. Deęilse, MQCONN ya da MQCONNX çağrısı bir MQRC_ANOTHER_Q_MGR_CONNECTED neden kodunda başarısız olur. QMgrName parametresinin deęeri boşluk ya da tek bir yıldız işareti (*) ise ya da yıldız işaretiyle başladıysa, MQCONN ya da MQCONNX çağrısı bir MQRC_Q_MGR_NAME_ERROR neden kodunda başarısız olur.

İstemci uygulaması, hareket yöneticisi aynı iş parçacığadaki xa_open çağrılmadan önce MQCONN ya da MQCONNX çağrılarak bir MQI kanalı başlattıysa, hareket yöneticisi bunun yerine bu MQI kanalını kullanır. İkinci bir MQI kanalı başlatılmamaktadır. Geniřletilmiş işlemsel istemci, xa_open dizesindeki QMNAME parametresinin deęerinin, sunucu kuyruęu yöneticisinin adı olduğunu denetler. Bu durumda deęilse, xa_open çağrısı başarısız olur.

Bir istemci uygulaması önce bir MQI kanalı başlatıyorsa, MQCONN ya da MQCONNX çağrısındaki QMgrName parametresinin deęeri boş olabilir ya da tek bir yıldız (*) işareti ya da yıldız imi ile başlayabilir. Ancak bu kořullar altında, uygulamanın baęlandıęı kuyruk yöneticisinin, aynı iş parçacığında xa_open çağrıldığında, işlem yöneticisinin kaynak yöneticisi olarak açmayı planladığı kuyruk yöneticisiyle aynı olduğundan emin olmanız gerekir. QMgrName parametresinin deęeri kuyruk yöneticisini açık bir şekilde tanımlıyorsa, bu nedenle daha az sorunla karřılařabilirsiniz.

TPM ve AXLIB parametreleri

Geniřletilmiş bir işlemsel istemci, hareket yöneticisinin ax_reg ve ax_unreg işlevlerini bulmak için TPM ve AXLIB parametrelerini kullanır. Bu işlevler, yalnızca kuyruk yöneticisi dinamik kayıt kullanıyorsa kullanılır.

TPM parametresi bir xa_open dizgisinde belirtilirse, ancak AXLIB parametresi sağlanmazsa, genişletilmiş işlemsel istemci, TPM parametresinin değerine dayalı olarak AXLIB parametresi için bir değer varsayar. AXLIB parametresinin varsayılan değerleri için bkz. Çizelge 16 sayfa 105 .

<i>Çizelge 16. AXLIB parametresinin varsayılan değerleri</i>		
TPM ' nin değeri	Altyapı	AXLIB ' nin üstlendiği değer
CICS	AIX	/usr/lpp/encina/lib/libEncServer.a(EncServer_shr.o)
CICS	HP-UX	/opt/encina/lib/libEncServer.sl
CICS	Solaris	/opt/encina/lib/libEncServer.so
CICS	Windows sistemleri	libEncSunucusu
Smokin	AIX	/usr/lpp/tuxedo/lib/libtux.a(libtux.so.60)
Smokin	HP-UX	/opt/tuxedo/lib/libtux.sl
Smokin	Solaris	/opt/tuxedo/lib/libtux.so.60
Smokin	Windows sistemleri	libtux

AXLIB parametresi bir xa_open dizgisinde verilirse, genişletilmiş işlemsel istemci, TPM parametresinin değerine dayalı olarak kabul edilen değeri geçersiz kılmak için değerini kullanır. AXLIB parametresi, TPM parametresinin belirtilen bir değeri olmayan bir hareket yöneticisi için de kullanılabilir.

xa_open için ek hata işleme

xa_open çağırısı bazı durumlarda başarısız olur.

Bu bölümdeki konular, xa_open çağırısının başarısız olduğu durumları açıklar. Ayrıca, aşağıdaki durumlardan biri ortaya çıkarsa da başarısız olur:

- xa_open dizgisinde hatalar var.
- Bir MQI kanalını başlatmak için yeterli bilgi yok.
- Bir MQI kanalı başlatma girişimi sırasında bir sorun oluştu (örneğin, sunucu kuyruk yöneticisi çalışmıyor).

Geniřletilmiş işlemsel işleme hatasının ardından kurtarma

Bir hata sonrasında, işlem yöneticisi eksik iş birimlerini kurtarabilmelidir. Bunu yapmak için, hareket yöneticisinin, hata sırasında tamamlanmamış bir iş birimine katılan herhangi bir kuyruk yöneticisi kaynak yöneticisi olarak açılabilmesi gerekir.

Herhangi bir yapılandırma bilgisini değiřtirmeniz gerekirse, değiřiklikleri yapmadan önce tüm eksik iş birimlerinin çözülmüş olduğundan emin olmalısınız. Diđer bir seçenek olarak, yapılandırma değiřikliklerinin, hareket yöneticisinin açılması gereken kuyruk yöneticilerini açabilme yeteneğini etkilemediğinden emin olmanız gerekir. Aşağıda, bu tür yapılandırma değiřikliklerine ilişkin örnekler yer alıyor:

- Xa_open dizgisinin içeriğini değiřtirme
- MQSERVER ortam değışkeninin değerini değiřtirme
- İstemci kanal tanımlama çizelgesindeki girişleri değiřtirme (CCDT)
- Sunucu Bařlantı Kanal Tanımlarının Silinmesi

XA anahtar yapıları

İki XA anahtar yapısı, her altyapıda genişletilmiş işlemsel istemciyle birlikte sağlanır.

Bu anahtar yapıları şunlardır:

MQRMIASwitch

Bu anahtar yapısı, bir kuyruk yöneticisi kaynak yöneticisi olarak işlev görerek dinamik kayıt kullanmıyorsa, bir hareket yöneticisi tarafından kullanılır.

MQRMIASwitchDynamic

Bu anahtar yapısı, bir kuyruk yöneticisi kaynak yöneticisi olarak hareket ederken, dinamik kaydı kullandığında bir hareket yöneticisi tarafından kullanılır.

Bu anahtar yapıları, Çizelge 17 sayfa 106 içinde gösterilen kitaplıklarda yer alır.

<i>Çizelge 17. XA anahtar yapılarını içeren WebSphere MQ kitaplıkları</i>	
Altyapı	XA anahtar yapılarını içeren kitaplık
AIX HP-UX Linux Solaris	<code>MQ_INSTALLATION_PATH/lib/libmqcxa</code>
Windows sistemleri	<code>MQ_INSTALLATION_PATH\bin\mqcxa.dll</code> ¹
<i>MQ_INSTALLATION_PATH</i> , WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder.	

Her anahtar yapısındaki WebSphere MQ kaynak yöneticisi adı MQSeries_XA_RMI, ancak birçok kuyruk yöneticisi aynı anahtar yapısını paylaşabilir.

İlgili kavramlar

“Dinamik kayıt ve genişletilmiş işlemsel işleme” sayfa 106

Dinamik kayıt işlevinin kullanılması, işlem yöneticisi tarafından yayınlanan xa _ işlev çağrılarının sayısını azaltabileceği için bir eniyileme biçimidir.

Dinamik kayıt ve genişletilmiş işlemsel işleme

Dinamik kayıt işlevinin kullanılması, işlem yöneticisi tarafından yayınlanan xa _ işlev çağrılarının sayısını azaltabileceği için bir eniyileme biçimidir.

Bir kuyruk yöneticisi dinamik kayıt kullanmıyorsa, bir hareket yöneticisi iş biriminin her biriminde kuyruk yöneticisini içerir. Kuyruk yöneticisinin iş birimi içinde güncellenen herhangi bir kaynağı olmasa da, hareket yöneticisi xa_start, xa_end ve xa_prepare çağrılarak bunu yapar.

Bir kuyruk yöneticisi dinamik kayıt kullanıyorsa, bir hareket yöneticisi kuyruk yöneticisinin bir iş biriminde yer almadığını varsayarak başlar ve xa_start işlevini çağırır. Daha sonra kuyruk yöneticisi, yalnızca kaynakları eşitleme noktası denetimi içinde güncellendiyse çalışma birimine dahil olur. Bu durum ortaya çıkarsa, genişletilmiş işlemsel istemci, kuyruk yöneticisinin katılıma kaydını kaydettirmek için ax_reg 'i çağırır.

Genişletilmiş işlemsel istemciyi SSL kanallarıyla kullanma

xa_open dizesini kullanarak bir SSL kanalı ayarlayamazsınız. İstemci kanal tanımlama çizelgesini (ccdt) kullanmak için bu yönergeleri izleyin.

Bu görev hakkında

xa_open xa_info dizisinin sınırlı boyutu nedeniyle, bir kuyruk yöneticisine bağlanmak için xa_open string yöntemini kullanarak bir SSL kanalı ayarlamak için gereken tüm bilgileri iletmeniz mümkün değildir. Bu nedenle, istemci kanal tanımlama çizelgesini kullanmanız ya da hareket yöneticiniz izin veriyorsa, xa_open çağrısını vermeden önce, kanalı MQCONNX ile yaratmalısınız.

İstemci kanal tanımlama çizelgesini kullanmak için aşağıdaki adımları izleyin:

Yordam

1. Yalnızca zorunlu qmname (kuyruk yöneticisi adı) parametresini içeren bir xa_open dizgisi belirtin, örneğin: XA_Open_String=qmname=MYQM
2. Gerekli SSL parametreleriyle bir CLNTCONN (istemci-bağlantı) kanalı tanımlamak için kuyruk yöneticisi kullanın. CLNTCONN tanımlamasındaki QMNAME özniteliğe kuyruk yöneticisi adını ekleyin. Bu, xa_open dizgisindeki qmname ile eşleştirilir.
3. CLNTCONN tanımlamasını, etkin dizinde istemci kanalı tanımlama çizelgesinde (CCDT) ya da Windows' da bulunan istemci sistemi için kullanılabilir yapın.
4. CCDT kullanıyorsanız, MQCHLLIB ortam değişkenlerini ve MQCHLTAB ortam değişkenlerini kullanarak CLNTCONN kanalının tanımlamasını içeren CCDT ' yi saptayın. Bu değişkenleri hem istemci uygulaması, hem de hareket yöneticisi tarafından kullanılan ortamlarda ayarlayın.

Sonuçlar

Bu, hareket yöneticisine, doğru kimlik doğrulaması yapmak için gerekli olan SSL öznitelikleriyle uygun kuyruk yöneticisine bir kanal tanımlaması verir (SSLCIPH, CipherSpecde dahil olmak üzere).

Configuring an extended transactional client for CICS

You configure an extended transactional client for use by CICS by adding an XAD resource definition to a CICS region.

Add the XAD resource definition by using the CICS resource definition online (RDO) command, **cicsadd**. XAD kaynak tanımlaması aşağıdaki bilgileri belirtir:

- Bir xa_open dizgisi
- Anahtar yükleme dosyasının nitelenmiş açık yol adı

Aşağıdaki platformların her birinde CICS tarafından kullanılmak üzere bir anahtar yükleme dosyası sağlanır: AIX, HP-UX, Solaris ve Windows sistemleri. Her anahtar yükleme dosyası, dinamik kayıt için kullanılan XA anahtar yapısına ilişkin bir gösterge döndüren bir işlev içerir, MQRMIXASwitchDynamic. Her bir anahtar yükleme dosyasının tam olarak nitelenmiş yol adı için bkz. [Çizelge 18 sayfa 107](#) .

Çizelge 18. Anahtar yükleme dosyaları	
Altyapı	Anahtar yükleme dosyası
AIX HP-UX Linux Solaris	MQ_INSTALLATION_PATH/lib/amqczsc
Windows sistemleri	MQ_INSTALLATION_PATH\bin\mqcc4swi.dll ¹
MQ_INSTALLATION_PATH , WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder.	

Aşağıda, Windows sistemleri için bir XAD kaynak tanımlaması örneği yer alıyor:

```
cicsadd -c xad -r REGION1 WMQXA \  
ResourceDescription="WebSphere MQ queue manager MARS" \  
XAOpen="channel=MARS.SVR,trptype=tcp,connname=MARS(1415),qmname=MARS,tpm=cics" \  
SwitchLoadFile="C:\Program Files\IBM\WebSphere MQ\bin\mqcc4swi.dll"
```

For more information about adding an XAD resource definition to a CICS region, see the *CICS Yönetim Başvurusu* and the *CICS Yönetim Kılavuzu* for your platform.

CICS kullanımıyla ilgili genişletilmiş işlemsel istemciyle ilgili aşağıdaki bilgileri not edin:

- You can add only one XAD resource definition for WebSphere MQ to a CICS region. Bu , bir bölgeyle yalnızca tek bir kuyruk yöneticisinin ilişkilendirebileceği ve bölgede çalışan tüm CICS uygulamalarının

yalnızca o kuyruk yöneticisine bağlanabileceği anlamına gelir. Farklı bir kuyruk yöneticisine bağlanan CICS uygulamalarını çalıştırmak istiyorsanız, uygulamaları farklı bir bölgede çalıştırmanız gerekir.

- Bir bölgedeki her uygulama sunucusu, başlatılırken xa_open çağrısını arar ve bir MQI kanalını, bölgeyle ilişkili kuyruk yöneticisine başlatır. Başka bir deyişle, kuyruk yöneticisinin bir uygulama sunucusu başlatılmadan önce başlatılması gerekir, tersi durumda xa_open çağrısı başarısız olur. Tüm WebSphere MQ MQI istemcisi uygulamaları daha sonra uygulama sunucusu tarafından işlenen aynı MQI kanalını kullanır.
- bir MQI kanalı başlatıldığında ve kanalın istemci ucunda bir güvenlik çıkışı yoksa, istemci sisteminden sunucu bağlantısı MCA ' sına akan kullanıcı kimliği cicsolur. Under certain circumstances, the queue manager uses this user ID for authority checks when the server connection MCA subsequently attempts to access the queue manager resources on behalf of a client application. Bu kullanıcı kimliği yetki denetimleri için kullanılıyorsa, erişim için gereken tüm kaynaklara erişim yetkisine sahip olduğundan emin olun.

Kuyruk yöneticisinin yetki denetimleri için bu kullanıcı kimliğini kullandığında ilgili bilgi edinmek için [Güvenlik](#) konusuna bakın.

- WebSphere MQ istemci sistemlerinde kullanılmak üzere sağlanan CICS görev sonlandırma çıkışları, [Çizelge 19](#) sayfa 108 içinde listelenir. Bu çıkışları, WebSphere MQ sunucusu sistemleri için karşılık gelen çıkışları yapılandırdığınız şekilde yapılandırıyorsunuz. Bu nedenle, bu bilgi için [CICS kullanıcı çıkışlarını etkinleştirme](#) ' e bakın.

Çizelge 19. CICS görev sonlandırma çıkışları		
Altyapı	Kaynak	Kitaplık
AIX HP-UX Linux Solaris	amqzscgx.c	amqzscg
Windows sistemleri	amqzscgn.c	mqcc1415.dll

Tuxedo için genişletilmiş bir işlemel istemci yapılandırılması

Tuxedo tarafından kullanılmak üzere XAD kaynak tanımlamasını yapılandırmak için UBBCONFIG dosyasını ve kaynak yöneticisi çizelgesini güncelleyin.

XAD kaynak tanımlamasını Tuxedo tarafından kullanılmak üzere yapılandırmak için aşağıdaki işlemleri gerçekleştirin:

- Bir uygulamaya ilişkin Tuxedo UBBCONFIG dosyasının GROUPS kısmında, xa_open dizgisi belirtmek için OPENINFO parametresini kullanın.

Bunu nasıl yapacağınıza ilişkin bir örnek için, Tuxedo örnek programlarıyla birlikte kullanılmak üzere sağlanan örnek UBBCONFIG dosyasına bakın. AIX, HP-UX ve Solaris üzerinde, dosyanın adı ubbstxcx.cfg ' dir ve Windows sistemlerinde dosyanın adı ubbstxcn.cfg olur.

- Tuxedo kaynak yöneticisi çizelgesinde bir kuyruk yöneticisine ilişkin girişte:
 - udataobj\RM (AIX, HP-UX ve Solaris)
 - udataobj\rm (Windows sistemleri)

Bir XA anahtar yapısının adını ve yapıyı içeren kitaplığın tam olarak nitelenmiş yol adını belirtin. Her platform için bunu nasıl yapacağınıza ilişkin bir örnek için bakınız: [SMOKIN Samples](#). Smokin, bir kaynak yöneticisinin dinamik kaydını destekler ve bu durumda MQRMIXASwitch ya da MQRMIXASwitchDynamickullanabilirsiniz.

Microsoft Transaction Server

MTS ' yi hareket yöneticisi olarak kullanabilmeniz için ek yapılandırma gerekmez. Ancak, notların bazı noktalar vardır.

MTS ' yi genişletilmiş işlemsel istemciyle kullanmaya ilişkin şu bilgileri not edin:

- MTS uygulaması, bir sunucu kuyruk yöneticisine bağlandığında her zaman bir MQI kanalı başlatır. MTS, bir hareket yöneticisi rolünde, kuyruk yöneticisiyle iletişim kurmak için aynı MQI kanalını kullanır.
- Bir başarısızlığın ardından, MTS ' nin tamamlanmamış iş birimlerini kurtarabilmesi gerekir. Bunu yapmak için, MTS ' nin hata sırasında tamamlanmamış bir iş birimine katılan herhangi bir kuyruk yöneticisiyle iletişim kurabilmesi gerekir.

MTS uygulaması bir sunucu kuyruk yöneticisine bağlandığında ve bir MQI kanalı başlattığında, genişletilmiş işlemsel istemci, gerekirse, kanal yeniden başlatılmasını etkinleştirmek için MQCONN ya da MQCONNX çağrısının değiştirgelerinden yeterli bilgileri alır. Genişletilmiş işlemsel istemci bilgileri MTS ' ye iletir ve MTS, bilgileri günlüğündeki bilgileri kaydeder.

MTS uygulaması bir MQCONN çağrısı yayınlarsa, bu bilgi yalnızca kuyruk yöneticisinin adıdır. MTS uygulaması bir MQCONNX çağrısı yayınlar ve bir kanal tanımlama yapısı (MQCD) sağlıyorsa, bu bilgiler aynı zamanda MQI kanalının adını, sunucu kuyruk yöneticisinin ağ adresini ve kanala ilişkin iletişim protokolünü de içerir.

Kurtarma durumunda, MTS bu bilgileri genişletilmiş işlemsel istemciye iletir ve genişletilmiş işlemsel istemci, MQI kanalını yeniden başlatmak için bu istemciyi kullanır.

Herhangi bir yapılandırma bilgisini değiştirmeniz gerekirse, değişiklikleri yapmadan önce tüm eksik iş birimlerinin çözümlendiğinden emin olun. Diğer bir seçenek olarak, yapılandırma değişikliklerinin, MTS tarafından kaydedilen bilgileri kullanarak bir MQI kanalını yeniden başlatmaya ilişkin genişletilmiş işlemsel istemcinin yeteneğini etkilemediğinden emin olun. Aşağıda, bu tür yapılandırma değişikliklerine ilişkin örnekler yer alıyor:

- MQSERVER ortam değişkeninin değerini değiştirme
 - İstemci kanal tanımlama çizelgesindeki girişleri değiştirme (CCDT)
 - Sunucu Başlantı Kanal Tanımlarının Silinmesi
- MTS ile genişletilmiş bir işlemsel istemci kullanırken aşağıdaki koşulları göz önünde bulundurun:
 - Tek bir iş parçacığıda, bir istemci uygulaması aynı anda yalnızca bir kuyruk yöneticisine bağlanabilir.
 - İstemci uygulamasının her bir iş parçacığı, farklı bir kuyruk yöneticisine bağlanabilir.
 - Bir istemci uygulaması paylaşılan bağlantı tanıtıcılarını kullanamaz.

MQI kanallarının tanımlanması

Yeni bir kanal yaratmak için, aynı kanal adını ve uyumlu kanal tiplerini kullanarak, bağlantının her bir ucu için bir tane olmak üzere **iki** kanal tanımlaması yaratmanız gerekir. Bu durumda, kanal tipleri *sunucu-bağlantısı* ve *istemci-bağlantı* tipleridir.

Kullanıcı tanımlı kanallar

When the server does not automatically define channels there are two ways of creating the channel definitions and giving the WebSphere MQ application on the WebSphere MQ MQI client machine access to the channel.

Bu iki yöntem ayrıntılı olarak açıklanmıştır:

1. WebSphere MQ istemcisinde bir kanal tanımlaması yaratın ve sunucuda başka bir kanal tanımlaması yaratın.

Bu, WebSphere MQ MQI istemcisi ve sunucu platformlarının herhangi bir birleşimi için geçerlidir. Sistemde çalışmaya başladığınızda ya da kurulumunuzu sınamak için bunu kullanın.

Bu yöntemin kullanılmasına ilişkin ayrıntılar için [“Farklı altyapılarda sunucu bağlantısı ve istemci-bağlantı tanımlamaları yaratılıyor” sayfa 111 ' e bakın.](#)

2. Sunucu makinesinde her iki kanal tanımlamasını da yaratın.

Birden çok kanalı ve WebSphere MQ MQI istemcisi makineleri aynı anda ayarlarken bu yöntemi kullanın.

Bu yöntemin kullanılmasına ilişkin ayrıntılar için [“Sunucuda sunucu bağlantısı ve istemci-bağlantı tanımları yaratılması” sayfa 114](#) ' e bakın.

Otomatik olarak tanımlı kanallar

z/OS dışındaki platformlarda WebSphere MQ ürünleri, sunucuda yoksa otomatik olarak bir kanal tanımlaması oluşturabilecek bir özellik içerir.

Bir istemciden gelen bağlantı isteği alındıysa ve o kuyruk yöneticisinden uygun bir sunucu bağlantısı tanımı bulunamazsa, WebSphere MQ otomatik olarak bir tanımlama yaratır ve bunu kuyruk yöneticisine ekler. Otomatik tanımlama, varsayılan sunucu bağlantısı kanalı SYSTEM.AUTO.SVRCONN. ALTER QMGR komutunu CHAD değiştirgesiyle (ya da ChannelAutoDef değiştirgesiyle PCF komutu Change Queue Manager) kullanarak, kuyruk yöneticisi nesnesini güncelleyerek, sunucu bağlantısı tanımlarının otomatik tanımlamasını etkinleştirmenizi sağlar.

Kanal tanımlarının otomatik olarak yaratılmasıyla ilgili ek bilgi için [Günlük nesnesinin ve sunucu bağlantısı kanallarının otomatik tanımlaması](#) başlıklı konuya bakın.

İlgili kavramlar

“Otomatik olarak tanımlı kanallar” [sayfa 110](#)

z/OS dışındaki platformlarda WebSphere MQ ürünleri, sunucuda yoksa otomatik olarak bir kanal tanımlaması oluşturabilecek bir özellik içerir.

“Kullanıcı tanımlı kanallar” [sayfa 110](#)

When the server does not automatically define channels there are two ways of creating the channel definitions and giving the WebSphere MQ application on the WebSphere MQ MQI client machine access to the channel.

“Kanal denetimi işlevi” [sayfa 52](#)

Kanal denetim işlevi, kanalları tanımlamanız, izlemeniz ve denetlemeniz için gerekli olanakları sağlar.

Otomatik olarak tanımlı kanallar

z/OS dışındaki platformlarda WebSphere MQ ürünleri, sunucuda yoksa otomatik olarak bir kanal tanımlaması oluşturabilecek bir özellik içerir.

Bir istemciden gelen bağlantı isteği alındıysa ve o kuyruk yöneticisinden uygun bir sunucu bağlantısı tanımı bulunamazsa, WebSphere MQ otomatik olarak bir tanımlama yaratır ve bunu kuyruk yöneticisine ekler. Otomatik tanımlama, varsayılan sunucu bağlantısı kanalı SYSTEM.AUTO.SVRCONN. ALTER QMGR komutunu CHAD değiştirgesiyle (ya da ChannelAutoDef değiştirgesiyle PCF komutu Change Queue Manager) kullanarak, kuyruk yöneticisi nesnesini güncelleyerek, sunucu bağlantısı tanımlarının otomatik tanımlamasını etkinleştirmenizi sağlar.

Kullanıcı tanımlı kanallar

When the server does not automatically define channels there are two ways of creating the channel definitions and giving the WebSphere MQ application on the WebSphere MQ MQI client machine access to the channel.

Bu iki yöntem ayrıntılı olarak açıklanmıştır:

1. WebSphere MQ istemcisinde bir kanal tanımlaması yaratın ve sunucuda başka bir kanal tanımlaması yaratın.

Bu, WebSphere MQ MQI istemcisi ve sunucu platformlarının herhangi bir birleşimi için geçerlidir. Sistemde çalışmaya başladığınızda ya da kurulumunuzu sınamak için bunu kullanın.

Bu yöntemin kullanılmasına ilişkin ayrıntılar için [“Farklı altyapılarda sunucu bağlantısı ve istemci-bağlantı tanımlamaları yaratılıyor” sayfa 111](#) ' e bakın.

2. Sunucu makinesinde her iki kanal tanımlamasını da yaratın.

Birden çok kanalı ve WebSphere MQ MQI istemcisi makineleri aynı anda ayarlarken bu yöntemi kullanın.

Bu yöntemin kullanılmasına ilişkin ayrıntılar için [“Sunucuda sunucu bağlantısı ve istemci-bağlantı tanımları yaratılması” sayfa 114](#) ' e bakın.

Farklı altyapılarda sunucu bağlantısı ve istemci-bağlantı tanımlamaları yaratılıyor

Geçerli olduğu bilgisayarda her bir kanal tanımlaması oluşturabilirsiniz. Bir istemci bilgisayarında kanal tanımlarının nasıl yaratılabileceğiyle ilgili kısıtlamalar vardır.

Tüm platformlarda, sunucu makinesinde bir sunucu bağlantısı kanalı tanımlamak için WebSphere MQ Script (MQSC) komutlarını, programlanabilir komut biçimi (PCF) komutlarını ya da IBM WebSphere MQ Explorer ' ı kullanabilirsiniz.

Because MQSC commands are not available on a machine where WebSphere MQ has been installed as a WebSphere MQ MQI client only, you must use different ways of defining a client-connection channel on the client machine.

İlgili kavramlar

[“IBM WebSphere MQ MQI istemcisinde istemci-bağlantı kanalı yaratılması” sayfa 112](#)

MQSERVER kullanarak istemci iş istasyonunda bir istemci bağlantısı kanalı tanımlayabilir ya da MQCONNX çağrısında MQCNO yapısını kullanabilirsiniz.

İlgili görevler

[“Sunucuda sunucu bağlantısı kanalı tanımlanması” sayfa 111](#)

Gerekliyse MQSC ' yi başlatın, sonra sunucu bağlantısı kanalını tanımlayın.

Sunucuda sunucu bağlantısı kanalı tanımlanması

Gerekliyse MQSC ' yi başlatın, sonra sunucu bağlantısı kanalını tanımlayın.

Yordam

1. İsteğe bağlı: Sunucu altyapınız z/OSdeğilse, önce bir kuyruk yöneticisi yaratın ve başlatın ve MQSC komutlarını başlatın.

- a) Örneğin, QM1 adında bir kuyruk yöneticisi yaratın:

```
crtmqm QM1
```

- b) Kuyruk yöneticisini başlat:

```
strmqm QM1
```

- c) MQSC komutlarını başlat:

```
runmqsc QM1
```

2. Seçilen adı ve kanal tipi *sunucu-bağlantısı* olan bir kanalı tanımlayın.

```
DEFINE CHANNEL(CHAN1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +  
DESCR('Server-connection to Client_1')
```

Bu kanal tanımlaması, sunucuda çalışan kuyruk yöneticisiyle ilişkilidir.

3. Kuyruk yöneticinizin gelen bağlantı erişimine izin vermek için aşağıdaki komutu kullanın:

```
SET CHLAUTH(CHAN1) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- Burada SET CHLAUTH, önceki adımda tanımlanan kanalın adını kullanır.
- Burada 'IP adresi' , istemcinin IP adresidir.
- Burada 'kullanıcı kimliği' , hedef kuyruklara erişim denetimi için kanala sağlamak istediğiniz tanıtıcıdır. Bu alan büyük ve küçük harfe duyarlıdır.

Gelen bağlantınızı bir dizi farklı öznitelik kullanarak tanımlamayı seçebilirsiniz. Bu örnek, IP adresini kullanır. Diğer öznitelikler, istemci kullanıcı kimliğini ve SSL ya da TLS Konusu Ayırt Edici Adını içerir. Ek bilgi için [Kanal kimlik doğrulama kayıtları](#) başlıklı konuya bakın.

IBM WebSphere MQ MQI istemcisinde istemci-bağlantı kanalı yaratılması

MQSERVER kullanarak istemci iş istasyonunda bir istemci bağlantısı kanalı tanımlayabilir ya da MQCONN çağrısında MQCNO yapısını kullanabilirsiniz.

MQSERVER olanağının kullanılması

Bir istemci bağlantısı kanalının basit tanımlamasını belirtmek için MQSERVER ortam değişkenini kullanabilirsiniz. Bu yöntemi kullanarak, kanalın yalnızca birkaç özneliğini belirtebileceğiniz anlamda basit bir yöntemdir.

- Windows ' da aşağıdaki gibi basit bir kanal tanımlaması belirleyin:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName
```

- UNIX and Linux sistemlerinde aşağıdaki gibi basit bir kanal tanımlaması belirleyin:

```
export MQSERVER=ChannelName/TransportType/ConnectionName
```

Burada:

- ChannelName , sunucuda tanımlanla aynı adı olmalıdır. Eğik çizgi içermemesi gerekir.
- TransportType , IBM WebSphere MQ MQI istemci altyapınıza bağlı olarak aşağıdaki değerlerden biri olabilir:
 - LU62
 - TCP
 - NETBIOS
 - SPX

Not: UNIX and Linux sistemlerinde, TransportType büyük ve küçük harfe duyarlıdır ve büyük harf olmalıdır. An MQCONN or MQCONNX call returns 2058 if the TransportType is not recognized

- ConnectionName , iletişim protokolünün (TransportType) tanımlandığı şekilde, sunucunun adıdır.

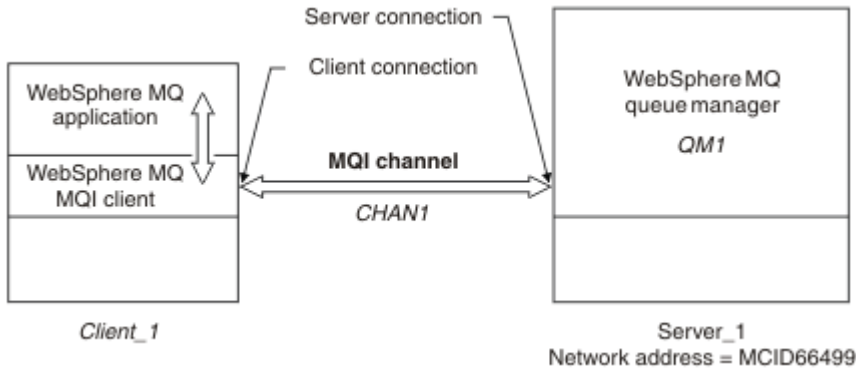
Örneğin, Windowsüzerinde:

```
SET MQSERVER=CHANNEL1/TCP/MCID66499
```

ya da UNIX and Linux sistemlerinde:

```
export MQSERVER=CHANNEL1/TCP/'MCID66499'
```

Not: TCP/IP kapı numarasını değiştirmek için bkz. [“MQSERVER” sayfa 145.](#)



Şekil 17. Basit kanal tanımlaması

Basit kanal tanımlamalarına ilişkin bazı örnekler şunlardır:

- Windows'ta:

```
SET MQSERVER=CHANNEL1/TCP/9.20.4.56
SET MQSERVER=CHANNEL1/NETBIOS/BOX643
```

- UNIX and Linux sistemlerinde:

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56'
export MQSERVER=CHANNEL1/LU62/BOX99
```

Burada BOX99 , LU 6.2 ConnectionName' dir.

On the IBM WebSphere MQ MQI client, all **MQCONN** or **MQCONNX** requests then attempt to use the channel you have defined, unless the channel is overridden in an **MQCD** structure referenced from the **MQCNO** structure supplied to **MQCONNX**.

Not: **MQSERVER** ortam değişkenine ilişkin ek bilgi için bkz. "**MQSERVER**" sayfa 145.

MQCONNX çağrısındaki MQCNO yapısını kullanma

Bir IBM WebSphere MQ MQI istemci uygulaması, bir istemci bağlantısı kanalının tanımını içeren bir kanal tanımlama yapısına, **MQCD** 'ye gönderme yapmak için **MQCONNX** çağrısında, bağlantı seçenekleri yapısını, **MQCNO**' yu kullanabilir.

Bu şekilde, istemci uygulaması yürütme sırasında bir kanala ilişkin **ChannelName**, **TransportType** ve **ConnectionName** özniteliklerini belirleyebilir ve istemci uygulamasının aynı anda birden çok sunucu kuyruğu yöneticisine bağlanmasını sağlar.

MQSERVER ortam değişkenini kullanarak bir kanal tanımlarsanız, yürütme sırasında **ChannelName**, **TransportType** ve **ConnectionName** özniteliklerinin belirtilmesine olanak olmadığını unutmayın.

A client application can also specify attributes of a channel such as **MaxMsgLength** and **SecurityExit**. Bu tür özniteliklerin belirtilmesi, istemci uygulamasının, varsayılan değer olmayan öznitelikler için değer belirtmesine ve kanal çıkış programlarının bir MQI kanalının istemci ucunda çağrılmasına olanak sağlar.

Bir kanal SSL (Secure Sockets Layer; Güvenli Yuva Katmanı) ya da TLS (Transport Layer Security; İletim Katmanı Güvenliği) kullanıyorsa, bir istemci uygulaması, **MQCD** yapısında SSL ya da TLS ile ilgili bilgileri de sağlayabilir. SSL ya da TLS ile ilgili ek bilgiler, SSL ya da TLS yapılandırma seçenekleri yapısında sağlanabilir. Bu yapıda, bir **MQCONNX** çağrısında **MQCNO** yapısı tarafından da gönderme yapılan **MQSCO**.

MQCNO, **MQCD** ve **MQSCO** yapılarına ilişkin ek bilgi için **MQCNO**, **MQCD** ve **MQSCO** başlıklı konuya bakın.

Not: **MQCONNX** için örnek program **amqscnxc** olarak adlandırılır. **amqssslc** adlı başka bir örnek program, **MQSCO** yapısının kullanımını gösterir.

Sunucuda sunucu bağlantısı ve istemci-bağlantı tanımları yaratılması

Sunucuda her iki tanımlamayı da yaratabilir ve istemci bağlantısı tanımlamasını istemci tarafından kullanılabilir duruma getirebilirsiniz.

Önce bir sunucu bağlantısı kanalı tanımlayın ve sonra bir istemci bağlantısı kanalı tanımlayın. Tüm altyapılarda, sunucu makinesinde bir sunucu bağlantısı kanalı tanımlamak için WebSphere MQ Script (MQSC) komutlarını, programlanabilir komut biçimi (PCF) komutlarını ya da IBM WebSphere MQ Explorer 'ı kullanabilirsiniz.

Sunucuda yaratılan istemci-bağlantı kanalı tanımlamaları, istemci kanal tanımlama çizelgesi (CCDT) kullanan istemcilerde kullanılır.

İlgili kavramlar

[“İstemci kanal tanımlama çizelgesi” sayfa 114](#)

İstemci kanalı tanımlama çizelgesi (CCDT), istemci uygulamalarının kuyruk yöneticisine bağlanmak için kullandığı kanal tanımlarını ve kimlik denetimi bilgilerini belirler. z/OS dışındaki platformlarda otomatik olarak bir CCDT yaratılır. Bundan sonra, istemci uygulamasının kullanımına sunulmalıdır.

İlgili görevler

[“Sunucu bağlantısı kanalının sunucuda tanımlanması” sayfa 116](#)

Kuyruk yöneticisi için bir sunucu bağlantısı kanal tanımlaması yaratın.

[“İstemci-bağlantı kanalının sunucu üzerinde tanımlanması” sayfa 117](#)

Sunucu bağlantısı kanalını tanımladığınızda, şimdi ilgili istemci bağlantısı kanalını tanımlırsınız.

[“İstemci-bağlantı kanalı tanımlarına erişilmesi” sayfa 117](#)

İstemci kanal tanımlama çizelgesini (CCDT) istemci uygulamaları tarafından kopyalayarak ya da paylaşarak kullanabildikten sonra, istemci bilgisayarında yerini ve adını belirtin.

İstemci kanal tanımlama çizelgesi

İstemci kanalı tanımlama çizelgesi (CCDT), istemci uygulamalarının kuyruk yöneticisine bağlanmak için kullandığı kanal tanımlarını ve kimlik denetimi bilgilerini belirler. z/OS dışındaki platformlarda otomatik olarak bir CCDT yaratılır. Bundan sonra, istemci uygulamasının kullanımına sunulmalıdır.

İstemci kanalı tanımlama çizelgesinin (CCDT) amacı, istemci uygulamalarının kuyruk yöneticisine bağlanmak için kullandığı kanal tanımlamalarını saptamasıdır. Kanal tanımlaması, bağlantılar için geçerli olan kimlik doğrulama bilgilerini de belirtir.

CCDT, ikili bir dosyadır. Bir kuyruk yöneticisi tarafından oluşturulur. Kuyruk yöneticisi CCDT dosyasını okumuyor.

z/OSdışındaki platformlarda, kuyruk yöneticisi oluşturulduğunda CCDT yaratılır. Client connection channels are added to the table when you use the **DEFINE CHANNEL** command, and their definitions altered when you issue the **ALTER CHANNEL** command.

SSL sertifikasının iptal olup olmadığını denetlemek için kimlik doğrulama bilgilerini içeren istemcilere sağlamak için CCDT 'yi kullanabilirsiniz. Kimlik doğrulama bilgileri nesnelere içeren bir ad listesi tanımlayın ve **SSLCRLNameList** kuyruk yöneticisi özneteliğini, ad listesinin adıyla ayarlayın.

İstemci uygulamasının CCDT kullanmasını sağlamak için bir dizi yol vardır. CCDT, istemci bilgisayarına kopyalanabilir. CCDT 'yi birden çok istemci tarafından paylaşılan bir konuma kopyalayabilirsiniz. CCDT 'yi istemci tarafından paylaşılan bir dosya olarak erişilebilir, ancak sunucu üzerinde kalır.

Dosyayı kopyalamak için FTP 'yi kullanırsanız, ikili kipi ayarlamak için bin seçeneğini kullanın; varsayılan ASCII kipini kullanmayın. CCDT 'yi hangi yöntemle kullanabildiğinizi seçerseniz, kanallarda yetkisiz değişiklikler yapılmasını önlemek için konum güvenli olmalıdır.

z/OSdışındaki platformlar

Bir kuyruk yöneticisi yarattığınızda varsayılan bir CCDT adı AMQCLCHL .TAB yaratılır.

Varsayılan değer olarak, AMQCLCHL.TAB bir sunucu üzerinde aşağıdaki dizinde bulunur:

- **UNIX** **Linux** UNIX and Linux sistemlerinde:

```
/prefix/qmgrs/QUEUENAME/@ipcc
```

QUEUENAME tarafından başvuru dizinin adı, UNIX and Linux sistemlerinde büyük/küçük harfe duyarlıdır. Kuyruk yöneticisi adının içinde özel karakterler varsa, dizin adı kuyruk yöneticisi adıyla aynı olmayabilir.

- **Windows** Windows sistemlerinde:

```
MQ_INSTALLATION_PATH\data\qmgrs\QUEUENAME\@ipcc
```

MQ_INSTALLATION_PATH , IBM WebSphere MQ ' in kurulu olduğu üst düzey dizini temsil eder.

Ancak, kuyruk yöneticisi verileri için farklı bir dizin kullanmayı seçmiş olabilirsiniz. You can specify the parameter **-md DataPath** when you used the **crtmqm** command. Bunu yapmazsanız, AMQCLCHL . TAB , belirttiğiniz *DataPath* ' nun @ipcc dizininde bulunur.

CCDT ' nin yolu MQCHLLIB olarak değiştirilebilir. MQCHLLIB ' u ayarladıysanız, aynı sunucuda birden çok kuyruk yöneticisi varsa, aynı CCDT konumunu paylaşırlar.

CCDT, kuyruk yöneticisi yaratıldığında yaratılır. Bir CCDT ' nin her girişi, belirli bir kuyruk yöneticisiyle olan istemci bağlantısını gösterir. A new entry is added when you define a client-connection channel using the **DEFINE CHANNEL** command, and the entry is updated when you alter the client-connection channels by using the **ALTER CHANNEL** command.

İstemcide CCDT ' nin yerini nasıl belirtmeliyim?

Bir istemci sisteminde CCDT ' nin yerini iki şekilde belirtebilirsiniz:

- Using the environment variables MQCHLLIB to specify the directory where the table is located, and MQCHLTAB to specify the file name of the table.
- İstemci yapılandırma dosyası kullanılıyor. CHANNELS stanza içinde, çizelgenin bulunduğu dizini belirtmek için ChannelDefinitionDirectory özniteliklerini ve dosya adını belirtmek için ChannelDefinitionFile değerini kullanın.

Konum hem istemci yapılandırma dosyasında, hem de ortam değişkenleri kullanılarak belirtilirse, ortam değişkenleri önceliğe sahip olur. Bu özelliği, istemci yapılandırma dosyasında standart bir yer belirtmek ve gerektiğinde ortam değişkenlerini kullanarak geçersiz kılmak için kullanabilirsiniz.

İlgili başvurular

[“MQCHLIB” sayfa 143](#)

MQCHLLIB , istemci kanal tanımlama çizelgesini (CCDT) içeren dosyaya ilişkin dizin yolunu belirtir. The file is created on the server, but can be copied across to the WebSphere MQ MQI client workstation.

İlgili bilgiler

[İptal edilen sertifikalarla çalışma](#)

Geçiş ve istemci kanal tanımlama çizelgeleri (CCDT)

Genel olarak, istemci kanal tanımlama çizelgesinin iç biçimi, IBM WebSphere MQ yayın düzeyinden sonraki yayın düzeyinden sonraki bir yayın düzeyine geçebilir. Sonuç olarak, bir IBM WebSphere MQ MQI istemcisi, istemci kanal tanımlama çizelgesini yalnızca istemci ile aynı yayın düzeyinde olan ya da daha önceki bir yayın düzeyinde olan bir sunucu kuyruk yöneticisi tarafından hazırlandığında kullanılabilir.

Bir Sürüm 7.1 IBM WebSphere MQ MQI istemcisi, bir Sürüm 6.0 kuyruk yöneticisi tarafından hazırlanmış bir istemci kanalı tanımlama çizelgesini kullanabilir. Ancak, Sürüm 6.0 istemcisi, Sürüm 7.1 kuyruk yöneticisi tarafından hazırlanmış istemci kanalı tanımlama çizelgesini kullanamaz.

Active Directory’indeki istemci bağlantı kanalları

On Pencereler systems that support the Active Directory, IBM WebSphere MQ publishes client connection channels in the Active Directory to provide dynamic client-server binding.

İstemci bağlantı kanalı nesnelere tanımlandığında, bunlar bir istemci kanal tanımlama dosyası, varsayılan olarak AMQCLCHL.TAB olarak adlandırılır. İçine yazılır. İstemci bağlantı kanalları TCP/IP protokolünü kullanırsa, IBM WebSphere MQ sunucusu bunları Active Directory (Active Directory) içinde de yayınlar. When the IBM WebSphere MQ client determines how to connect to the server, it looks for a relevant client connection channel object definition using the following search order:

1. MQCONNX MQCD veri yapısı
2. MQSERVER ortam değişkeni
3. istemci kanal tanımlama dosyası
4. Active Directory

Bu sipariş, herhangi bir geçerli uygulamanın herhangi bir değişiklikten etkilenmediği anlamına gelir. You can think of these entries in the Active Directory as records in the client channel definition file, and the IBM WebSphere MQ client processes them in the same way. İstemci bağlantı kanalı tanımlamalarını Active Directory' de yayınlamaya ilişkin desteği yapılandırmak ve denetlemek için [setmqscpi](#) içinde açıklandığı gibi `setmqscp` komutunu kullanın.

Sunucu bağlantısı kanalının sunucuda tanımlanması

Kuyruk yöneticisi için bir sunucu bağlantısı kanal tanımlaması yaratın.

Yordam

1. Sunucu makinesinde, seçilen adı ve kanal tipi *sunucu-bağlantısı* olan bir kanalı tanımlayın. Örneğin:

```
DEFINE CHANNEL(CHAN2) CHLTYPE(SVRCONN) TRPTYPE(TCP) +  
DESCR('Server-connection to Client_2')
```

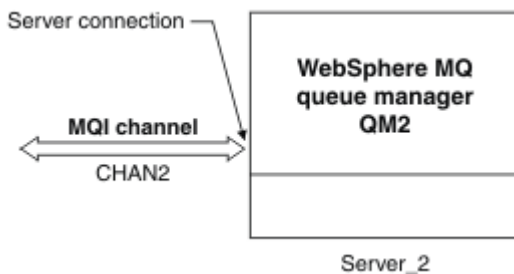
2. Kuyruk yöneticinizin gelen bağlantı erişimine izin vermek için aşağıdaki komutu kullanın:

```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- Burada SET CHLAUTH, önceki adımda tanımlanan kanalın adını kullanır.
- Burada 'IP adresi' IP adresi, istemcinin IP adresidir.
- Burada 'kullanıcı kimliği', hedef kuyruklara erişim denetimi için kanala sağlamak istediğiniz tanıtıcıdır. Bu alan büyük ve küçük harfe duyarlıdır.

Gelen bağlantınızı bir dizi farklı öznitelik kullanarak tanımlamayı seçebilirsiniz. Bu örnek, IP adresini kullanır. Diğer öznitelikler, istemci kullanıcı kimliğini ve SSL ya da TLS Konusu Ayırt Edici Adını içerir. Ek bilgi için [Kanal kimlik doğrulama kayıtları](#) başlıklı konuya bakın.

Bu kanal tanımlaması, sunucuda çalışan kuyruk yöneticisiyle ilişkilidir.



Şekil 18. Sunucu bağlantısı kanalının tanımlanması

İstemci-bağlantı kanalının sunucu üzerinde tanımlanması

Sunucu bağlantısı kanalını tanımladığınızda, şimdi ilgili istemci bağlantısı kanalını tanımlırsınız.

Başlamadan önce

Sunucu bağlantısı kanalını tanımlayın.

Yordam

1. Sunucu bağlantısı kanalıyla aynı adı taşıyan bir kanal tanımlayın, ancak kanal tipi *istemci-bağlantı*. Bağlantı adını (CONNNAME) belirtmelisiniz. TCP/IP için, bağlantı adı, sunucu makinelerinin ağ adresidir ya da anasistem adıdır. It is also advisable to specify the queue manager name (QMNAME) to which you want your IBM WebSphere MQ application, running in the client environment, to connect. Kuyruk yöneticisi adını değiştirdiğinizde, farklı kuyruk yöneticilerine bağlanmak için bir kanal kümesi tanımlayabilirsiniz.

```
DEFINE CHANNEL (CHAN2) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +  
CONNNAME(9.20.4.26) QMNAME(QM2) DESCR('Client-connection to Server_2')
```

2. Kuyruk yöneticinizin gelen bağlantı erişimine izin vermek için aşağıdaki komutu kullanın:

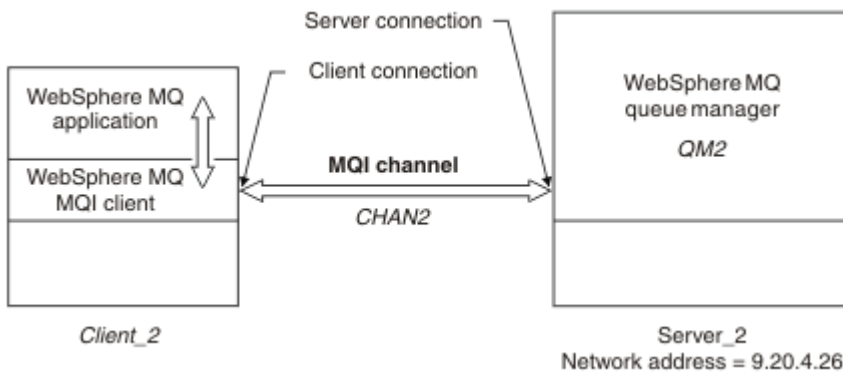
```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP-address') MCAUSER('userid')
```

- Burada SET CHLAUTH, önceki adımda tanımlanan kanalın adını kullanır.
- Burada 'IP adresi', istemcinin IP adresidir.
- Burada 'kullanıcı kimliği', hedef kuyruklara erişim denetimi için kanala sağlamak istediğiniz tanıtıcıdır. Bu alan büyük ve küçük harfe duyarlıdır.

Gelen bağlantınızı bir dizi farklı öznitelik kullanarak tanımlamayı seçebilirsiniz. Bu örnek, IP adresini kullanır. Diğer öznitelikler, istemci kullanıcı kimliğini ve SSL ya da TLS Konusu Ayırt Edici Adını içerir. Ek bilgi için [Kanal kimlik doğrulama kayıtları](#) başlıklı konuya bakın.

Sonuçlar

z/OS'dışındaki platformlarda, bu kanal tanımlaması kuyruk yöneticisiyle ilişkili istemci kanal tanımlama çizelgesi (CCDT) adlı bir dosyada saklanır. İstemci kanalı tanımlama çizelgesi birden çok istemci-bağlantı kanalı tanımlaması içerebilir. İstemci kanal tanımlama tablosuyla ilgili daha fazla bilgi için ve istemci-bağlantı kanalı tanımlamalarının z/OS'ünde nasıl depolanmış olduğuna ilişkin bilgi edinmek için bkz. "[İstemci kanal tanımlama çizelgesi](#)" sayfa 114.



Şekil 19. İstemci-bağlantı kanalının tanımlanması

İstemci-bağlantı kanalı tanımlarına erişilmesi

İstemci kanal tanımlama çizelgesini (CCDT) istemci uygulamaları tarafından kopyalayarak ya da paylaşarak kullanabildikten sonra, istemci bilgisayarında yerini ve adını belirtin.

Başlamadan önce

Gereksinim duyardığınız istemci bağlantı kanallarını tanımladınız.

z/OSüzerinde bir CCDT yaratmış buldunuz. Diğer platformlarda, CCDT otomatik olarak yaratılır ve güncellenir.

Bu görev hakkında

İstemci kanal tanımlama çizelgesini (CCDT) kullanmak için bir istemci uygulaması için, CCDT 'yi kullanılabilir duruma getirmeniz ve yerini ve adını belirtmeniz gerekir.

Yordam

1. CCDT 'yi istemci uygulamaları için şu üç yöntemden birini kullanarak yapabilirsiniz:
 - a) İsteğe bağlı: CCDT 'yi istemci bilgisayarına kopyalayın.
 - b) İsteğe bağlı: CCDT 'yi birden çok istemci tarafından paylaşılan bir konuma kopyalayın.
 - c) İsteğe bağlı: CCDT 'yi sunucuda bırakın, ancak istemci tarafından paylaşılabilir duruma getirilebilir. CCDT için hangi konumun seçileceğini belirlerseniz, kanallarda yetkisiz değişiklikler yapılmasını önlemek için konum güvenli olmalıdır.
2. İstemcide, CCDT 'yi içeren dosyanın yerini ve adını şu üç yöntemden birini kullanarak belirtin:
 - a) İsteğe bağlı: İstemci yapılandırma kütüğünün KANALLAR kISINı kullanın. Daha fazla bilgi için "[İstemci yapılandırma dosyasının STANA kısmı](#)" sayfa 133 başlıklı konuya bakın.
 - b) İsteğe bağlı: MQCHLLIB ve MQCHLTAB ortam değişkenlerini kullanın.
Örneğin, aşağıdaki bilgileri yazarak ortam değişkenlerini ayarlayabilirsiniz:
 - HP Integrity NonStop Serverve UNIX and Linux sistemlerinde:

```
export MQCHLLIB=MQ_INSTALLATION_PATH/qmgrs/QUEUMANAGERNAME/@ipcc
export MQCHLTAB=AMQCLCHL.TAB
```

Burada `MQ_INSTALLATION_PATH` , WebSphere MQ 'un kurulu olduğu üst düzey dizini temsil eder.

- c) İsteğe bağlı: On Windows only, use the **setmqscp** control command to publish the client-connection channel definitions in Active Directory

MQSERVER ortam değişkeni ayarlandıysa, bir WebSphere MQ istemcisi, istemci kanal tanımlama çizelgesindeki tanımlamalar için, MQSERVER tarafından belirtilen istemci-bağlantı kanalı tanımlamasını kullanır.

MQI kanallarına ilişkin kanal-çıkış programları

UNIX, Linux ve Pencerele sistemlerinde, WebSphere MQ MQI istemci ortamına üç tip kanal çıkışı vardır.

Bu bilgiler şunlardır:

- Çıkış gönder
- Çıkış al
- Güvenlik Çıkışı

Bu çıkışlar hem istemcide, hem de kanalda kullanılabilir. MQSERVER ortam değişkenini kullanıyorsanız, bu çıkışlar uygulamanız için kullanılabilir değildir. Kanal çıkışları, [ileti alışverişi kanallarına ilişkin kanal çıkış programları](#) içinde açıklanmıştır.

Gönderme ve alma çıkışları birlikte çalışır. Bunları kullanabilmeniz için kullanılabilecek birkaç yol vardır:

- İletinin bölünmesi ve yeniden kurulması
- Bir iletide verileri sıkıştırma ve sıkıştırma işlemi (bu işlevsellik WebSphere MQ' un bir parçası olarak sağlanır, ancak farklı bir sıkıştırma tekniği kullanmak isteyebilirsiniz)

- Kullanıcı verilerinin şifrelenmesi ve şifrelerinin çözülmesi (bu işlevsellik, WebSphere MQ' un bir parçası olarak sağlanır, ancak farklı bir şifreleme tekniği kullanmak isteyebilirsiniz)
- Gönderilen ve alınan her iletiyi günlüğe kaydetme

You can use the security exit to ensure that the WebSphere MQ client and server are correctly identified, and to control access.

Kanal yönetim ortamının sunucu bağlantısı tarafındaki gönderme ya da alma çıkışları, ilişkili oldukları bağlantıda MQI çağruları gerçekleştirilmeye gereksinim duyarsa, bu aramaları MQCXP Hconn alanında sağlanan bağlantı tanıtıcısını kullanır. İstemci-bağlantı gönderme ve alma çıkışlarının MQI çağrılarını gerçekleştirmediğinden haberdar olmalısınız.

İlgili kavramlar

[“İstemci bağlantısındaki güvenlik çıkışları” sayfa 119](#)

Bir kanalın diğer ucundaki ortağın gerçek olduğunu doğrulamak için güvenlik çıkış programlarını kullanabilirsiniz. Bir istemci bağlantısına güvenlik çıkışı uygulandığında dikkat edilmesi gereken özel noktalar vardır.

[Kullanıcı çıkışları, API çıkışları ve WebSphere MQ kurulabilir hizmetleri](#)

İlgili görevler

[Kuyruk yöneticisi olanaklarının genişletmesi](#)

İlgili başvurular

[“Çıkışa giden yol” sayfa 119](#)

İstemci yapılandırma dosyasında, kanal çıkışlarının yeri için varsayılan yol tanımlıdır. Kanal çıkışı, bir kanal başlatıldığında yüklenir.

[“Gönderme ya da alma çıkış programındaki API çağrısının tanımlanması” sayfa 121](#)

İstemciler için MQI kanallarını kullandığınızda, aracı arabelleğindeki bayt 10 'u, bir gönderme ya da alma çıkışı çağrıldığında kullanılan API çağrısını tanımlar. Bu, kullanıcı verilerini içeren ve şifreleme ya da dijital imzalama gibi işleme gerektirebilecek kanal akışlarını tanımlamak için kullanışlıdır.

Çıkışa giden yol

İstemci yapılandırma dosyasında, kanal çıkışlarının yeri için varsayılan yol tanımlıdır. Kanal çıkışı, bir kanal başlatıldığında yüklenir.

UNIX, Linux ve Windows sistemlerinde, WebSphere MQ MQI istemcisi kurulumu sırasında sisteminize bir istemci yapılandırma dosyası eklenir. Bu dosyada, istemcideki kanal çıkışlarının yeri için varsayılan bir yol tanımlıdır:

```
ClientExitPath:  
  ExitsDefaultPath=string  
  ExitsDefaultPath64=string
```

Burada *dizgi* , platforma uygun bir biçimde bir dosya konusudur.

Bir kanal başlatıldığında, bir MQCONN ya da MQCONNX çağrısından sonra istemci yapılandırma dosyası aranır. ClientExitYolu izlenir ve kanal tanımlamasında belirtilen kanal çıkışları yüklenir.

İstemci bağlantısındaki güvenlik çıkışları

Bir kanalın diğer ucundaki ortağın gerçek olduğunu doğrulamak için güvenlik çıkış programlarını kullanabilirsiniz. Bir istemci bağlantısına güvenlik çıkışı uygulandığında dikkat edilmesi gereken özel noktalar vardır.

Şekil 20 sayfa 121 , bir kullanıcının kimliğini doğrulamak için WebSphere MQ nesne yetkili yöneticisi kullanılarak istemci bağlantısında güvenlik çıkışlarının kullanılmasını gösterir. İstemcideki MQCNO yapısında SecurityParmsPtr ya da SecurityParmsGörelili Konumu ayarlıdır ve kanalın her iki ucunda da güvenlik çıkışları vardır. Olağan güvenlik iletisi değış tokası sona erdikten ve kanal çalışmaya hazır olduktan sonra, MQCXP SecurityParms alanından erişilen MQCSP yapısı, istemcideki güvenlik çıkışa geçirilir. Çıkış tipi MQXR_SEC_PARMS olarak ayarlıdır. Güvenlik çıkışı, kullanıcı kimliği ve parolaya

hiçbir şey yapmamayı seçebilir ya da her ikisini birden değiştirebilir. Çıkıştan döndürülen veriler, daha sonra, kanaldan sunucu bağlantısı ucuna gönderilir. MQCSP yapısı, kanalın sunucu bağlantısı ucunda yeniden oluşturulur ve MQCXP SecurityParms alanından erişilen sunucu bağlantısı güvenlik çıkışa geçirilir. Güvenlik çıkışı bu verileri alır ve işler. Bu işlem tipik olarak, istemci çıkışındaki kullanıcı kimliği ve parola alanlarında yapılan değişiklikleri tersine çevirerek, kuyruk yöneticisi bağlantısı için yetki vermek için kullanılır. Sonuçtaki MQCSP yapısına, kuyruk yöneticisi sistemindeki MQCNO yapısındaki SecurityParmsPtr kullanılarak gönderme yapılıyor.

If SecurityParmsPtr or SecurityParmsOffset are set in the MQCNO structure and there is a security exit at only one end of the channel, the security exit receives and processes the MQCSP structure. Tamamlayıcı işlemi gerçekleştirmek için herhangi bir çıkış olmadığı için, şifreleme gibi işlemler tek bir kullanıcı çıkışı için uygun değildir.

If SecurityParmsPtr and SecurityParmsOffset are not set in the MQCNO structure and there is a security exit at either or both ends of the channel, the security exit or exits are called. Güvenlik çıkışı, SecurityParmsPtr 'si aracılığıyla adreslenen kendi MQCSP yapısını döndürebilir; güvenlik çıkışı sona erdirilinceye kadar yeniden çağrılmaz (MQXR_TERM 'inExitReason ' ı). Çıkış yazma programı, o aşamada MQCSP için kullanılan belleği serbest bir şekilde serbest kılabilir.

Bir sunucu bağlantısı kanal yönetim ortamı birden çok etkileşim paylaşırken, güvenlik çıkışa çağrılarının kalıbı ikinci ve sonraki etkileşimler üzerinde sınırlandırılmıştır.

İlk etkileşim için, örüntü, kanal örneğinin sohbetleri paylaşmadığı gibi aynıdır. İkinci ve sonraki etkileşimler için, MQXR_INIT, MQXR_INIT_SEC ya da MQXR_SEC_MSG ile güvenlik çıkışı hiçbir zaman çağrılmızm. Bu, MQXR_SEC_PARMS ile çağrılır.

Sohbet paylaşımı içeren bir kanal örneğinde, MQXR_TERM yalnızca son etkileşim için çağrılır.

Her etkileşim MQCD ' yi değiştirmek için çıkışa ilişkin MQXR_SEC_PARMS çağrısına sahiptir; bu özelliğin sunucu bağlantısı ucunda, bu özelliğin sunucu bağlantısı sonunda değişiklik göstermeleri yararlı olabilir; örneğin, bağlantı kuyruk yöneticisinde yapılmadan önce MCAUserIdentifier ya da LongMCAUserIdPtr değerleri kullanılır.

Server-connection exit	Client-connection exit
	Invoked with MQXR_INIT Responds with MQXCC_OK
Invoked with MQXR_INIT Responds with MQXCC_OK	
	Invoked with MQXR_INIT_SEC Responds with MQXCC_OK
Invoked with MQXR_INIT_SEC Responds with MQXCC_OK	
	Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK
Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK	
Data transfer begins	
Invoked with MQXR_TERM Responds with MQXCC_OK	Invoked with MQXR_TERM Responds with MQXCC_OK

Şekil 20. İstemci bağlantısı-güvenlik deęiřtirgelerini kullanarak istemci bağlantısı için sözleşmeyle birlikte başlatılan deęiř tokuř

Not: WebSphere MQ v7.1 yayın düzeyinden önce oluşturulan güvenlik çıkıřı uygulamaları güncellenmesini gerektirebilir. Ek bilgi için bkz. [Kanal güvenlik çıkıř programları](#).

Gönderme ya da alma çıkıř programındaki API çağrısının tanımlanması

İstemciler için MQI kanallarını kullandığınızda, aracı arabelleęindeki bayt 10 'u, bir gönderme ya da alma çıkıřı çağrıldığında kullanılan API çağrısını tanımlar. Bu, kullanıcı verilerini içeren ve şifreleme ya da dijital imzalama gibi işleme gerektirebilecek kanal akıřlarını tanımlamak için kullanılıdır.

Ařaęıdaki çizelge, bir API çağrısı işlenirken kanal akıřının 10. baytında görüntölenen verileri göstermektedir.

Not: Bunlar, bu baytın tek deęerleri deęil. Dięer **ayrılmıř** deęerler var.

Çizelge 20. API çağrılarını tanımlama		
API çağrısı	İstek için bayt 10 deęeri	Yanıt için bayt 10 deęeri
MQCONN " 1 " sayfa 122, " 2 " sayfa 122	X'81 '	X' 91 '
MQDISC " 1 " sayfa 122	X'82 '	X' 92 '
MQOPEN " 3 " sayfa 122	X'83 '	X' 93

Çizelge 20. API çağrılarını tanımlama (devamı var)

API çağırısı	İstek için bayt 10 değeri	Yanıt için bayt 10 değeri
MQCLOSE	X'84 '	X' 94 '
MQGET "4" sayfa 122	X'85 '	X' 95 '
MQPUT "4" sayfa 122	X'86 '	X' 96 '
MQPUT1 isteği "4" sayfa 122	X'87 '	X' 97 '
MQSET isteği	X'88 '	X' 98 '
MQINQ isteği	X'89 '	X' 99 '
MQCMIT isteği	X'8A'	X'9A'
MQBACK isteği	X'8B'	X'9B'
MQSTAT isteği	X'8D'	X'9D'
MQSUB isteği	X'8E'	X'9E'
MQSUBRQ isteği	X'8F'	X'9F'
xa_start isteği	X'A1'	X'B1'
xa_end isteği	X'A2'	X'B2'
xa_open isteği	X'A3'	X'B3'
xa_close isteği	X'A4'	X'B4'
xa_hazırlama isteği	X'A5'	X'B5'
xa_commit isteği	X'A6'	X'B6'
xa_geridönüş isteği	X'A7'	X'B7'
xa_forget isteği	X'A8'	X'B8'
xa_kurtarma isteği	X'A9'	X'B9'
xa_complete isteği	X'AA'	X'BA '

Notlar:

1. İstemci ile sunucu arasındaki bağlantı, istemci uygulaması tarafından MQCONN kullanan bir bağlantı tarafından başlatılır. Bu nedenle, özellikle bu komut için, başka ağ akışları da vardır. Aynı durum, ağ bağlantısını sonlandıran MQDISC için de geçerlidir.
2. MQCONNX, istemci-sunucu bağlantısının amaçlarına ilişkin MQCONN ile aynı şekilde ele alınır.
3. Büyük bir dağıtım listesi açılırsa, tüm gerekli verileri SVRCONN MCA ' ya geçirebilmek için, MQOPEN çağırısı başına birden çok ağ akışı olabilir.
4. Büyük iletiler iletim kesimi büyüklüğünü aşabilir. Bu gerçekleşirse, tek bir API çağırısından kaynaklanan birçok ağ akışı olabilir.

İstemcinin kuyruk paylaşım grubuna bağlanması

Bir istemci ile kuyruk paylaşım grubunun üyesi olan bir sunucuda kuyruk yöneticisi arasında bir MQI kanalı yaratarak, bir istemciyi kuyruk paylaşım grubuna bağlayabilirsiniz.

Kuyruk paylaşım grubu, aynı paylaşılan kuyruklara erişebilen bir kuyruk yöneticisi kümesiyle oluşur.

Paylaşılan bir kuyruğa yerleştiren bir istemci, kuyruk paylaşım grubunun herhangi bir üyeye bağlanabiliyor. Bir kuyruk paylaşım grubuna bağlanmanın yararları, ön uç ve arka uç kullanılabilirliğine ve kapasite artırımına olası artışlardır. Belirli bir kuyruk yöneticisine ya da soysal arabirimle bağlantı kurabilirsiniz.

Bir kuyruk paylaşım grubundaki bir kuyruk yöneticisine doğrudan bağlanmak, iletileri paylaşılan bir hedef kuyruğuna yerleştirebileceğiniz ve arka uç kullanılabilirliğini artıran bir avantaj sağlar.

Bir kuyruk paylaşım grubunun soysal arabirimine bağlanma işlemi, gruptaki kuyruk yöneticilerinden biriyle oturum açar. İstemci kuyruk yöneticisi gruptaki herhangi bir kuyruk yöneticisiyle bağlantı kurabildiği için, bu durum ön uç kullanılabilirliğini artırır. Kuyruk paylaşım grubu içindeki belirli bir kuyruk yöneticisine bağlanmak istemiyorsanız, soysal arabirimi kullanarak gruba bağlanırsınız.

Soysal arabirim, bir WLM/DNS grubu adı ya da bir VTAM soysal kaynak adı ya da kuyruk paylaşım grubu için başka bir ortak arabirim olabilir.

Bir kuyruk paylaşım grubunun soysal arabirimine bağlanmak için, gruptaki herhangi bir kuyruk yöneticisinin erişebileceği kanal tanımlarını yaratmanız gerekir. Bunu yapmak için, gruptaki her kuyruk yöneticisinde de aynı tanımlara sahip olması gerekir.

SVRCONN kanalını aşağıdaki gibi tanımlayın:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') QSGDISP(GROUP)
```

Sunucudaki kanal tanımları paylaşılan bir DB2 havuzunda saklanır. Kuyruk paylaşım grubundaki her kuyruk yöneticisi, tanımın yerel bir kopyasını oluşturur ve bir MQCONN ya da MQCONNX çağrısı yayınlarken her zaman doğru sunucu bağlantısı kanalına bağlanmanızı sağlar.

CLNTCONN kanalını aşağıdaki gibi tanımlayın:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNNAME(WLM/DNS groupname) QMNAME(QSG1) +
DESCR('Client-connection to Queue Sharing Group QSG1') QSGDISP(GROUP)
```

Kuyruk paylaşım grubunun soysal arabirimi, istemci-bağlantı kanalındaki CONNNAME alanında saklandığından, artık gruptaki herhangi bir kuyruk yöneticisine bağlanabilirsiniz ve o grubun sahip olduğu paylaşılan kuyruklara yerleştirebilirsiniz.

İlgili kavramlar

[“Kanal tanımlamaları yaratılması” sayfa 123](#)

Bir kuyruk paylaşım grubunun soysal arabirimine bağlanmak için, gruptaki herhangi bir kuyruk yöneticisinin erişebileceği kanal tanımlarını yaratmanız gerekir. Bunu yapmak için, gruptaki her kuyruk yöneticisinde de aynı tanımlara sahip olması gerekir.

Kanal tanımlamaları yaratılması

Bir kuyruk paylaşım grubunun soysal arabirimine bağlanmak için, gruptaki herhangi bir kuyruk yöneticisinin erişebileceği kanal tanımlarını yaratmanız gerekir. Bunu yapmak için, gruptaki her kuyruk yöneticisinde de aynı tanımlara sahip olması gerekir.

SVRCONN kanalını aşağıdaki gibi tanımlayın:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') QSGDISP(GROUP)
```

Sunucudaki kanal tanımları paylaşılan bir DB2 havuzunda saklanır. Kuyruk paylaşım grubundaki her kuyruk yöneticisi, tanımın yerel bir kopyasını oluşturur ve bir MQCONN ya da MQCONNX çağrısı yayınlarken her zaman doğru sunucu bağlantısı kanalına bağlanmanızı sağlar.

CLNTCONN kanalını aşağıdaki gibi tanımlayın:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
```

```
CONNNAME(WLM/DNS_groupname) QMNAME(QSG1) +  
DESCR('Client-connection to Queue Sharing Group QSG1') QSGDISP(GROUP)
```

Kuyruk paylaşım grubunun soysal arabirimi, istemci-bağlantı kanalındaki CONNNAME alanında saklandığından, artık gruptaki herhangi bir kuyruk yöneticisine bağlanabilirsiniz ve o grubun sahip olduğu paylaşılan kuyruklara yerleştirebilirsiniz.

Yapılandırma dosyası kullanarak istemci yapılandırılması

Bir metin dosyasındaki öznitelikleri kullanarak istemcilerinizi yapılandırın. Bu öznitelikler, ortam değişkenleriyle ya da altyapıya özgü diğer yollarla geçersiz kılınabilir.

IBM WebSphere MQ MQI client platformunuzu, UNIX and Linux platformlarında kullanılan kuyruk yöneticisi yapılandırma dosyasına (qm.ini) benzer bir metin dosyası kullanarak yapılandırabilirsiniz. Dosya, her biri **attribute-name=değer** biçiminin bir dizi satırlarını içeren bir dizi stanzas içerir.

Bu belgede, bu dosyaya *WebSphere MQ MQI istemcisi yapılandırma dosyası* adı verilir; dosya adı genel olarak mqclient.ini' dir, ancak başka bir ad vermeyi seçebilirsiniz. Bu dosyadaki yapılandırma bilgileri tüm platformlar için ve MQI, IBM WebSphere MQ classes for Java, IBM WebSphere MQ classes for JMS, IBM WebSphere MQ classes for .NET, ve XMSkullanan istemcilere uygulanır.

IBM WebSphere MQ MQI client yapılandırma dosyasındaki öznitelikler çoğu IBM WebSphere MQ istemcisi için geçerli olsa da, yönetilen .NET ve XMS .NET istemcileri tarafından ya da IBM WebSphere MQ classes for Java ya da IBM WebSphere MQ classes for JMSkullanan istemciler tarafından okunmayan bazı öznitelikler vardır. Daha fazla bilgi için, bkz. [“IBM WebSphere MQ istemcilerinin her özneliği okuyabilecekleri” sayfa 126.](#)

Konfigürasyon özellikleri, bir istemci uygulamasının kuyruk yöneticilerine ayrı ayrı bir bağlantıya özel olmak yerine, kuyruk yöneticilerine yaptığı tüm bağlantılar için geçerlidir. Tek bir kuyruk yöneticisine yönelik bir bağlantıyla ilgili öznitelikler, örneğin bir MQCD yapısı kullanılarak ya da bir İstemci Kanal Tanımlama Çizelgesi (CCDT) kullanılarak programlanabilir olarak yapılandırılabilir.

Environment variables which were supported in releases of IBM WebSphere MQ earlier than Version 7.0 continue to be supported, and where such an environment variable matches an equivalent value in the client configuration file, the environment variable overrides the client configuration file value.

IBM WebSphere MQ classes for JMSkullanan bir istemci uygulaması için, istemci yapılandırma dosyasını aşağıdaki yöntemlerle de geçersiz kılabilirsiniz:

- JMS yapılandırma dosyasındaki özellikleri ayarlama
- JMS yapılandırma dosyasını da geçersiz kılan Java sistem özelliklerini ayarlama

.NET istemcisi için, istemci yapılandırma kütüğünü ve eşdeğer ortam değişkenlerini, .NET uygulaması yapılandırma kütüğünü kullanarak da geçersiz kılabilirsiniz.

İstemci yapılandırma dosyasını kullanarak birden çok kanal bağlantısı ayarlayamayacağına dikkat edin.

Örnek istemci yapılandırma dosyası

```
## Module Name: mqclient.ini                                ##  
## Type       : WebSphere MQ MQI client configuration file  ##  
## Function   : Define the configuration of a client        ##  
##           ##  
##*****##  
## Notes     :                                             ##  
## 1) This file defines the configuration of a client      ##  
##           ##  
##*****##  
  
ClientExitPath:  
  ExitsDefaultPath=/var/mqm/exits  
  ExitsDefaultPath64=/var/mqm/exits64  
  
TCP:  
  Library1=DLLName1  
  KeepAlive = Yes  
  ClntSndBufSize=32768
```

```
ClntRcvBuffSize=32768
Connect_Timeout=0

MessageBuffer:
  MaximumSize=-1
  Updatepercentage=-1
  PurgeTime=0

LU62:
  TPName
  Library1=DLLName1
  Library2=DLLName2

PreConnect:
  Module=amqldapi
  Function=myFunc
  Data=ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
  Sequence=1

CHANNELS:
  DefRecon=YES
  ServerConnectionParms=SALES.SVRCONN/TCP/hostname.x.com(1414)
```

İlgili başvurular

[“İstemci yapılandırma dosyasının konumu” sayfa 125](#)

Bir IBM WebSphere MQ MQI istemcisi yapılandırma dosyası, bir dizi yerde tutulabilir.

[“İstemci yapılandırma dosyasının STANA kısmı” sayfa 133](#)

İstemci kanallarına ilişkin bilgileri belirtmek için KANALLAR kısmını kullanın.

[“ClientExitPath stanza of the client configuration file” sayfa 135](#)

İstemci kanal çıkışlarının varsayılan konumlarını belirtmek için ClientExitYol Stanzası 'nı kullanın.

[“İstemci yapılandırma dosyasınınLU62, NETBIOS ve SPX kısmı” sayfa 136](#)

Yalnızca Windows sistemlerinde, belirtilen ağ protokollerine ilişkin yapılandırma değişikliklerini belirtmek için bu stanzaları kullanın.

[“İstemci yapılandırma dosyasınınMessageBuffer kısmı” sayfa 137](#)

İleti arabelleklerine ilişkin bilgileri belirtmek için MessageBuffer kısmı kullanın.

[“İstemci yapılandırma dosyasının SSL kısmı” sayfa 138](#)

SSL ya da TLS kullanımına ilişkin bilgileri belirtmek için SSL standı kullanın.

[“İstemci yapılandırma dosyasının TCP kısmı” sayfa 140](#)

TCP ağ iletişim kuralı yapılandırma parametrelerini belirlemek için TCP stanzını kullanın.

[“WebSphere MQ ortam değişkenlerinin kullanılması” sayfa 142](#)

Bu kısımda, WebSphere MQ MQI istemci uygulamalarıyla kullanabileceğiniz ortam değişkenleri açıklanmaktadır.

[“Kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi” sayfa 422](#)

Burada açıklanan öznitelikler, tek bir kuyruk yöneticisinin yapılandırmasını değiştirir. Bunlar, WebSphere MQ için herhangi bir ayarı geçersiz kılar.

İstemci yapılandırma dosyasının konumu

Bir IBM WebSphere MQ MQI istemcisi yapılandırma dosyası, bir dizi yerde tutulabilir.

İstemci uygulaması, IBM WebSphere MQ MQI istemcisi yapılandırma dosyasını bulmak için aşağıdaki arama yolunu kullanır:

1. MQCLNTCF ortam değişkeniyle belirtilen yer.

Bu ortam değişkeninin biçimi tam bir URL adresidir. This means the file name might not necessarily be mqclient.ini and facilitates placing the file on a network attached file-system.

Aşağıdakileri unutmayın:

- C, .NET ve XMS istemcileri yalnızca file: iletişim kuralını destekler; URL dizesi protocol: ile başlamazsa file: iletişim kuralı kabul edilir.

- Ortam deęişkenlerini okumayı desteklemeyen Java 1.4.2 JRE 'lerine izin vermek için, MQCLNTCF ortam deęişkeni bir MQCLNTCF Java System özellięiyle geçersiz kılınabilir.
2. Uygulamanın çalışma dizininde bulunan mqclient.ini adlı bir dosya.
 3. IBM WebSphere MQ veri dizininde Windows, UNIX and Linux sistemleri için mqclient.ini adlı bir dosya.

Aşağıdakileri unutmayın:

- IBM WebSphere MQ veri dizini, belirli platformlarda (örneğin, IBM i ve z/OS gibi) ya da istemcinin başka bir ürünle birlikte sağlandığı durumlarda yok.
 - UNIX and Linux sistemlerinde izin şöyledir: /var/mqm
 - Pencereler altyapılarında, kuruluş sırasında MQ_FILE_PATH ortam deęişkenini, veri dizinini gösterecek şekilde yapılandırınız. Olağan durumda C:\Program Files\IBM\WebSphere MQ
 - Okuma ortamı deęişkenlerini desteklemeyen Java 1.4.2 JRE 'lerine izin vermek için, MQ_FILE_PATH ortam deęişkenini bir MQ_FILE_PATH Java Sistem Özellięi ile el ile geçersiz kılabilirsiniz.
4. Platforma uygun standart bir dizinde mqclient.ini adlı bir dosya ve kullanıcılar tarafından erişilebilir:
 - Tüm Java istemcileri için bu deęer, user.home Java System Property 'nin deęeridir.
 - UNIX and Linux platformlarındaki C istemcileri için bu, HOME ortam deęişkeninin deęeri olur.
 - For C clients on Pencereler this is the concatenated values of the HOMEDRIVE and HOMEPATH environment variables.

Not: For the IBM WebSphere MQ client for HP Integrity NonStop Server, the mqclient.ini file must be located in the OSS file system. Guardian uygulamaları mqclient.ini dosyasını IBM WebSphere MQ veri dizinine yerleştirmeli ya da MQCLNTCF ortam deęişkenini OSS dosya sisteminde bir yere ayarlamalıdır.

IBM WebSphere MQ istemcilerinin her öznelięi okuyabilecekleri

IBM WebSphere MQ MQI client yapılandırma dosyasındaki özneliklerin çoęu, C istemcisi ve yönetilmeyen .NET istemcilerinde kullanılabilir. Ancak, yönetilen .NET ve XMS .NET istemcileri tarafından okunmayan bazı öznelikler ya da IBM WebSphere MQ classes for Java ya da IBM WebSphere MQ classes for JMS kullanan istemciler tarafından okunmayan öznelikler vardır.

Çizelge 21. Her istemci tipi için geçerli olan öznelikler						
mqclient.ini stanza adı ve öznelikleri	Tanım	C ve yönetilmeyen .NET	Java	JMS	Yönetilen .NET	Yönetilen XMS .NET
KANALLARI KANALLAR						
<u>CCSID</u>	Kullanılacak kodlanmış karakter takımı numarası.	Evet	Hayır	Hayır	Evet	Evet
<u>ChannelDefinitionDizini</u>	İstemci kanal tanımlama çizelgesini içeren dosyanın dizin yolu.	Evet	Hayır	Hayır	Evet	Evet

Çizelge 21. Her istemci tipi için geçerli olan öznitelikler (devamı var)

mqclient.ini stanza adı ve öznitelikleri	Tanım	C ve yönetilmeyen .NET	Java	JMS	Yönetilen .NET	Yönetilen XMS .NET
<u>ChannelDefinitionDosyası</u>	İstemci kanal tanımlama çizelgesini içeren dosyanın adı.	Evet	Hayır	Hayır	Evet	Evet
<u>ReconDelay</u>	Otomatik yeniden bağlanma işlemi yapan istemci programları için yeniden bağlanma gecikmesini yapılandırma için kullanılan bir yönetim seçeneği.	Evet	Hayır	Evet	Evet	Evet
<u>DefRecon</u>	İstemci programlarının otomatik olarak yeniden bağlanmasını sağlamak ya da otomatik olarak yeniden bağlanmaya yazılan bir istemci programının otomatik olarak yeniden bağlanmasını devre dışı bırakmak için bir yönetim seçeneği.	Evet	Hayır	Evet	Evet	Evet
<u>MQReconnectTimeout</u>	İstemciye yeniden bağlanmak için saniye cinsinden zamanaşımı oluştu.	Evet	Hayır	Hayır	Evet	Hayır

Çizelge 21. Her istemci tipi için geçerli olan öznelikler (devamı var)

mqclient.ini stanza adı ve öznelikleri	Tanım	C ve yönetilmeyen .NET	Java	JMS	Yönetilen .NET	Yönetilen XMS .NET
ServerConnectionParms	The location of the IBM WebSphere MQ server and the communication method to be used.	Evet	Hayır	Hayır	Evet	Evet
Put1DefaultAlwaysSync	Controls the behavior of the MQPUT1 function call with the option MQPMO_RESPONSE_AS_Q_DEF.	Evet	Evet	Evet	Evet	Evet
ClientExitYol gösterişi						
ExitsDefaultYolu	İstemciler için 32 bitlik kanal çıkışlarının yerini belirtir.	Evet	Hayır	Hayır	Evet	Evet
ExitsDefaultPath64	İstemciler için 64 bitlik kanal çıkışlarının yerini belirtir.	Evet	Hayır	Hayır	Evet	Evet
JavaExitsClassPath	Java çıkışı çalıştırıldığı da sınıf yoluna eklenecek değerler.	Hayır	Evet	Evet	Hayır	Hayır
MessageBuffer kısmı						
MaximumSize	Önden okuma arabelleğindeki büyüklük (KB), aralık 1-999 999).	Evet	Evet	Evet	Evet	Evet

Çizelge 21. Her istemci tipi için geçerli olan öz nitelikler (devamı var)

mqclient.in stanza adı ve öz nitelikleri	Tanım	C ve yönetilmeye n .NET	Java	JMS	Yönetilen .N ET	Yönetilen XMS .NET
<u>PurgeTime</u>	Önokuma arabelleğindeki iletilerin silineceği saniye cinsinden aralık.	Evet	Evet	Evet	Evet	Evet
<u>UpdatePercentage</u>	Güncelleme yüzdesi değeri, bir istemci uygulamasının sunucuya ne zaman yeni bir istek yaptığını belirlemek için eşik değerini hesaplarken kullanılan 1-100 aralığında kullanılır.	Evet	Evet	Evet	Evet	Evet
SSL stanza						
<u>CDPCheckExtensions</u>	Bu kuyruk yöneticisinde SSL ya da TLS kanallarının, CrlDistributionPoint sertifika uzantılarında adı geçen CDP sunucularını denetleyip denetlemeyecğini belirtir.	Evet	Hayır	Hayır	Hayır	Hayır
<u>CertificateLabel</u>	Kanal tanımlamasının sertifika etiketi.	Evet	Hayır	Hayır	Hayır	Hayır

Çizelge 21. Her istemci tipi için geçerli olan öznelikler (devamı var)

mqclient.i ni stanza adı ve öznelikleri	Tanım	C ve yönetilmeye n .NET	Java	JMS	Yönetilen .N ET	Yönetilen XMS .NET
CertificateVal İlkesi	Kullanılan sertifika geçerlilik denetimi tipini belirler.	Evet	Hayır	Hayır	Hayır	Hayır
ClientRevoca tionDenetiml eri	İstemci bağlanma çağrısı bir SSL/TLS kanalı kullanıyorsa, sertifika iptal denetiminin nasıl yapılandırılıp yapılandırılac ağını belirler.	Evet	Hayır	Hayır	Hayır	Hayır
EncryptionPo licySuiteB	Bir kanalın Suite-B uyumlu şifreleme kullanıp kullanmayac ağını ve hangi düzeyde kalınlık kullanılacağı nı belirler.	Evet	Hayır	Hayır	Hayır	Hayır
OCSPAAuthent ication	OCSP etkinleştirildi ğinde IBM WebSphere MQ davranışını tanımlar ve OCSP iptal denetimi, sertifika iptal durumunu belirleyeme mektedir.	Evet	Hayır	Hayır	Hayır	Hayır

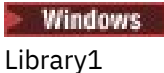
Çizelge 21. Her istemci tipi için geçerli olan öznitelikler (devamı var)

mqclient.inistanzaadı ve öznitelikleri	Tanım	C ve yönetilmeyen .NET	Java	JMS	Yönetilen .NET	Yönetilen XMS .NET
OCSPCheckExtensions	IBM WebSphere MQ 'in AuthorityInfo Erişimi sertifikası uzantılarına göre hareket edip etmeyeceğini denetler.	Evet	Hayır	Hayır	Hayır	Hayır
SSLCryptoHardware	Sistemde PKCSyapılandırma donanımını yapılandırma k için gereken parametre dizilimini ayarlar.	Evet	Hayır	Hayır	Hayır	Hayır
SSLFipsRequired	Specifies whether only FIPS-certified algorithms are to be used if cryptography is carried out in IBM WebSphere MQ.	Evet	Hayır	Hayır	Hayır	Hayır
SSLHTTPProxyName	Bu dizgi, GSKit for OCSP denetimlerini kullanarak kullanılacak HTTP yetkili sunucusunun anasistem adı ya da ağ adresidir.	Evet	Hayır	Hayır	Hayır	Hayır

Çizelge 21. Her istemci tipi için geçerli olan öznelikler (devamı var)

mqclient.i ni stanza adı ve öznelikleri	Tanım	C ve yönetilmeye n .NET	Java	JMS	Yönetilen .N ET	Yönetilen XMS .NET
SSLKeyRepository	Kullanıcının dijital sertifikasını kök biçiminde tutan anahtar havuzunun konumu.	Evet	Hayır	Hayır	Hayır	Hayır
SSLKeyReset Sayı	Gizli anahtar yeniden anlaşılmadan önce bir SSL ya da TLS kanalına gönderilen ve alınan şifrelenmemiş baytların sayısı.	Evet	Hayır	Hayır	Hayır	Hayır
TCP stanza						
ClntRcvBuffSize	İstemci-bağlantı sunucusu bağlantı kanalının istemci ucu tarafından kullanılan TCP/IP alma arabelleğindeki bayt cinsinden büyüklük.	Evet	Evet	Evet	Evet	Evet
ClntSndBuffSize	İstemci-bağlantı sunucusu bağlantı kanalının istemci ucu tarafından kullanılan TCP/IP gönderme arabelleğinin bayt cinsinden büyüklüğü.	Evet	Evet	Evet	Evet	Evet

Çizelge 21. Her istemci tipi için geçerli olan öznitelikler (devamı var)

mqclient.i ni stanza adı ve öznitelikleri	Tanım	C ve yönetilmeye n .NET	Java	JMS	Yönetilen .N ET	Yönetilen XMS .NET
Connect_Timeout	Yuva zamanlarını dışarı bağlama girişiminden önce geçecek süre (saniye).	Evet	Evet	Evet	Hayır	Hayır
IPAddressVersion	Bir kanal bağlantısı için hangi IP protokollerinin kullanılacağını belirler.	Evet	Hayır	Hayır	Evet	Evet
KeepAlive	KeepAlive (KeepAlive) işlevini açar ya da kapatır.	Evet	Evet	Evet	Evet	Evet
 Library1	Yalnızca Windows üzerinde, TCP/IP yuvaları DLL 'inin adı.	Evet	Hayır	Hayır	Hayır	Hayır

HP Integrity NonStop Server için, TMF/Gateway ile iletişim kurmak için [TMF](#) ve [TmfGateway](#) stanzaları 'nı kullanabilirsiniz.

İstemci yapılandırma dosyasının STANA kısmı

İstemci kanallarına ilişkin bilgileri belirtmek için KANALLAR kısmını kullanın.

Aşağıdaki öznitelikler KANALLARDA yer alabilirler:

CCSID=sayı

Kullanılacak kodlanmış karakter takımı numarası.

CCSID numarası, MQCCSID ortam parametresine eşdeğerdir.

ChannelDefinitionDirectory=yol

İstemci kanal tanımlama çizelgesini içeren dosyanın dizin yolu.

Windows sistemlerinde varsayılan değer IBM WebSphere MQ kuruluş dizinidir (genellikle C:\Program Files\IBM\WebSphere MQ). UNIX and Linux sistemlerinde varsayılan değer /var/mqm' dir.

ChannelDefinitionDizin yolu, MQCHLLIB ortam parametresine eşdeğerdir.

ChannelDefinitionFile=kütükadı| AMQCLCHL.TAB

İstemci kanal tanımlama çizelgesini içeren dosyanın adı.

İstemci kanalı tanımlama çizelgesi, MQCHLTAB ortam parametresine eşdeğerdir.

ReconDelay=(delay [, rand])(delay [, rand]) ...

ReconDelay özneliği, otomatik yeniden bağlanma olanağı sağlayan istemci programları için yeniden bağlanma gecikmesi yapılandırmak üzere bir yönetim seçeneği sağlar. Aşağıda bir örnek yapılanış vardır:

```
ReconDelay=(1000,200) (2000,200) (4000,1000)
```

Gösterilen örnekte, bir saniyenin ilk gecikmesi, artı 200 milisaniyeye kadar rasgele bir aralık tanımlanmaktadır. Bir sonraki gecikme, iki saniye artı 200 milisaniyeye kadar rasgele bir aralığa sahip. Sonraki tüm gecikmeler dört saniyedir, artı 1000 milisaniyeye kadar olan rasgele bir aralıkla.

DefRecon=NO|YES|QMGR|DISABLE

DefRecon özneliği, istemci programlarının otomatik olarak yeniden bağlanmasını ya da otomatik olarak yeniden bağlanmaya yazılan bir istemci programının otomatik olarak yeniden bağlantısını geçersiz kılacak bir yönetim seçeneği sağlar. Bir program, yeniden bağlantıyla uyumsuz olan MQPMO_LOGICAL_ORDER gibi bir seçeneği kullanırsa, ikincisini ayarlamayı tercih edebilirsiniz.

DefRecon seçeneklerinin yorumlanması, istemci programında bir MQCNO_RECONNECT_* değerinin de ayarlanıp ayarlanmadığına ve hangi değer ayarlanıp ayarlanmadığına bağlıdır.

İstemci programı MQCONNkomutunu kullanarak bağlanırsa ya da MQCONNXseçeneğini kullanarak MQCNO_RECONNECT_AS_DEF seçeneğini ayarlarsa, DefRecon tarafından ayarlanan yeniden bağlanma değeri yürürlüğe girer. Programda yeniden bağlanma değeri ayarlanmadıysa ya da DefRecon seçeneği belirlenmezse, istemci programı otomatik olarak yeniden bağlanmaz.

Otomatik istemci yeniden bağlantısı, Java için IBM WebSphere MQ sınıfları tarafından desteklenmez.

HAYIR

MQCONNXtarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanmaz.

EVET

MQCONNXtarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanır.

QMGR

MQCONNXtarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanır, ancak yalnızca aynı kuyruk yöneticisi tarafından değiştirilir. MMGR seçeneği, MQCNO_RECONNECT_Q_MGR ile aynı etkiye sahiptir.

DEVRE DIŞI

Reconnection is disabled, even if requested by the client program using the MQCONNX MQI call.

Otomatik istemci yeniden bağlantısı iki değere bağlıdır:

- Uygulamadaki yeniden bağlanma seçeneği belirlendi
- mqclient.ini dosyasındaki DefRecon değeri

Çizelge 22. Otomatik yeniden bağlantı, uygulamadaki ve mqclient.ini dosyasında ayarlanan değerlere bağlıdır.

mqclient. ini içindeki DefRecon değeri	Yeniden bağlanma seçenekleri uygulama içinde ayarlandı			
	MQCNO_RECONNE CT	MQCNO_RECONNECT_Q _MGR	MQCNO_RECONNECT_AS _DEF	MQCNO_RECONNECT_DIS ABLED
HAYIR	EVET	MMGR	HAYIR	HAYIR
EVET	EVET	MMGR	EVET	HAYIR
QMGR	EVET	MMGR	MMGR	HAYIR
DEVRE DIŞI	HAYIR	HAYIR	HAYIR	HAYIR

MQReconnectTimeout

İstemciye yeniden bağlanmak için saniye cinsinden zaman aşımı oluşturdu. Varsayılan değer 1800 saniyedir (30 dakika).

IBM WebSphere MQ classes for XMS .NET istemcileri, `XMSC.WMQ_CLIENT_RECONNECT_TIMEOUT` özelliğini kullanarak yeniden bağlanmak için bir zaman aşımı belirtebilirler. Bu özelliğe ilişkin varsayılan değer 1800 saniyedir (30 dakika).

ServerConnectionParms

ServerConnectionParms is equivalent to the MQSERVER environment parameter and specifies the location of the IBM WebSphere MQ server and the communication method to be used. ServerConnectionParms özneliği yalnızca basit bir kanal tanımlar; bir SSL kanalı ya da kanal çıkışları olan bir kanal tanımlamak için kullanamazsınız. Bu, *ChannelName/TransportType/ConnectionName* biçiminin bir dizisidir, *ConnectionName* tam olarak nitelenmiş bir ağ adı olmalıdır. *ChannelName*, eğik çizgi ("/") içeremez bu karakter, kanal adını, iletim tipini ve bağlantı adını ayırmak için kullanıldığı için karakterdir.

İstemci kanalını tanımlamak için ServerConnectionParms kullanıldığında, ileti uzunluğu üst sınırı 100 MB kullanılır. Bu nedenle, kanal için geçerli olan ileti büyüklüğü üst sınırı, sunucudaki SVRCONN kanalında belirtilen değerdir.

Yalnızca tek bir istemci kanalı bağlantısının yapılabileceğinin dikkat edilmesi gerekir. Örneğin, iki girişiniz varsa:

```
ServerConnectionParms=R1.SVRCONN/TCP/localhost(1963)
ServerConnectionParms=R2.SVRCONN/TCP/localhost(1863)
```

Yalnızca ikincinin kullanılması gerekir.

Specify *ConnectionName* as a comma-separated list of names for the stated transport type. Genel olarak, yalnızca bir ad gereklidir. Aynı özelliklerle birden çok bağlantı yapılandırmak için birden çok *anasistem adı* sağlayabilirsiniz. Bağlantılar, bağlantı başarıyla kuruluncaya kadar, bağlantı listesinde belirttikleri sırayla denir. Hiçbir bağlantı başarılı olmadıysa, istemci yeniden işlemeye başlar. Bağlantı listeleri, yeniden bağlanabilir istemcilere ilişkin bağlantıları yapılandırmak için kuyruk yöneticisi gruplarına bir alternatiftir.

Put1DefaultAlwaysSync=HAYIR|EVET

Controls the behavior of the MQPUT1 function call with the option MQPMO_RESPONSE_AS_Q_DEF.

HAYIR

If MQPUT1 is set with MQPMO_SYNCPOINT, it behaves as MQPMO_ASYNC_RESPONSE. Similarly, if MQPUT1 is set with MQPMO_NO_SYNCPOINT, it behaves as MQPMO_SYNC_RESPONSE. Bu varsayılan değerdir.

EVET

MQPUT1, MQPMO_SYNCPOINT ya da MQPMO_NO_SYNCPOINT ' un ayarlanıp ayarlanmamasından bağımsız olarak MQPMO_SYNC_RESPONSE ayarlıysa gibi davranır.

ClientExitPath stanza of the client configuration file

İstemciye kanal çıkışlarının varsayılan konumlarını belirtmek için ClientExitYol Stanzası 'nı kullanın.

ClientExitYol gösteriminde aşağıdaki öznitelikler de bulunabilir:

ExitsDefaultPath=dizgi

İstemciler için 32 bitlik kanal çıkışlarının yerini belirtir.

ExitsDefaultPath64=dizgi

İstemciler için 64 bitlik kanal çıkışlarının yerini belirtir.

JavaExitsClassPath=dizgi

Java çıkışı çalıştırıldığında sınıf yoluna eklenecek değerler. Bu, diğer herhangi bir dildeki çıkışlar tarafından dikkate alınmaz.

JMS yapılandırma dosyasında, JavaExitsClassPath adına standart com.ibm.mq.cfg adı verilir. örnek ve bu tam ad, Websphere MQ V7.0 sistem özelliğinde de kullanılır. At Version 6.0 this attribute was

specified using system property com.ibm.mq.exitClasspath, which was documented in the Version 6.0 readme. com.ibm.mq.exitClasspath kullanımını kullanımdan kaldırılmıştır. Hem JavaExitsClassPath , hem de exitClasspath varsa, JavaExitsClassPath onur edilir. Yalnızca exitClasspath kullanımı varsa, bu, Websphere MQ V7.0' da yine de onurlandırılır.

İstemci yapılandırma dosyasının LU62, NETBIOS ve SPX kısmı

Yalnızca Windows sistemlerinde, belirtilen ağ protokollerine ilişkin yapılanış değiştirgelerini belirtmek için bu stanzaları kullanın.

LU62

SNA LU 6.2 protokolü yapılanış değiştirgelerini belirtmek için LU62 kısmı kullanın. Aşağıdaki öznitelikler bu gösteriye dahil edilebilir:

Library1=DLLName| WCPIC32

APPC DLL ' in adı.

Library2=DLLName| WCPIC32

Kod iki ayrı kitaplıkta saklandıysa, Library1 ile aynıdır.

TPN Adı

Uzak yerde başlatılacak TP adı.

NETBIOS

Use the NETBIOS stanza to specify NetBIOS protocol configuration parameters. Aşağıdaki öznitelikler bu gösteriye dahil edilebilir:

AdapterNum=sayı| 0

LAN bağdaştırıcısının numarası.

Library1=DLLName| NETAPI32

NetBIOS DLL ' inin adı.

LocalName=ad

Bu bilgisayarın LAN üzerinde bilindiği ad.

Bu, MQNAME ortam parametresine eşdeğerdir.

NumCmds=sayı| 1

Ayrılacak kaç komut var.

NumSess=sayı| 1

Ayrılacak kaç oturum var.

SPX

SPX iletişim kuralı yapılandırma parametrelerini belirlemek için SPX Stanza 'yı kullanın. Aşağıdaki öznitelikler bu gösteriye dahil edilebilir:

BoardNum=sayı| 0

LAN bağdaştırıcısı numarası.

KeepAlive=YES|NO

KeepAlive işlevini açık ya da kapalı olarak değiştirin.

KeepAlive=YES (EVET) seçeneği, SPX 'in, bağlantının diğer ucunun hala kullanılabilir olup olmadığını düzenli aralıklarla denetmesine neden olur. Değilse, kanal kapatılır.

Library1=DLLName| WSOCK32.DLL

SPX DLL ' inin adı.

Library2=DLLName| WSOCK32.DLL

Kod iki ayrı kitaplıkta saklandıysa, Library1 ile aynıdır.

Socket=sayı| 5E86

Onaltılı gösterimde SPX yuvası sayısı.

İstemci yapılandırma dosyasının MessageBuffer kısmı

İleti arabelleklerine ilişkin bilgileri belirtmek için MessageBuffer kısmı kullanın.

MessageBuffer kısmına aşağıdaki öznitelikler eklenebilir:

MaximumSize=tamsayı|_1

Önden okuma arabelleğindeki büyüklük (KB), aralık 1-999 999).

Aşağıdaki özel değerler vardır:

-1

Uygun değeri istemci belirler.

0

İstemciye ilişkin okuma seçeneği geçersiz kılındı.

PurgeTime=tamsayı|_600

Önokuma arabelleğindeki iletilerin silineceği saniye cinsinden aralık.

İstemci uygulaması MsgId ya da CorrelId değerine dayalı iletileri seçiyorsa, ileriye okuma arabelleğindeki önceden istenen bir MsgId ya da CorrelId ile istemciye gönderilen iletiler içerilebilir. Bu iletiler, uygun bir MsgId ya da CorrelId ile bir MQGET komutu verilinceye kadar, ileriye doğru okuma arabelleğinde mahsur kalır. PurgeTime ayarını kullanarak, iletileri okuma arabelleğinden temizleyebilirsiniz. Temizleme aralığından daha uzun süre okuma arabelleğinde kalan iletiler otomatik olarak temizlenir. Bu iletiler kuyruk yöneticisinden zaten kuyruktan kaldırılmış, bu nedenle göz atılmadıkları sürece kaybedilir.

Geçerli değer aralığı 1-999 999 saniye arasındadır ya da özel değer 0, temizleme yapılmadığı anlamına gelir.

UpdatePercentage=tamsayı|_-1

Güncelleme yüzdesi değeri, bir istemci uygulamasının sunucuya ne zaman yeni bir istek yaptığını belirlemek için eşik değerini hesaplarken kullanılan 1-100 aralığında kullanılır. -1 özel değeri, istemcinin uygun değeri saptadığını gösterir.

İstemci düzenli olarak sunucuya, istemci uygulamasının ne kadar veri tükettiğini belirten bir istek gönderir. MQGET çağrılarında göre istemci tarafından alınan bayt sayısı n , bir eşik (T) aştığında bir istek gönderilir. Sunucuya her yeni bir istek gönderildiğinde, n sıfırlanır.

T eşik değeri aşağıdaki gibi hesaplanır:

$$T = Upper - Lower$$

Üst öge, *MaximumSize* özniteliği tarafından (kilobayt cinsinden) belirtilen, önden okuma arabelleği büyüklüğüyle aynıdır. Varsayılan değer 100 Kb 'dir.

Daha düşük, Üst ' ten daha düşüktür ve *UpdatePercentage* özniteliği tarafından belirtilir. Bu öznitelik, 1-100 aralığında bir sayıdır ve varsayılan değeri 20 'dir. Alt öge aşağıdaki gibi hesaplanır:

$$Lower = Upper \times UpdatePercentage / 100$$

Örnek 1:

MaximumSize ve UpdatePercentage öznitelikleri, 100 Kb ve 20 Kb varsayılan değerlerini alır.

İstemci, bir iletiyi almak için MQGET ' yi çağırır ve bunu sürekli olarak yapar. Bu, MQGET n byte ' ı tüketinceye kadar devam eder.

Hesaplamayı kullanma

$$T = Upper - Lower$$

T (100-20) = 80 Kb 'dir.

Bu nedenle, MQGET çağrıları 80 Kb ' i bir kuyruktan kaldırdığında, istemci otomatik olarak yeni bir istek yapar.

Örnek 2:

MaximumSize öznitelikleri, varsayılan 100 Kb değerini alır ve UpdatePercentage için 40 değerini alır.

İstemci, bir iletiyi almak için MQGET ' yi çağırır ve bunu sürekli olarak yapar. Bu, MQGET n byte 'ı tüketinceye kadar devam eder.

Hesaplamayı kullanma

```
T = Upper - Lower
```

$T (100-40) = 60$ Kb 'dir

Bu nedenle, MQGET çağrıları 60 Kb 'i kuyruktan kaldırdığında, istemci otomatik olarak yeni bir istek yapar. Bu, ÖRNEK 1 'den daha önce, varsayılan değerlerin kullanıldığı bir yer değildir.

Bu nedenle, daha büyük bir eşik (T) seçilmesi, isteklerin istemciden sunucuya gönderilme sıklığını azaltmaya neden olur. Conversely choosing a smaller threshold S tends to increase the frequency of requests that are sent from client to server.

Ancak, büyük bir eşik T seçilmesi, ileriye doğru okuma arabelleğinin boş olma olasılığı artabileceği için, ileriye doğru okuma değerinin azaltılmış olması anlamına gelir. Bu durumda bir MQGET çağrısının duraklatılması gerekebilir; bu durum, sunucudan veri gelmesini bekliyor olabilir.

İstemci yapılandırma dosyasının SSL kısmı

SSL ya da TLS kullanımına ilişkin bilgileri belirtmek için SSL standı kullanın.

SSL kısmına aşağıdaki öznitelikler eklenebilir:

CDPCheckExtensions=YES|NO

CDPCheckExtensions , bu kuyruk yöneticisinde SSL ya da TLS kanallarının, CrlDistributionPoint sertifika uzantılarında adı geçen CDP sunucularını denetleyip denetmeyeceğini belirtir.

Bu öznitelik aşağıdaki olası değerlere sahiptir:

- YES: SSL ya da TLS kanalları, bir dijital sertifikana geri verilip verilmediğini belirlemek için CDP sunucularını denetleyip denetmeye çalışır.
- NO: SSL ya da TLS kanalları, CDP sunucularını denetlimeyi denemez. Bu, varsayılan değerdir.

CertificateLabel = *dizgi*

Kanal tanımlamasının sertifika etiketi.

Bu öznitelik C tarafından okunabilir ve yönetilmeyen .NET istemcileri tarafından okunabilir.

CertificateValPolicy=*dizgi*

Kullanılan sertifika geçerlilik denetimi tipini belirler.

HERHANGİ BİRİ

Temeldeki güvenli yuva kitaplığı tarafından desteklenen herhangi bir sertifika geçerlilik denetimi ilkesini kullanın. Bu ayar, varsayılan ayardır.

RFC5280

Yalnızca RFC 5280 standardına uygun olan sertifika geçerlilik denetimini kullanın.

ClientRevocationChecks = **REQUIRED | **OPTIONAL** | **DISABLED****

İstemci bağlanma çağrısı bir SSL/TLS kanalı kullanıyorsa, sertifika iptal denetiminin nasıl yapılandırılıp yapılandırılacağını belirler. Ayrıca bkz. [OCSPAuthentication](#).

Bu öznitelik C tarafından okunabilir ve yönetilmeyen .NET istemcileri tarafından okunabilir.

Bu öznitelik aşağıdaki olası değerlere sahiptir:

GEREKLI (varsayılan)

CCDT ' den sertifika iptal yapılandırmasını yükleme ve yapılandırılmış olarak iptal denetimini gerçekleştirme girişimleri. CCDT dosyası açılmazsa ya da sertifikenin geçerliliğini denetlemek olanaklı değilse (örneğin, bir OCSP ya da CRL sunucusu kullanılmadığından) MQCONN çağrısı

başarısız olur. CCDT ' nin iptal yapılandırması içermiyorsa, iptal denetimi gerçekleştirilmez; ancak bu işlem kanala hata oluşmasına neden olmaz.

Windows Windows sistemlerinde, CRL iptal denetimi için Active Directory seçeneğini de kullanabilirsiniz. OCSP iptal denetimi için Active Directory seçeneğini kullanamazsınız.

İsteğe Bağlı

REQUIREMSolarak, ancak sertifika iptal yapılandırmasını yüklemek olanaklı değilse, kanal başarısız olmaz.

DEVRE DIŞI

CCDT ' den sertifika iptal yapılandırması yükleme girişiminde bulunulmadı ve sertifika iptal denetimi gerçekleştirilmedi.

Not: MQCONN çağrıları yerine MQCONNX kullanıyorsanız, MQSCO aracılığıyla kimlik doğrulama bilgisi kayıtları (MQAIR) sağlamanız gerekebilir. MQCONNX ile varsayılan davranış, CCDT dosyası açılmazsa, ancak bir MQAIR sağladığınızı varsaymak için (bunu yapmamayı seçmiş olsanız bile) başarısız olmak için bu işlev başarısız olur.

EncryptionPolicySuiteB=dizgi

Bir kanalın Suite-B uyumlu şifreleme kullanıp kullanmayacağını ve hangi düzeyde kalınlık kullanılacağını belirler. Olası değerler şunlardır:

YOK

Suite-B uyumlu şifreleme kullanılmıyor. Bu ayar, varsayılan ayardır.

128_BIT,192_BIT

Güvenlik gücünü hem 128 bitlik, hem de 192 bit düzeylere ayarlar.

128_BIT

Güvenlik gücünü 128 bit düzeyine ayarlar.

192_BIT

Güvenlik gücünü 192 bit düzeyine ayarlar.

OCSPAuthentication=OPTIONAL| REQUIRED|WARN

Defines the behavior of WebSphere MQ when OCSP is enabled and the OCSP revocation check is unable to determine the certificate revocation status. Üç olası değer vardır:

İsteğe Bağlı

OCSP denetimi tarafından belirlenemeyen bir iptal durumu olan tüm sertifikalar kabul edilir ve hiçbir uyarı ya da hata iletisi oluşturulamaz. SSL ya da TLS bağlantısı, herhangi bir iptal denetimi yapılmıyorsa gibi devam eder.

ZORUNLU

OCSP denetimi, denetlenen her SSL ya da TLS sertifikası için kesin bir iptal sonucu vermelidir. Geri alma durumuna doğrulanamayan herhangi bir SSL ya da TLS sertifikası, bir hata iletisiyle reddedilir. Kuyruk yöneticisi SSL olay iletileri etkinleştirilirse, MQRC_SSL_HANDSHAKE_ERROR ReasonQualifier ile bir MQRC_CHANNEL_SSL_ERROR iletisi üretilir. Bağlantı kapatıldı.

Bu değer, varsayılan değerdir.

WARN

OCSP iptal denetimi, herhangi bir SSL ya da TLS sertifikasının iptal durumunu belirleyemezse, kuyruk yöneticisi hata günlüklerinde bir uyarı bildirilir. Kuyruk yöneticisi SSL olay iletileri etkinleştirilirse, MQRC_SSL_UNKNOWN_REVOCATION ReasonQualifier adlı bir MQRC_CHANNEL_SSL_UYARI iletisi oluşturulur. Bağlantının devam etmesine izin verilir

OCSPCheckExtensions=YES|NO

WebSphere MQ ' un AuthorityInfoErişimi sertifikası uzantılarına göre hareket edip etmeyeceğini denetler. Değer NO(Hayır) olarak ayarlanırsa, WebSphere MQ AuthorityInfoErişim sertifikası uzantılarını yoksayar ve bir OCSP güvenlik denetimi girişiminde bulunmaz. Varsayılan değer YES(EVET) değeridir.

SSLCryptoHardware=dizgi

Sistemde PKCS cryptographic şifreleme donanımını yapılandırmak için gereken parametre dizilimini ayarlar.

Şu biçimde bir dizgi belirtin: GSK_PKCS11=*sürücü yolu ve dosya adı*; *simge etiketi*; *simge parolası*; *simetrik şifre ayarı*;

Örneğin: GSK_PKCS11=/usr/lib/pkcs11/
PKCS11_API.so;tokenlabel;passwd;SYMMETRIC_CIPHER_ON

Sürücü yolu, PKCS #11 kartı için destek sağlayan paylaşılan kitaplık için mutlak bir yoldur. Sürücü dosyası adı, paylaşılan kitaplığın adıdır. PKCS #11 sürücü yolu ve dosya adı için gereken değer örneği: /usr/lib/pkcs11/PKCS11_API.so. GSKit üzerinden simetrik şifre işlemlerine erişmek için simetrik şifre ayarı parametresini belirtin. Bu parametrenin değeri aşağıdakilerden biri olabilir:

SIMETRIC_CIPHER_OFF

Simetrik şifre işlemlerine erişmeyin. Bu ayar, varsayılan ayardır.

SIMETRIC_CIPHER_ON

Simetrik şifre işlemlerine erişir.

Dizilimin uzunluk üst sınırı 256 karakterdir. Varsayılan değer boştur. Doğru biçimde olmayan bir dizilim belirlerseniz, hata ortaya çıktı.

SSLFipsRequired=YES|_NO

Specifies whether only FIPS-certified algorithms are to be used if cryptography is carried out in WebSphere MQ. Şifreleme donanımı yapılandırıldıysa, kullanılan şifreleme modülleri, donanım ürünü tarafından sağlanan modüllerdir. Bu, donanım ürününe bağlı olarak, belirli bir düzey için FIPS onaylı olabilir ya da olmayabilir.

SSLHTTPProxyName=dizgi

Bu dizgi, GSKit for OSCP denetimlerini kullanarak kullanılacak HTTP yetkili sunucusunun anasistem adı ya da ağ adresidir. Bu adres, isteğe bağlı bir kapı numarası ve parantez içine alınmış olarak izlenebilir. Kapı numarasını belirtmezseniz, varsayılan HTTP kapısı 80 olarak kullanılır. HP-UX PA-RISC ve Sun Solaris SPARC altyapılarında ve AIX üzerindeki 32 bit istemciler için, ağ adresi yalnızca bir IPv4 adresi olabilir; diğer altyapılarda IPv4 ya da IPv6 adresi olabilir.

Örneğin, bir güvenlik duvarı OSCP yanıtlayıcıya erişimi engelliyorsa, bu öznelik gerekli olabilir.

SSLKeyRepository=yol adı

Kullanıcının dijital sertifikasını kök biçiminde tutan anahtar havuzunun konumu. Yani, tam yolu ve uzantısız dosya adını içerir.

SSLKeyResetCount=tamsayı|_0

Gizli anahtar yeniden anlaşılmadan önce bir SSL ya da TLS kanalına gönderilen ve alınan şifrelenmemiş baytların sayısı.

Değer 0-999999999 aralığında olmalıdır.

Varsayılan değer 0 'tır, bu da gizli anahtarların hiçbir zaman yeniden görüşmemekte olduğunu gösterir.

1-32768 arasında bir değer belirtirseniz, SSL ya da TLS kanalları, 32768 (32Kb) gizli anahtar sıfırlama sayısını kullanır. Bu, küçük gizli anahtar ilk duruma getirme değerleri için oluşacağı aşırı anahtar sıfırlamalarından kaçınmak içindir.

İstemci yapılandırma dosyasının TCP kısmı

TCP ağ iletişim kuralı yapılandırma parametrelerini belirlemek için TCP stanzını kullanın.

TCP Stanza 'ya aşağıdaki öznelikler eklenebilir:

ClntRcvBuffSize=sayı|_32768

İstemci-bağlantı sunucusu bağlantı kanalının istemci ucu tarafından kullanılan TCP/IP alma arabelleğindeki bayt cinsinden büyüklük. A value of zero indicates that the operating system will manage the buffer sizes, as opposed to the buffer sizes being fixed by WebSphere MQ.

ClntSndBuffSize=sayı|_32768

İstemci-bağlantı sunucusu bağlantı kanalının istemci ucu tarafından kullanılan TCP/IP gönderme arabelleğinin bayt cinsinden büyüklüğü. A value of zero indicates that the operating system will manage the buffer sizes, as opposed to the buffer sizes being fixed by WebSphere MQ.

Connect_Timeout=sayı

Yuva saatlerini dışarı bağlama girişiminden önceki saniye sayısı; kanal sıfır olmayan bir istemci kanalı ağırlıklandırmasıyla yapılandırılmadıkça, varsayılan değer 0 olur. Varsayılan değer, varsayılan değer 5 'dir.

IPAddressVersion=MQIPADDR_IPV4|MQIPADDR_IPV6

Bir kanal bağlantısı için hangi IP protokollerinin kullanılacağını belirler.

Olası dizgi değerleri MQIPADDR_IPV4 ya da MQIPADDR_IPV6 olabilir. Bu değerler, **ALTER QMGR IPADDRV** 'de IPV4 ve IPV6 ile aynı anlamlara sahiptir.

KeepAlive=YES|NO

KeepAlive işlevini açık ya da kapalı olarak değiştirin. KeepAlive=YES, TCP/IP ' nin bağlantının diğer ucunun hala kullanılabilir durumda olduğunu düzenli aralıklarla denetmesine neden olur. Değilse, kanal kapatılır.

Windows Library1=DLLName|_WSOCK32

(Yalnızca Windows) TCP/IP yuvaları DLL ' inin adı.

TMF ve TMF/Gateway stanzaları

The IBM WebSphere MQ provided TMF/Gateway runs in a Pathway environment. Use the TMF and TMF/Gateway stanzas to specify the required configuration parameters for the IBM WebSphere MQ client for HP Integrity NonStop Server to communicate with the TMF/Gateway.

TMF kullanmak istiyorsanız, iletişim kurduğunuz her kuyruk yöneticisi için bir TMF stanza ve bir TmfGateway stanza tanımlamanız gerekir. Tüm değerler yapılandırılardan türetilir.

TMF kısmı

PathMon=ad

TMF/Ağ Geçidi için sunucu sınıflarını tanımlayan tanımlı Pathmon sürecinizin adı.

TmfGateway stanza

Aşağıdaki öznitelikler bu gösteriye dahil edilebilir:

QManager=ad

Kuyruk yöneticisinin adı.

Server=ad

Bu kuyruk yöneticisi için yapılandırılan TMF/Gateway için sunucu sınıfı adı.

Örnek

Aşağıda, farklı sunucularda iki farklı kuyruk yöneticisi için iki TmfGateway stanzasıyla tanımlanmış bir TMF stanza örneği gösterilmektedir:

```
TMF:
  PathMon=$PSD1P

TmfGateway:
  QManager=MQ5B
  Server=MQ-MQ5B

TmfGateway:
  QManager=MQ5C
  Server=MQ-MQ5C
```

WebSphere MQ ortam deęişkenlerinin kullanılması

Bu kısımda, WebSphere MQ MQI istemci uygulamalarıyla kullanabileceğiniz ortam deęişkenleri açıklanmaktadır.

Ortam deęişkenlerini aşağıdaki şekillerde kullanabilirsiniz:

- Kalıcı bir deęişiklik yapmak için sistem profilinizde deęişkenleri ayarlayın
- Yalnızca bu oturum için bir deęişiklik yapmak üzere komut satırından bir komut verin
- Bir ya da daha çok deęişkene, çalışmakta olan uygulamaya baęımlı bir deęer vermek için, uygulama tarafından kullanılan komut dosyasına komutlar ekleyin.

WebSphere MQ , ayarlamadığınız deęişkenler için varsayılan deęerleri kullanır.

Tersi belirtilmedięi sürece, komutlar tüm WebSphere MQ MQI istemcisi altyapılarında kullanılabilir.

Her ortam deęişkeni için, yürürlükteki ayarı görüntülemek ya da bir deęişkenin deęerini ilk duruma getirmek için altyapınızla ilgili komutu kullanın.

Örneğin:

Bir ortam deęişkeninin deęerini ayarlama ya da ilk durumuna getirme		
Efekt	Komut	
	Pencereler	UNIX and Linux sistemleri
Deęişkeni kaldırır	SET MQSERVER=	MQSERVER ayarı kaldır
Geçerli ayarı görüntüler	MQSERVER	echo \$MQSERVER
Oturuma ilişkin tüm ortam deęişkenlerini görüntüler.	küme	küme

Tek tek deęişkenlerle ilgili bilgi için aşağıdaki alt başlıklara bakın:

İlgili kavramlar

[“Yapılandırma dosyası kullanarak istemci yapılandırılması” sayfa 124](#)

Bir metin dosyasındaki öznitelikleri kullanarak istemcilerinizi yapılandırın. Bu öznitelikler, ortam deęişkenleriyle ya da altyapıya özgü dięer yollarla geçersiz kılınabilir.

İlgili başvurular

[Ortam deęişkenleri](#)

MQCCSID

MQCCSID, kullanılacak kodlanmış karakter takımı numarasını belirler ve sunucunun yapılandırıldığı CCSID deęerini geçersiz kılar.

Ek bilgi için [İstemci ya da sunucu kodlanmış karakter takımı tanıtıcısı \(CCSID\) seçilmesi](#) başlıklı konuya bakın.

Bu deęişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows için:

```
SET MQCCSID=number
```

- UNIX and Linux sistemleri için:

```
export MQCCSID=number
```

MQCERTVPOL

MQCERTVPOL, kullanılan sertifika geçerlilik denetimi ilkesini belirtir.

WebSphere MQ' da sertifika geçerlilik denetimi ilkeleriyle ilgili daha fazla bilgi için [WebSphereMQ](#) içindeki [CertificateSertifika doğrulama ilkeleri](#). konusuna bakın.

Bu ortam değişkeni, istemci ini dosyasının SSL kısmında bulunan *CertificateValPolicy* ayarını geçersiz kılar. Değişken şu iki değerden birine ayarlanabilir:

HERHANGİ BİRİ

Temeldeki güvenli yuva kitaplığı tarafından desteklenen herhangi bir sertifika geçerlilik denetimi ilkesini kullanın.

RFC5280

Yalnızca RFC 5280 standardına uygun olan sertifika geçerlilik denetimini kullanın.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows için:

```
SET MQCERTVPOL=value
```

- UNIX and Linux sistemleri için:

```
export MQCERTVPOL=value
```

MQCHLIB

MQCHLLIB , istemci kanal tanımlama çizelgesini (CCDT) içeren dosyaya ilişkin izin yolunu belirtir. The file is created on the server, but can be copied across to the WebSphere MQ MQI client workstation.

MQCHLLIB belirlenmezse, istemcinin varsayılan yolu şu değerlere ayarlanır:

- **Windows** Pencereleri için: *MQ_INSTALLATION_PATH*
- **UNIX** **Linux** UNIX and Linux sistemleri için: */var/mqm/*

crtmqm ve *strmqm* komutları için, yol varsayılan olarak iki yol kümelerinden birine ayarlanır. *datapath* ayarlandıysa, yol varsayılan olarak ilk ayardan birine ayarlanır. *datapath* ayarlanmazsa, yol varsayılan olarak ikinci ayardan birine ayarlanır.

- **Windows** Pencereleri için: *datapath\@ipcc*
- **UNIX** **Linux** For UNIX and Linux systems: *datapath/@ipcc*

Ya da:

- **Windows** Pencereleri için: *MQ_INSTALLATION_PATH\data\qmgrs\qmgrname\@ipcc*
- **UNIX** **Linux** UNIX and Linux sistemleri için: */prefix/qmgrs/qmgrname/@ipcc*

Burada:

- *MQ_INSTALLATION_PATH* , IBM WebSphere MQ ' in kurulu olduğu üst düzey dizini temsil eder.
- Varsa, *datapath* , kuyruk yöneticisi stanza içinde tanımlanan DataPath değeridir.
- *prefix* , kuyruk yöneticisi stanzasında tanımlanan önek değeridir. Prefix is typically */var/mqm* on UNIX and Linux platforms.
- *qmgrname* , kuyruk yöneticisi stanzasında tanımlı olan Directory özniteliğinin değeridir. Değer, gerçek kuyruk yöneticisi adından farklı olabilir. Değer, özel karakterlerin yerine konması için değiştirilmiş olabilir.
- The queue manager stanza is defined in the *mq.s.ini* file on UNIX, and Linux, and in the registry on Pencereler

Notlar:

1. Ayarlanırsa, MQCHLLIB CCDT ' yi bulmak için kullanılan yolu geçersiz kılar.

2. Environment variables, such as MQCHLLIB, can be scoped to a process, or a job, or system-wide, in a platform-specific way.
3. MQCHLLIB sistemini bir sunucuya ayarlıyorsanız, sunucudaki tüm kuyruk yöneticileri için aynı yolu CCDT dosyası ile ayarlar. MQCHLLIB ortam değişkenini ayarlamadıysanız, yol her kuyruk yöneticisi için farklı olur. Queue managers read the value of MQCHLLIB, if it is set, on either the **crtmqm** or **strmqm** command.
4. Bir sunucuda birden çok kuyruk yöneticisi oluşturursanız, aşağıdaki nedenden dolayı ayırım önemli olur. MQCHLLIB sistem genelini ayarlıyorsanız, her kuyruk yöneticisi aynı CCDT dosyasını güncelleştirir. Dosya, sunucudaki tüm kuyruk yöneticilerinden istemci-bağlantı tanımlarını içerir. Aynı tanımlama birden çok kuyruk yöneticisi üzerinde varsa, SYSTEM . DEF . CLNTCONN örneğin, dosya en son tanımı içerir. Bir kuyruk yöneticisi yarattığınızda, MQCHLLIB ayarlandıysa, CCDT ' de SYSTEM . DEF . CLNTCONN güncellenir. Güncelleme, farklı bir kuyruk yöneticisi tarafından yaratılan SYSTEM . DEF . CLNTCONN ' in üzerine yazılıyor. Önceki tanımlamayı değiştirdiyseniz, değişiklikleriniz kaybedilir. For this reason, you must consider finding alternatives to setting MQCHLLIB as a system-wide environment variable on the server.
5. İstemci-bağlantı tanımlamasındaki MQSC ve PCF NOREPLACE seçeneği, CCDT dosyasının içeriğini denetlemez. NOREPLACE seçeneği ne olursa olsun, bu kuyruk yöneticisi tarafından değil, önceden yaratılmış, ancak bu kuyruk yöneticisi tarafından olmayan bir istemci-bağlantı kanalı tanımlaması başkasıyla değiştirilir. Tanımlama daha önce aynı kuyruk yöneticisi tarafından yaratıldıysa, tanımlama değiştirilmez.
6. Komut, **rcrmqobj** -t c1chl1tab CCDT dosyasını siler ve yeniden yaratır. Bu kütük, komutun üzerinde çalışmakta olduğu kuyruk yöneticisinden yalnızca istemci bağlantısı tanımlamalarıyla yaratılır.
7. CCDT ' yi güncelleyen diğer komutlar yalnızca aynı kanal adına sahip istemci-bağlantı kanallarını değiştirir. Kütükteki diğer istemci bağlantısı kanalları değiştirilmez.
8. MQCHLIB yolunun tırnak işareti (quote) işaretlerine gerek yoktur.

Örnekler

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- **Windows** Windows için:

```
SET MQCHLLIB=pathname
```

Örneğin:

```
SET MQCHLLIB=C:\wmqtest
```

- **UNIX** **Linux** UNIX and Linux sistemleri için:

```
export MQCHLLIB=pathname
```

MQCHLTAB

MQCHLTAB, istemci kanal tanımlama çizelgesini (ccdt) içeren dosyanın adını belirtir. Varsayılan dosya adı AMQCLCHL.TAB' dir.

Bir sunucuda istemci kanal tanımlama tablolarının nerede bulunduğu hakkında bilgi için bkz. "[İstemci kanal tanımlama çizelgesi](#)" sayfa 114.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows'ta:

```
SET MQCHLTAB=filename
```


- UNIX and Linux sistemlerinde:

```
export MQCHLTAB=filename
```

Örneğin:

```
SET MQCHLTAB=ccdf1.tab
```

İstemci ile aynı şekilde, sunucudaki MQCHLTAB ortam değişkeni, istemci kanal tanımlama çizelgesinin adını belirtir.

MQIPADDRV

MQIPADDRV, bir kanal bağlantısı için hangi IP protokollerinin kullanılacağını belirtir. Olası dizgi değerleri "MQIPADDR_IPV4" ya da "MQIPADDR_IPV6" olabilir. Bu değerler, ALTER QMGR IPADDRV ' de IPV4 ve IPV6 ile aynı anlamlara sahiptir. Ayarlanmazsa, "MQIPADDR_IPV4" varsayıldır.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows için:

```
SET MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6
```

- UNIX and Linux sistemleri için:

```
export MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6
```

MQNAME

MQNAME, WebSphere MQ işlemlerinin kullanabildiği yerel NetBIOS adını belirtir.

Tam açıklama ve istemciye ve sunucuya ilişkin öncelik kuralları için [“Windows üzerinde NetBIOS bağlantısının tanımlanması” sayfa 85](#) başlıklı konuya bakın.

Bu değişkeni ayarlamak için şu komutu kullanın:

```
SET MQNAME=Your_env_Name
```

Örneğin:

```
SET MQNAME=CLIENT1
```

The NetBIOS on some platforms requires a different name (set by MQNAME) for each application if you are running multiple WebSphere MQ applications simultaneously on the WebSphere MQ MQI client.

MQSERVER

MQSERVER ortam değişkeni, en alt düzeyde kanal tanımlamak için kullanılır. MQSERVER, kullanılacak WebSphere MQ Server sunucusunun ve kullanılacak iletişim yönteminin yerini belirtir.

Bir SSL kanalı ya da kanal çıkışları olan bir kanal tanımlamak için MQSERVER ' ı kullanamazsınız. SSL kanalı tanımlamaya ilişkin ayrıntılar için [SSL ile kanalları korumabaşlıklı konuya](#) bakın.

ConnectionName tam olarak nitelenmiş bir ağ adı olmalıdır. Bu karakter kanal adını, iletim tipini ve bağlantı adını ayırmak için kullanıldığından *ChannelName* , eğik çizgi (/) karakterini içeremez. Bir istemci kanalını tanımlamak için MQSERVER ortam değişkeni kullanılırsa, 100 MB ' lik bir ileti uzunluğu üst sınırı (MAXMSGL) kullanılır. Bu nedenle, kanal için geçerli olan ileti büyüklüğü üst sınırı, sunucudaki SVRCONN kanalında belirtilen değerdir.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows için:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName
```

- UNIX and Linux sistemleri için:

```
export MQSERVER='ChannelName/TransportType/ConnectionName'
```

TransportType can be one of the following values, depending on your IBM WebSphere MQ client platform:

- LU62
- TCP
- NETBIOS
- SPX

ConnectionName , bağlantı adlarından oluşan virgülle ayrılmış bir liste olabilir. Listedeki bağlantı adları, istemci bağlantı çizelgesinde birden çok bağlantıya benzer bir şekilde kullanılır. The *ConnectionName* list might be used as an alternative to queue manager groups to specify multiple connections for the client to try. Çok eşgörümlü bir kuyruk yöneticisi yapılandırıyorsanız, farklı kuyruk yöneticisi eşgörümleri belirtmek için bir *ConnectionName* listesi kullanabilirsiniz.

TCP/IP varsayılan kapısı

Varsayılan olarak, TCP/IP için WebSphere MQ , kanalın 1414 numaralı kapıya bağlandığını varsayar.

Bunu aşağıdaki gibi değiştirebilirsiniz:

- Kapı numarasının, ConnectionName ürününün son parçası olarak köşeli parantez içine eklenmesi:

- Windows için:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(PortNumber)
```

- UNIX and Linux sistemleri için:

```
export MQSERVER='ChannelName/TransportType/ConnectionName(PortNumber)'
```

- Protokol adına kapı numarası eklenerek `mqclient.ini` dosyasının değiştirilmesi, örneğin:

```
TCP:  
port=2001
```

- Adding WebSphere MQ to the services file as described in [“TCP/IP dinleyicisinin kullanılması” sayfa 92.](#)

SPX varsayılan yuvası

Varsayılan olarak, SPX için, WebSphere MQ , kanalın 5E86 yuvasına bağlanacağını varsayar.

Bunu aşağıdaki gibi değiştirebilirsiniz:

- Yuva numarasını, ConnectionName ürününün son parçası olarak köşeli ayraç içine ekleme:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(SocketNumber)
```

SPX bağlantıları için `network.node(socket)` biçiminde *ConnectionName* ve yuvayı belirtin. WebSphere MQ istemcisi ve sunucusu aynı ağdaysa, ağ belirtilmemesi gerekir. Varsayılan yuva kullanıyorsanız, yuva belirlenmemesi gerekir.

- Protokol adına kapı numarası eklenerek `qm.ini` dosyasının değiştirilmesi, örneğin:

```
SPX:  
socket=5E87
```

MQSERVER olanağının kullanılması

If you use the MQSERVER environment variable to define the channel between your WebSphere MQ MQI client machine and a server machine, this is the only channel available to your application, and no reference is made to the client channel definition table (CCDT).

Bu durumda, sunucu makinesinde çalıştırdığınız dinleyici programı uygulamanızın bağlanacağı kuyruk yöneticisini belirler. Dinleyici programının bağlı olduğu kuyruk yöneticisi aynı olacaktır.

If the MQCONN or MQCONNX request specifies a queue manager other than the one the listener is connected to, or if the MQSERVER parameter *TransportType* is not recognized, the MQCONN or MQCONNX request fails with return code MQRC_Q_MGR_NAME_ERROR.

UNIX and Linux sistemlerinde, MQSERVER 'ı aşağıdaki örneklerden birine göre tanımlayabilirsiniz:

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56(2002)'  
export MQSERVER=CHANNEL1/LU62/BOX99
```

Tüm MQCONN ya da MQCONNX istekleri, MQCONNX' a sağlanan MQCNO yapısından bir MQCD yapısına gönderme yapılmadığı sürece tanımladığınız kanalı kullanmayı dener; bu durumda, MQASCD yapısı tarafından belirtilen kanal, MQSERVER ortam değişkeninin belirlediği herhangi bir değişkene göre öncelik alır.

MQSERVER ortam değişkeni, MQCHLLIB ve MQCHLTAB ' ın gösterdiği tüm istemci kanalı tanımlamalarına göre öncelik alır.

MQSERVER iptal ediyor

MQSERVER 'ı iptal etmek ve MQCHLLIB ve MQCHLTAB tarafından gösterilen istemci kanal tanımlama çizelgesine dönmek için, aşağıdaki bilgileri girin:

- Pencerelelerinde:

```
SET MQSERVER=
```

- UNIX and Linux sistemlerinde:

```
unset MQSERVER
```

MQSSLCRYP

MQSSLCRYP, sistemde bulunan şifreleme donanımını yapılandırmanıza olanak tanıyan bir parametre dizisi içerir. İzin verilen değerler, ALTER QMGR komutunun SSLCRYP parametresiyle aynıdır.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows sistemlerinde:

```
SET MQSSLCRYP=string
```

- UNIX and Linux sistemlerinde:

```
export MQSSLCRYP=string
```

İlgili başvurular

ALTER QMGR komutu **SSLCRYP** parametresi

MQSSLFIPS

MQSSLFIPS specifies whether only FIPS-certified algorithms are to be used if cryptography is carried out in WebSphere MQ. Değerler, ALTER QMGR komutunun SSLFIPS parametresiyle aynıdır.

FIPS onaylı algoritmaların kullanımı, şifreleme donanımını kullandığından etkilenir ve MQI istemcisinde çalıştırma sırasında yalnızca FIPS onaylı CipherSpecs ' in kullanıldığını belirtme.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows sistemlerinde:

```
SET MQSSLFIPS=YES|NO
```

- UNIX and Linux sistemlerinde:

```
export MQSSLFIPS=YES|NO
```

Varsayılan değer NO' dir.

MQSSLKEYR

MQSSLKEYR, kullanıcının kök biçiminde, kullanıcıya ait dijital sertifikayı bulduran anahtar havuzunun yerini belirtir. Kök biçimi, tam yolu ve dosya adını uzantısız olarak içereceği anlamına gelir. Tüm ayrıntılar için, ALTER QMGR komutunun SSLKEYR parametresine bakın.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows sistemlerinde:

```
SET MQSSLKEYR=pathname
```

- UNIX and Linux sistemlerinde:

```
export MQSSLKEYR=pathname
```

Varsayılan değer yoktur.

MQSSLPROXY

MQSSLPROXY, GSKit for OCSP denetimlerini kullanarak kullanılacak HTTP yetkili sunucusunun anasistem adını ve kapı numarasını belirtir.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows sistemlerinde:

```
SET MQSSLPROXY=string
```

- UNIX and Linux sistemlerinde:

```
export MQSSLPROXY="string"
```

Bu dizgi, GSKit for OCSP denetimleri tarafından kullanılacak olan HTTP Yetkili sunucusunun anasistem adı ya da ağ adresidir. Bu adres, isteğe bağlı bir kapı numarası ve parantez içine alınmış olarak izlenebilir. Kapı numarasını belirtmezseniz, varsayılan HTTP kapısı 80 olarak kullanılır.

Örneğin, UNIX and Linux sistemlerinde, aşağıdaki komutlardan birini kullanabilirsiniz:

- ```
export MQSSLPROXY="proxy.example.com(80) "
```

- ```
export MQSSLPROXY="127.0.0.1"
```

MQSSLRESET

MQSSLRESET, gizli anahtar yeniden anlaşılmadan önce bir SSL ya da TLS kanalına gönderilen ve alınan şifrelenmemiş baytların sayısını gösterir.

Gizli anahtar yeniden görüşmeyle ilgili ek bilgi için [SSL ve TLS gizli anahtarlarının sıfırlanması](#) başlıklı konuya bakın.

0 ile 999 999 999 aralığında bir tamsayıya ayarlanabilir. Varsayılan değer 0, gizli anahtarların hiçbir zaman yeniden anlaşma vermediğini gösterir. SSL ya da TLS gizli anahtar sıfırlama sayısı, 1 bayt-32 KB arasında bir değer belirlerseniz, SSL ya da TLS kanalları, 32 KB 'lık bir gizli anahtar sıfırlama sayısı kullanır. Bu gizli ilk duruma getirme sayısı, küçük SSL ya da TLS gizli anahtar ilk duruma getirme değerleri için oluşacağı aşırı anahtar sıfırlamalarından kaçınmamaktadır.

Bu değişkeni ayarlamak için şu komutlardan birini kullanın:

- Windows sistemlerinde:

```
SET MQSSLRESET=integer
```

- UNIX and Linux sistemlerinde:

```
export MQSSLRESET=integer
```

Kuyruğa alınan yayınlama/abone olma denetimi

Kuyruğa yollanmış yayınlama/abone olma durumlarının durumunu başlatabilir, durdurabilir ve görüntüleyebilirsiniz. Ayrıca, akışları ekleyebilir ve kaldırabilir, bir aracı sıradüzeninden kuyruk yöneticileri ekleyebilir ve silebilirsiniz.

Kuyruğa alınan yayınlama/abone olma bilgilerinin denetlenmesine ilişkin ek bilgi için aşağıdaki alt başlıklara bakın:

Kuyruğa alınan yayınlama/abone olma iletisi özniteliklerini ayarlama

Kuyruk yöneticisi özniteliklerini kullanarak bazı yayınlama/abone olma ileti özniteliklerinin davranışını denetliyorsunuz. Diğer öznitelikler, `qm.ini` dosyasının *Aracı* kısmında yer alan bir diğer özniteliklere sahip olur.

Bu görev hakkında

Şu yayınlama/abone olma özniteliklerini ayarlayabilirsiniz: ayrıntılar için bkz. [Kuyruk yöneticisi değiştirgeleri](#)

Çizelge 23. Yapılandırma parametrelerini yayınlama/abone ol	
Tanım	MQSC değiştirgesi adı
Komut iletisi yeniden deneme sayısı	PSRTCNT
Teslim edilemeyen komut giriş iletisini at	PSNPMMSG
Teslim edilemeyen komut yanıt iletisini izleyen davranış	PSNPRES
Komut iletilerini eşitleme noktası altında işle	PSSYNCPT

Aracı kısmı, aşağıdaki yapılandırma ayarlarını yönetmek için kullanılır:

- `PersistentPublishRetry=yes | force`

Evetseçeneğini belirlerseniz, kuyruğa alınan yayınlama/abone olma arabirimiyle kalıcı bir iletinin yayınlanması başarısız olursa ve olumsuz yanıt istenmediyse, yayınlama işlemi yeniden denenir.

Olumsuz bir yanıt iletisi istediyseniz, olumsuz yanıt gönderilir ve başka bir yeniden deneme gerçekleştirilmez.

Zorlaseçeneğini belirlerseniz, kuyruğa alınan yayınlama/abone olma arabirimiyle kalıcı bir iletinin yayınlanması başarısız olursa, yayınlama işlemi başarıyla işleninceye kadar yeniden denenir. Olumsuz yanıt gönderilmez.

- `NonPersistentPublishRetry= yes | force`

Evetseçeneğini belirlerseniz, kuyruğa alınan yayınlama/abone olma arabirimiyle kalıcı olmayan bir iletinin yayınlanması başarısız olursa ve olumsuz yanıt istenmediyse, yayınlama işlemi yeniden denir.

Olumsuz bir yanıt iletisi istediyseniz, olumsuz yanıt gönderilir ve başka bir yeniden deneme gerçekleştirilmez.

Zorlaseçeneğini belirlediyseniz, kuyruğa alınan yayınlama/abone olma arabirimiyle kalıcı olmayan bir iletinin yayınlanması başarısız olursa, yayınlama işlemi başarıyla işleninceye kadar yeniden denir. Olumsuz yanıt gönderilmez.

Not: If you want to enable this functionality for non-persistent messages, then as well as setting the `NonPersistentPublishRetry` value you must also ensure that the queue manager attribute **PSSYNCPT** is set to Evet.

Bunu yapmak, kalıcı olmayan yayınların işlenmekte olan performansının, STREAM kuyruğundan **MQGET** olarak işlenmesinin, uyumluluk noktası altında ortaya çıkması gibi bir etkisi de olabilir.

- `PublishBatchBoyut =sayı`

Aracı olağan durumda, yayınlama iletilerini eşitleme noktası içinde işler. Her yayını tek tek kesinleştirmek için verimsiz olabilir ve bazı durumlarda, aracı tek bir iş biriminde birden çok yayınlama iletisini işleyebilir. Bu değıştirge, tek bir iş biriminde işlenebilecek yayınlama iletileri sayısı üst sınırını belirtir.

`PublishBatchBoyutu` için varsayılan değer 5 'tür.

- `PublishBatchAralık =sayı`

Aracı olağan durumda, yayınlama iletilerini eşitleme noktası içinde işler. Her yayını tek tek kesinleştirmek için verimsiz olabilir ve bazı durumlarda, aracı tek bir iş biriminde birden çok yayınlama iletisini işleyebilir. Bu parametre, bir toplu işteki ilk ileti ile aynı grupta yer alan sonraki yayınların arasında geçen süre üst sınırını (milisaniye cinsinden) belirtir.

0 toplu iş aralığı, iletilerin hemen kullanılabilir olması koşuluyla, `PublishBatchBoyutu` iletilerine kadar işlenebileceğini belirtir.

`PublishBatchInterval` için varsayılan değer sıfırdır.

Yordam

Yayınlama/abone olma davranışını denetleyen kuyruk yöneticisi özneliklerini değıştirmek için WebSphere MQ Explorer (Windows Gezgini), programlanabilir (programlanabilir) komutları ya da **runmqsc** komutunu kullanın.

Örnek

```
ALTER QMGR PSNPRES (GÜVENLİ)
```

Kuyruğa alınan yayınlama/abone olma başlatılıyor

Başlamadan önce

Yayınlama/abone olma üç kipinin anlaşılması için [PSMODE](#) açıklamasını okuyun:

- COMPAT
- DEVRE DIŐI
- Etkinleştirildi

Not: Yükseltilmiş bir kuyruk yöneticisiyle çalışıyorsanız, Version 6.0 ' tan yeni düzeye geçirdiyseniz, Version 6.0 yayınlama/abone olma aracısı durumunu geçirmek için **stirmqbrk** ögesini kullanmalısınız. Bu, z/OS için geçerli değildir.

Bu görev hakkında

QMGR PSMODE özneliğini, kuyruklanan yayınlama/abone olma arabirimini (aracı olarak da bilinir) ya da yayınlama/abone olma motoru (Sürüm 7 yayınlama/abone olma olarak da bilinir) ya da her ikisini birden başlatmaya ayarlayın. Kuyruğa alınmış yayınlama/abone olma işlemini başlatmak için PSMODE ögesini ENABETLEolarak ayarlamamız gerekir. Varsayılan değer ENABETLE' dir.

Yordam

Arabirim önceden etkinleştirilmediyse, kuyruğa alınan yayınlama/abone olma arabirimini etkinleştirmek için WebSphere MQ Explorer 'ı ya da **runmqsc** komutunu kullanın.

Örnek

```
ALTER QMGR PSMODE(ENABLED)
```

Sonraki adım

WebSphere MQ , kuyruğa alınan yayınlama/abone olma komutlarını ve yayınlama/abone olma İleti Kuyruğu Arabirimi (MQI) çağrılarını işler.

Kuyruğa alınan yayınlama/abone olma işlemi durduruluyor

Başlamadan önce

Kuyruğa alınan yayınlama/abone olma işlemi kullanımdan kaldırıldı.

Yayınlama/abone olma üç kipinin anlaşılması için [PSMODE](#) açıklamasını okuyun:

- ŞİRKET
- DEVRE DIŞI
- Etkinleştirildi

Bu görev hakkında

QMGR PSMODE özneliğini, kuyruğa alınmış yayınlama/abone olma arabirimini (aracı olarak da bilinir) ya da yayınlama/abone olma motoru (Sürüm 7 yayınlama/abone olma olarak da bilinir) ya da her ikisini durdurabilmek için ayarlayın. Kuyruğa alınmış yayınlama/abone olma işlemi durdurmak için PSMODE ' u ŞİRKETolarak ayarlamamız gerekir. Yayınlama/abone olma motorunu tamamen durdurmak için PSMODE seçeneğini DISABLEolarak ayarlayın.

Yordam

Kuyruğa alınan yayınlama/abone olma arabirimini geçersiz kılmak için WebSphere MQ Explorer 'ı ya da **runmqsc** komutunu kullanın.

Örnek

```
ALTER QMGR PSMODE(COMPAT)
```

Akış eklenmesi

Akışları, Version 6.0 kuyruk yöneticilerinden geçirilen akışlarla birlikte var olacak şekilde el ile ekleyebilirsiniz.

Başlamadan önce

[Akışlar ve konular](#) konusunu okuyarak yayınlama/abone olma akışlarının çalışma şeklini tanıyın.

Bu görev hakkında

Bu adımları gerçekleştirmek için PCF komutunu, **runmqsc**ya da IBM WebSphere MQ Explorer kullanın.

Not: Adım 1 ve 2 'yi istediğiniz sırada gerçekleştirebilirsiniz. Yalnızca adım 1 ve 2 tamamlandıktan sonra adım 3 'ü gerçekleştirin.

Yordam

1. Version 6.0 akışıyla aynı ada sahip bir yerel kuyruk tanımlayın.
2. Version 6.0 akışıyla aynı ada sahip bir yerel konu tanımlayın.
3. Kuyruğun adını ad listesine ekle, SYSTEM.QPUBSUB.QUEUE.NAMELIST
4. Yayınlama/abone olma sıradüzeninde olan Version 7.1 ya da daha yüksek düzeydeki tüm kuyruk yöneticileri için bu işlemi yineleyin.

ekleme 'Sport'

Akış paylaşma örneğinde 'Sport', Version 6.0 ve Version 7.1 kuyruk yöneticileri aynı yayınlama/abone olma sıradüzeninde çalışıyor. Version 6.0 kuyruk yöneticileri, 'Sport' adlı bir akışı paylaşır. Bu örnekte, sürüm 6 akışıyla paylaşılan 'Sport' konu dizgisiyle 'Sport' Version 7.1 'Sport' adlı kuyruk yöneticilerinde bir kuyruk ve konunun nasıl yaratılacağı gösterilmektedir.

'Soccer/Results' konu dizgisiyle 'Sport' konusuna yayınlayan bir Version 7.1 yayınlama uygulaması, 'Sport/Soccer/Results' sonuçtaki konu dizgisini oluşturur. Version 7.1 kuyruk yöneticilerindeki 'Sport' konusuna abone olan 'Soccer/Results' konu dizgisiyle birlikte yayını alır.

Version 6.0 kuyruk yöneticileriyle ilgili olarak, akışa abone olan 'Sport', 'Soccer/Results' konu dizgisiyle birlikte yayını alır.

```
runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
define qlocal('Sport')
  1 : define qlocal('Sport')
AMQ8006: WebSphere MQ queue created.
define topic('Sport') topicstr('Sport')
  2 : define topic('Sport') topicstr('Sport')
AMQ8690: WebSphere MQ topic created.
alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
  3 : alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport',
'SYSTEM.BROKER.DEFAULT.STREAM', 'SYSTEM.BROKER.ADMIN.STREAM')
AMQ8551: WebSphere MQ namelist changed.
```

Not: **alter namelist** komutuna eklediğiniz yeni adların yanı sıra namelist nesnesinde var olan adları da sağlamanız gerekir.

Sonraki adım

Akışla ilgili bilgiler, sıradüzendeki diğer araçlara iletilir.

Bir aracı Version 6.0 ise, bunu Version 6.0 aracı olarak denetleyin. Başka bir deyişle, akış kuyruğunu el ile yaratma ya da gerektiğinde aracının akış kuyruğunu dinamik olarak yaratmasına izin verme seçeneğiniz vardır. Kuyruk, model kuyruğu tanımlamasına (SYSTEM.BROKER.MODEL.STREAM) dayalıdır.

Bir aracı Version 7.1 ise, sıradüzendeki her bir Version 7.1 kuyruk yöneticisini el ile yapılandırmanız gerekir.

Akış silinmesi

Bir akışı IBM WebSphere MQ Version 7.1 ya da daha sonraki bir yayın düzeyinden silebilirsiniz.

Başlamadan önce

Kuyruğa alınmış yayınlama/abone olma kullanımı IBM WebSphere MQ Version 7.1 içinde kullanımdan kaldırılmıştır.

Bir akışı silmeden önce akışa kalan abonelik olmadığından emin olmanız ve akışı kullanan tüm uygulamaları susturmanız gerekir. Yayınlar silinen bir akışa akmaya devam ederse, sistemi temiz bir çalışma durumuna geri döndürmek için çok fazla yönetim çalışması gerekir.

Bu görev hakkında

Akışın bağlı olduğu herhangi bir Version 6.0 kuyruk yöneticisinden silinmesine ilişkin yönergeler için bkz. [Akışın silinmesi \(v6.0 belgelerindeps11870_.htm\)](#).

Yordam

1. Bu akışı barındıran tüm bağlı araçları bulun.
2. Tüm araçlarda akışa olan tüm abonelikleri iptal edin.
3. Kuyruğu (akımla aynı ada sahip) ad en alt sırasından (SYSTEM.QPUBSUB.QUEUE.NAMELIST) kaldırın.
4. Akışla aynı ada sahip tüm iletileri kuyruktan silin ya da silin.
5. Akımla aynı ada sahip kuyruğu silin.
6. İlişkili konu nesnesini silin.

Sonraki adım

1. Akışı barındıran diğer tüm Version 7.1 ya da daha sonraki kuyruk yöneticileri için 3-5 arasındaki adımları yineleyin.
2. Akışı diğer tüm bağlı Version 6.0 ya da daha önceki kuyruk yöneticilerinden kaldırın.

Abonelik noktası ekleme

How to add a subscription point that was not migrated from IBM WebSphere MQ Event Broker or IBM WebSphere MQ Message Broker by **migmbbrk**. IBM WebSphere MQ Event Broker ya da IBM WebSphere MQ Message Broker 'dan yeni bir abonelik noktasıyla geçirdiğiniz, varolan bir kuyruğa alınmış yayınlama/abone olma uygulamasını genişletin.

Başlamadan önce

1. IBM WebSphere MQ Event Broker 'dan ve IBM WebSphere MQ Message Broker Version 6.0 olanağında IBM WebSphere MQ Version 7.1 ürününe geçiş işlemi tamamlayın.
2. Abonelik noktasının SYSTEM.QPUBSUB.SUBPOINT.NAMELIST içinde önceden tanımlı olmadığından emin olun.
3. Abonelik noktasıyla aynı adı taşıyan bir konu nesnesi mi, yoksa bir konu dizgisi olup olmadığını denetleyin.

Bu görev hakkında

Var olan IBM WebSphere MQ Event Broker uygulamaları abonelik noktalarını kullanır. Yeni IBM WebSphere MQ Version 7.1 uygulamaları, abonelik noktalarını kullanmaz, ancak abonelik noktası geçiş mekanizmasını kullanarak, var olan uygulamalarla birlikte çalışabilir.

Bir abonelik noktası, geçiş sırasında abonelik noktası kullanılmıyorsa, **migmbbrk** tarafından geçirilmemiş olabilir.

IBM WebSphere MQ Event Broker 'dan geçirilen var olan kuyruğa alınmış yayınlama/abone olma programlarına bir abonelik noktası eklemek isteyebilirsiniz.

Abonelik noktaları, IBM WebSphere MQ Version 6.0 programından ya da daha önceki bir sürümden yeni düzeye geçirilen MQRFH1 üstbilgilerini kullanan kuyruğa alınmış yayınlama/abone olma programları ile çalışmaz.

There is no need to add subscription points to use integrated publish/subscribe applications written for IBM WebSphere MQ Version 7.1.

Yordam

1. Abonelik noktasının adını SYSTEM.QPUBSUB.SUBPOINT.NAMELIST olarak ekleyin.
 - z/OS üzerinde **NLTYPE** , varsayılan değer olan NONE(HİÇBİRİ) olarak ayarlıdır.
 - Aynı yayınlama/abone olma topolojisinde bağlı olan her kuyruk yöneticisinde adımı yineleyin.
2. Bir konu nesnesi ekleyin. Tercihen, abonelik noktasının adıyla eşleşen bir konu dizgisiyle, abonelik noktasının adını veren bir nesne ekleyin.
 - Abonelik noktası bir kümese, konu nesnesini küme konusu ana makinesinde küme konusu olarak ekleyin.
 - Bir konu nesnesi, abonelik noktasının adıyla aynı konu dizgisiyle varsa, var olan konu nesnesini kullanın. Var olan bir konuyu yeniden kullanarak abonelik noktasının sonuçlarını anlamamız gerekir. Var olan konu var olan bir uygulamanın bir parçasıysa, aynı adı taşıyan iki konu arasındaki çakışmayı çözümleniz gerekir.
 - Bir konu nesnesi, abonelik noktasıyla aynı adı taşıyan, ancak farklı bir konu dizgisiyle varsa, farklı bir adla bir konu oluşturun.
3. **Topic** özniteliğini JOKER değerini BLOCK değerine ayarlayın.

Abonelikleri # ya da * joker aboneliklerini abonelik noktalarına yalıtmayı engelle, bkz. [Vahşi kartlar ve abonelik noktaları](#).
4. Konu nesnesinde gerek duyduğunuz öznitelikleri ayarlayın.

Örnek

Örnekte, iki abonelik noktası (USD ve GBP) ekleyen bir **runmqsc** komut dosyası gösterilmektedir.

```
DEFINE TOPIC(USD) TOPICSTR(USD)
DEFINE TOPIC(GBP) TOPICSTR(GBP) WILDCARD(BLOCK)
ALTER NL(SYSTEM.QPUBSUB.SUBPOINT.NAMELIST) NAMES(SYSTEM.BROKER.DEFAULT.SUBPOINT, USD, GBP)
```

Not:

1. Varsayılan abonelik noktasını, **ALTER** komutu kullanılarak eklenen abonelik noktaları listesine ekleyin. **ALTER** , ad listesinde var olan adları siler.
2. Ad listesini değiştirmeden önce konuları tanımlayın. Kuyruk yöneticisi yalnızca, kuyruk yöneticisi başladığında ve ad listesi değiştirildiğinde ad listesini denetler.

Kuyruk yöneticisini aracı sıradüzenine bağla

Bir aracı sıradüzenini değiştirmek için yerel bir kuyruk yöneticisini üst kuyruk yöneticisine bağlayabilirsiniz.

Başlamadan önce

1. Kuyruğa alınmış yayınlama/abone olma kipini etkinleştirin. Bkz. [Kuyruktaki yayınlama/abone olma işlemini başlatma](#).
2. Bu değişiklik, bir IBM WebSphere MQ bağlantısı kullanılarak üst kuyruk yöneticisine yayılır. Bağlantıyı kurmanın iki yolu vardır.
 - Kuyruk yöneticilerinin bir IBM WebSphere MQ kümesine bağlanması için [Kümeyle kuyruk yöneticisi eklenmesi](#) başlıklı konuya bakın.

- Üst kuyruk yöneticisiyle aynı adı taşıyan bir iletim kuyruğunu ya da kuyruk yöneticisi diğer adını kullanarak noktadan noktaya iletişim kanalı bağlantısı kurun. Noktadan noktaya iletişim kanal bağlantısı kurulmasına ilişkin ek bilgi için [WebSphere MQ distributed-messaging \(dağıtılmış ileti sistemi\) teknikleri](#) başlıklı konuya bakın.

Bu görev hakkında

Alt öğeleri üst öğelere bağlamak için ALTER QMGR PARENT (*PARENT_NAME*) runmqsc komutunu kullanın.

Dağıtılmış yayınlama/abone olma, kuyruk yöneticisi kümeleri ve kümelenebilir konu tanımlamaları kullanılarak gerçekleştirilir. IBM WebSphere MQ Version 6.0 ve WebSphere Message Broker Version 6.1 ve WebSphere Event Broker Version 6.1 ve önceki sürümleriyle birlikte çalışabilirlik için, kuyruğa alınmış yayınlama/abone olma kipi etkinleştirildiği sürece Version 7.1 ya da sonraki kuyruk yöneticilerini bir aracı sıradüzenine bağlayabilirsiniz.

Yordam

QMGR ÜST ÖĞESİNİ DEĞİŞTİR (ÜST)

Örnek

İlk örnekte, QM2 'nin QM1alt öğesi olarak nasıl bağlanacağı ve daha sonra, bağlantısı için QM2 ' nin nasıl sorgulanacağı gösterilmektedir:

```
C:>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2
alter qmgr parent(QM1)
  1 : alter qmgr parent(QM1)
AMQ8005: WebSphere MQ queue manager changed.
display pubsub all
  2 : display pubsub all
AMQ8723: Display pub/sub status details.
      QMNAME(QM2)                TYPE(LOCAL)
      STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
      QMNAME(QM1)                TYPE(PARENT)
      STATUS(ACTIVE)
```

Sonraki örnekte, bağlantıları için QM1 sorgulanırken elde edilen sonuç gösterilmektedir:

```
C:\Documents and Settings\Admin>runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
display pubsub all
  2 : display pubsub all
AMQ8723: Display pub/sub status details.
      QMNAME(QM1)                TYPE(LOCAL)
      STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
      QMNAME(QM2)                TYPE(CHILD)
      STATUS(ACTIVE)
```

Sonraki adım

Bir aracıya ya da kuyruk yöneticisinde, bağlı kuyruk yöneticilerindeki yayıncıların ve abonelerin kullanabileceği konular tanımlayabilirsiniz. Daha fazla bilgi için [Yönetim konusunu tanımlama](#) başlıklı konuya bakın.

İlgili kavramlar

[Akışlar ve konular](#)

[WebSphere MQ yayınlama/abone olma ileti sistemine giriş](#)

İlgili başvurular

[PUBSUB 'I GÖRÜNTÜLE](#)

Bir kuyruk yöneticisinin aracı sıradüzeninden bağlantısını kes

Aracı sıradüzenindeki bir üst kuyruk yöneticisiyle alt kuyruk yöneticisinin bağlantısını kesin.

Bu görev hakkında

Bir kuyruk yöneticisinin aracı sıradüzeniyle bağlantısını kesmek için **ALTER QMGR** komutunu kullanın. İsteddiğiniz zaman bir kuyruk yöneticisinin bağlantısını kesebilirsiniz.

Kuyruk yöneticileri arasındaki bağlantı çalışırken üst öğeyi güncelleme isteği gönderilir.

Yordam

```
ALTER QMGR PARENT('')
```

Örnek

```
C:\Documents and Settings\Admin>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2.
  1 : alter qmgr parent('')
AMQ8005: WebSphere MQ queue manager changed.
  2 : display pubsub type(child)
AMQ8147: WebSphere MQ object not found.
display pubsub type(parent)
  3 : display pubsub type(parent)
AMQ8147: WebSphere MQ object not found.
```

Sonraki adım

Artık gerekli olmayan akışları, kuyrukları ve el ile tanımlanan kanalları silebilirsiniz.

Kuyruk yöneticisi kümesinin yapılandırılması

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

Başlamadan önce

Kümeleme kavramlarına giriş için aşağıdaki konulara bakın:

- [Kümeleme nasıl çalışır](#)
- [“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158](#)
- [“Bir kümenin bileşenleri” sayfa 161](#)

Kuyruk yöneticisi kümenizi tasarlarken bazı kararlar almanız gerekir. Önce kümedeki hangi kuyruk yöneticilerinin küme bilgilerinin tam havuzlarını tutacağına karar vermeniz gerekir. Yarattığınız herhangi bir kuyruk yöneticisi bir kümede çalışabilir. Bu amaçla istediğiniz sayıda kuyruk yöneticisi seçebilirsiniz, ancak ideal sayı ikidir. Tam havuzları tutmak üzere kuyruk yöneticilerini seçme hakkında bilgi için bkz. [“Tüm havuzları tutmak için küme kuyruğu yöneticilerinin nasıl seçileceği” sayfa 173.](#)

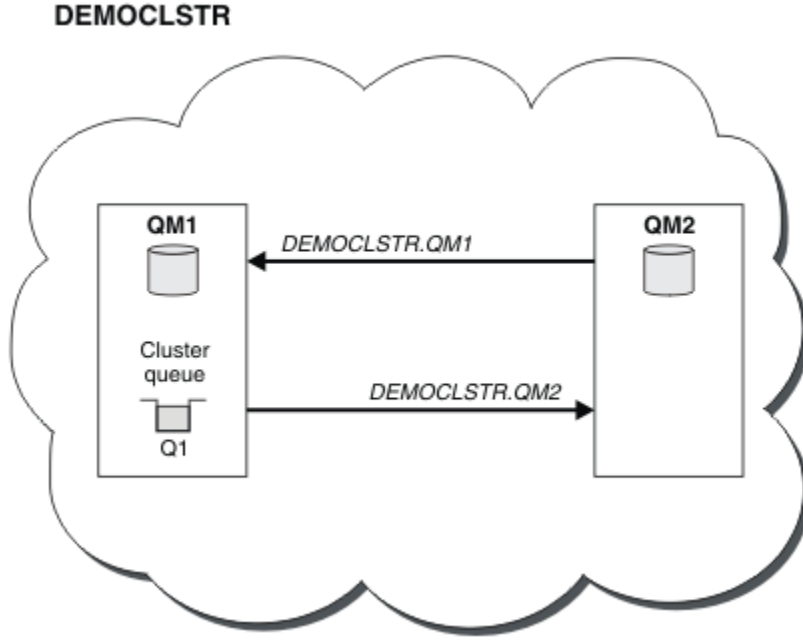
Kümenizi tasarlamaya ilişkin ek bilgi için aşağıdaki konulara bakın:

- [“Küme düzenleme” sayfa 175](#)
- [“Küme adlandırma kuralları” sayfa 175](#)
- [“Çakışan kümeler” sayfa 176](#)

Örnek

Olası en küçük küme yalnızca iki kuyruk yöneticisi içerir. Bu durumda, her iki kuyruk yöneticisi de tam havuz içerir. Kümeyi ayarlamak için yalnızca birkaç tanımlama gerekir, ancak her kuyruk yöneticisinde yüksek düzeyde bir özerklik vardır.

Şekil 21 sayfa 157 , QM1 ve QM2adlı iki kuyruk yöneticisine sahip DEMOCLSTR adlı bir kümeyi gösterir.



Şekil 21. İki kuyruk yöneticisinden oluşan küçük bir küme

- Kuyruk yöneticilerinin uzun adları vardır; örneğin, LONDON ve NEWYORK. Gelişmiş ve iş yükü dengeleme görevlerinde aynı adlar kullanılır. IBM WebSphere MQ for z/OS üzerinde, kuyruk yöneticisi adları dört karakterle sınırlıdır.
- Kuyruk yöneticilerinin adları, her kuyruk yöneticisinin ayrı bir makinede olduğu anlamına gelir. Bu görevleri, aynı makinedeki tüm kuyruk yöneticileriyle gerçekleştirebilirsiniz.
- Görevler, sistem yöneticisi tarafından **MQSC** komutları kullanılarak girileceği şekilde IBM WebSphere MQ komut dosyası komutlarını kullanır. Daha kolay IBM WebSphere MQ Explorer 'ı kullanmak da dahil olmak üzere, komutları girmenin başka yolları da vardır. WebSphere MQ komut dosyası komutlarının kullanılmasının amacı, görevlerde hangi IBM WebSphere MQ komutlarının kullanıldığını göstermektir.

Benzer bir örnek küme ayarlamaya ilişkin yönergeler için bkz. [“Yeni bir küme ayarlanıyor” sayfa 181.](#)

Sonraki adım

Kümeleri yapılandırma ve kümelerle çalışma hakkında daha fazla bilgi için aşağıdaki konulara bakın:

- [“Bir kümede iletişim kurulması” sayfa 179](#)
- [“IBM WebSphere MQ kümelerini yönetme” sayfa 181](#)
- [“İletileri kümeden ve kümeden yönlendirme” sayfa 244](#)
- [“İş yükü yönetimi için kümeleri kullanma” sayfa 258](#)

Kümenizi yapılandırmanıza yardımcı olacak daha fazla bilgi için bkz. [“Kümeleme ipuçları” sayfa 177.](#)

İlgili kavramlar

[Kümeler](#)

Erişim denetimi ve birden çok küme iletim kuyruğu

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, `SYSTEM.CLUSTER.TRANSMIT.QUEUE` 'a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanıtlara karşı denetlenir.

IBM WebSphere MQ , yerel olarak ya da yerel olarak ya da yerel olarak, bir kullanıcının uzak bir kuyruğa ileti koyma iznine sahip olup olmadığını denetleme seçeneğini sunar. Tipik bir IBM WebSphere MQ

uygulamasını yalnızca yerel denetimlerini kullanır ve uzak kuyruk yöneticisine dayalı olarak, yerel kuyruk yöneticisinde yapılan erişim denetimlerini güvenerek kullanır. Uzak denetim kullanılmıyorsa, ileti, uzak ileti kanalı işleminin yetkisiyle hedef kuyruğa konabilir. Uzaktan denetimi kullanmak için giriş kanalının koyma yetkisini bağlam güvenliğine ayarlamalısınız.

Yerel denetimler, uygulamanın açıldığını kuyruğa alır. Dağıtılmış kuyruklarda, uygulama genellikle uzak bir kuyruk tanımlaması açar ve uzak kuyruk tanımlamasına karşı erişim denetimlerine erişir. İleti tam bir yöneltme üstbilgisiyle konursa, çekler iletim kuyruğunda yapılır. Bir uygulama yerel kuyruk yöneticisinde olmayan bir küme kuyruğunu açarsa, denetilecek bir yerel nesne yoktur. Erişim denetimi denetimleri, küme iletim kuyruğuna (SYSTEM . CLUSTER . TRANSMIT . QUEUE) ilişkin olarak yapılır. Birden çok küme iletim kuyrukları olsa da, Version 7.5olanağından, uzak küme kuyrukları için yerel erişim denetimi denetimlerinin SYSTEM . CLUSTER . TRANSMIT . QUEUE' a karşı yapıldığını denetler.

Yerel ya da uzaktan denetime ilişkin seçenek, iki uç nokta arasında bir seçimdir. Uzaktan kontrol etmek çok ince bir işarettir. Her kullanıcının herhangi bir küme kuyruğuna konması için, kümedeki her kuyruk yöneticisinde bir erişim denetimi tanıtımı olmalıdır. Yerel olarak kontrol etmek kaba bir şekilde işarettir. Her kullanıcının, bağlı oldukları kuyruk yöneticilikteki küme iletim kuyruğu için tek bir erişim denetimi tanıtımı gerekir. Bu profille, herhangi bir kümedeki herhangi bir kuyruk yöneticisinde herhangi bir küme kuyruğuna ileti yerleştirebilir.

Version 7.1' ten bu yana, yöneticilerin küme kuyrukları için erişim denetimi ayarlamaya yönelik başka bir yolu vardır. **setmqaut** komutunu kullanarak kümedeki herhangi bir kuyruk yöneticisinde küme kuyruğu için bir güvenlik profili yaratabilirsiniz. Uzak bir küme kuyruğunu yerel olarak açmanızı, yalnızca kuyruk adını belirterek, tanıtıma etki eder. Ayrıca, uzak kuyruk yöneticisi için bir tanıtım da ayarlayabilirsiniz. Bunu yapmazsanız, kuyruk yöneticisi tam olarak nitelenmiş bir ad sağlayarak, küme kuyruğunu açan kullanıcının tanıtımını denetleyebilir.

Yeni tanıtımlar yalnızca, kuyruk yöneticisi stanza 'yı (**ClusterQueueAccessControl** - RQMAd1) değiştirdiğinizde çalışır. Varsayılan değer Xm1q' dur. Varolan tüm küme kuyrukları için tanıtımlar yaratmalısınız, varolan uygulamalar küme kuyruklarını kullanır. Profilleri oluşturmadan stanza 'yı RQMName ' e çevirdiğinizde, uygulamaların başarısız olma olasılığı yüksektir.

İpucu: Version 7.1 içindeki geri verme denetiminde bulunan küme kuyruğunda yapılan değişiklikler uzak kuyruklama için geçerli değildir. Erişim denetimleri hala yerel tanımlara göre yapılır. Bu değişiklikler, küme kuyruklarında ve küme konularındaki erişim denetimini yapılandırmak için aynı yaklaşımı izleyebileceğiniz anlamına gelir.

İlgili kavramlar

“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

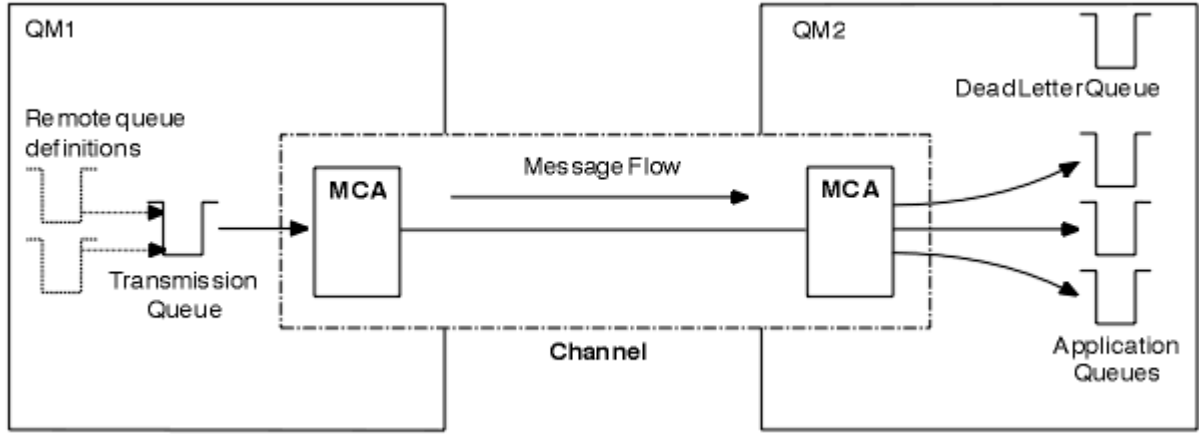
Kümeleme ve dağıtılmış kuyruklama karşılaştırması

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

Kümeleme kullanmayarsanız, kuyruk yöneticilerinizin dağıtılmış kuyruğa alma ile bağımsız ve bağımsız bir şekilde iletişim kurmanızı sağlar. Bir kuyruk yöneticisinin başka bir kuyruk yöneticisine ileti göndermesi gerekiyorsa, şunları tanımlamanız gerekir:

- İletim kuyruğu
- Uzak kuyruk yöneticisine bir kanal

Şekil 22 sayfa 159 , dağıtılmış kuyruklama için gereken bileşenleri gösterir.



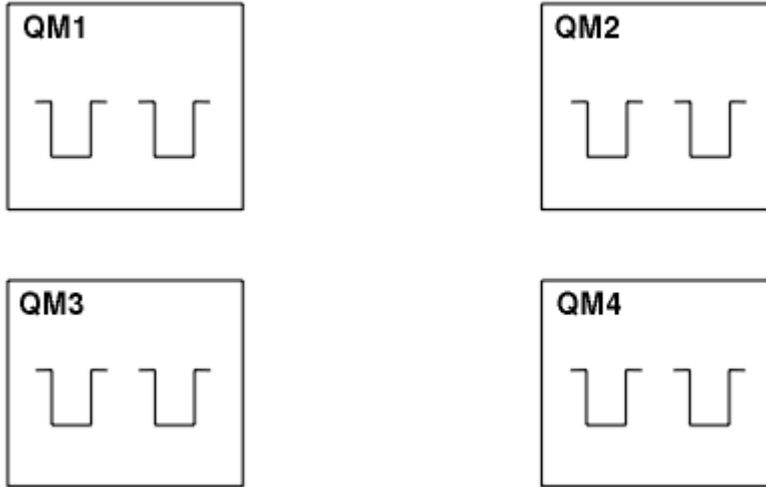
Şekil 22. Dağıtılmış kuyruklama

Bir kümedeki kuyruk yöneticilerini gruplarsanız, herhangi bir kuyruk yöneticisinde bulunan kuyruklar, kümedeki başka bir kuyruk yöneticisinde kullanılabilir. Herhangi bir kuyruk yöneticisi, belirtilik tanımlamalarla aynı kümedeki başka bir kuyruk yöneticisine ileti gönderebilir. Her hedef için kanal tanımları, uzak kuyruk tanımları ya da iletim kuyrukları sağlamıyor musunuz? Bir kümedeki her kuyruk yöneticisinin, iletileri kümedeki diğer bir kuyruk yöneticisine iletebileceği tek bir iletim kuyruğu vardır. Bir küme içindeki her kuyruk yöneticisinin yalnızca tanımlanması gerekir:

- İletilerin alınacağı bir küme-alıcı kanalı
- Kendi kendisini tanıtan ve kümeyi öğrenen bir küme-gönderici kanalı

Bir küme ve dağıtılmış kuyruklama ayarına ilişkin tanımlar

Her biri iki kuyrukla dört kuyruk yöneticisini gösteren Şekil 23 sayfa 159' a bakın. Dağıtılmış kuyruklama özelliğini kullanarak bu kuyruk yöneticilerini bağlamak için kaç tanım gerektiğini göz önünde bulundurun. Bir kümeyle aynı ağı kurmak için gereken tanımların sayısını karşılaştırın.



Şekil 23. Dört kuyruk yöneticilerinden oluşan bir ağ

Dağıtılmış kuyruklama kullanarak bir ağ oluşturmak için tanımlar

Dağıtılmış kuyruklama kullanarak Şekil 22 sayfa 159 içinde gösterilen ağı ayarlamak için aşağıdaki tanımlara sahip olabilirsiniz:

<i>Çizelge 24. Dağıtılmış kuyruğa alma tanımlarına ilişkin tanımlar</i>		
Tanım	Kuyruk yöneticisi başına sayı	Toplam sayı
Diğer tüm kuyruk yöneticilerine iletilerin gönderileceği bir kanala ilişkin gönderici kanal tanımlaması	3	12
Diğer tüm kuyruk yöneticilerinden iletilerin alınacağı bir kanala ilişkin bir günlük nesnesi kanal tanımlaması	3	12
Diğer kuyruk yöneticiliklerine iletim kuyruğu için bir iletim kuyruğu tanımlaması	3	12
Her yerel kuyruk için yerel kuyruk tanımlaması	2	8
Bu kuyruk yöneticisinin iletileri koymak istediği her uzak kuyruk için bir uzak kuyruk tanımlaması	6	24

Soysal alıcı kanal tanımlamalarını kullanarak bu tanımlama sayısını azaltabilir. Tanımlama sayısı üst sınırı, bu ağ için toplam 68 olan her bir kuyruk yöneticisinde en çok 17 olabilir.

Kümeleri kullanarak bir ağ oluşturmak için tanımlar

Şekil 22 sayfa 159 ' ta gösterilen ağ, kümeleri kullanarak ayarlamak için aşağıdaki tanımlara gerek duyarsınız:

<i>Çizelge 25. Kümelemeye ilişkin tanımlar</i>		
Tanım	Kuyruk yöneticisi başına sayı	Toplam sayı
Bir havuz kuyruğu yöneticisine ileti göndermek için kullanılan bir kanala ilişkin küme gönderen kanal tanımlaması	1	4
Kümedeki diğer kuyruk yöneticilerinden iletilerin alınacağı kanala ilişkin bir küme alıcı kanal tanımlaması	1	4
Her yerel kuyruk için yerel kuyruk tanımlaması	2	8

Bu kuyruk yöneticisi kümesini (iki tam havuzla) ayarlamak için, her kuyruk yöneticisi için toplam on altı tanım olmak üzere dört tanımlama gerekir. Ayrıca, kuyruk yöneticisi tanımlamalarını, küme için tam havuz kuyruk yöneticileri yapmak üzere iki kuyruk yöneticisine ilişkin kuyruk yöneticisi tanımlamalarını değiştirmeniz de gerekir.

Yalnızca bir CLUSTSDR ve bir CLUSTRVR kanal tanımlaması gerekli. Küme tanımlandığında, diğer kuyruk yöneticilerine herhangi bir kesinti olmadan kuyruk yöneticisi (havuz kuyruğu yöneticileri dışında) ekleyebilir ya da var olan bir kuyruk yöneticisini kaldırabilirsiniz.

Bir küme kullanılması, birçok kuyruk yöneticisi içeren bir ağ oluşturmak için gereken tanımların sayısını azaltır.

Daha az tanımın olması, hata riskinin daha az olduğunu ortaya çıkartır:

- Nesne adları her zaman eşleşir; örneğin, bir gönderen alıcı çiftindeki kanal adı.
- Bir kanal tanımlamasında belirtilen iletim kuyruğu adı, her zaman doğru iletim kuyruğu tanımlamasıyla ya da uzak bir kuyruk tanımlamasında belirlenen iletim kuyruğu adıyla eşleşir.
- Bir QREMOTE tanımlaması her zaman uzak kuyruk yöneticisinde doğru kuyruğu işaret eder.

Bir küme ayarlandıktan sonra, başka bir kuyruk yöneticisinden herhangi bir sistem yönetimi çalışması yapmak zorunda kalmadan, küme kuyruklarını küme içinde bir kuyruk yöneticisinden diğerine

taşıyabilirsiniz. Kanal, uzak kuyruk ya da iletim kuyruğu tanımlarını silmeyi ya da değiştirmeyi unutmanın bir şansı yoktur. Var olan ağda herhangi bir kesinti olmadan, bir kümeye yeni kuyruk yöneticileri ekleyebilirsiniz.

Bir kümenin bileşenleri

Kümeler, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

Küme bileşenlerinin her biriyle ilgili bilgi için aşağıdaki alt başlıklara bakın:

İlgili kavramlar

Kümeler

[“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158](#)

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

[“IBM WebSphere MQ kümelerini yönetme” sayfa 181](#)

IBM WebSphere MQ kümelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

İlgili görevler

[“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156](#)

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

[“Yeni bir küme ayarlanıyor” sayfa 181](#)

Örnek kümeyi ayarlamak için bu yönergeleri izleyin. Ayrı yönergeler, TCP/IP, LU 6.2 üzerinde ve tek bir iletim kuyruğu ya da birden çok iletim kuyruğundaki kümeyi ayarlayıp tanımlamalarını açıklar. Bir kuyruk yöneticisinden diğerine ileti göndererek, kümenin çalıştığını test edin.

Küme havuzu

Havuz, bir kümenin üyesi olan kuyruk yöneticilerine ilişkin bilgi toplamlıdır.

Havuz bilgileri, kuyruk yöneticisi adlarını, yerlerini, anasistemleri ve diğer bilgileri kuyrukları kuyruklarını içeren yerleri içerir. Bilgiler, SYSTEM. CLUSTER. REPOSITORY. QUEUE adlı bir kuyruksa bulunan iletiler biçiminde saklanır. Kuyruk, varsayılan nesnelere biridir. z/OS için WebSphere MQ dışında bir WebSphere MQ kuyruk yöneticisi yarattığınızda bu tanımdır.

Tipik olarak, bir kümedeki iki kuyruk yöneticisi tam havuzu tutar. Kalan kuyruk yöneticileri bir kısmı havuzu tutar.

Tüm havuz ve kısmi havuz

Kümedeki her kuyruk yöneticisine ilişkin eksiksiz bir bilgi kümesini barındıran bir kuyruk yöneticisi tam bir havuza sahiptir. Kümedeki diğer kuyruk yöneticileri, tam havuzlardaki bilgilerin bir alt kümesini içeren kısmi havuzlara sahiptir.

Kısmi havuz, yalnızca kuyruk yöneticisinin ileti alışverişi için gereken kuyruk yöneticilerine ilişkin bilgileri içerir. Kuyruk yöneticileri, güncellemeleri gereken bilgilere göre ister; bu nedenle, değişiklik olursa, tüm havuz kuyruk yöneticisi yeni bilgileri gönderir. Çoğu zaman, kısmi bir havuz, bir kuyruk yöneticisinin küme içinde gerçekleştirmesi gereken tüm bilgileri içerir. Bir kuyruk yöneticisinin bazı ek bilgilere gereksinim duyması, tam havuzun sorgularını yapar ve kısmi havuzunu günceller. Kuyruk yöneticileri, havuzlara ilişkin güncellemeleri istemek ve bu güncellemeleri almak için SYSTEM. CLUSTER. COMMAND. QUEUE adlı bir kuyruk kullanır. This queue is one of the default objects.

Küme kuyruk yöneticisi

Küme kuyruk yöneticisi, küme üyesi olan bir kuyruk yöneticidir.

Kuyruk yöneticisi birden çok kümeden bir üye olabilir. Her küme kuyruk yöneticisinin, üyesi olduğu tüm kümelerde benzersiz bir adı olmalıdır.

Küme kuyruk yöneticisi, kümedeki diğer kuyruk yöneticilerine duyurusu yapan kuyrukları barınebilir. Bir küme kuyruk yöneticisinin herhangi bir kuyrukları barınması ya da herhangi bir kuyruğa yayınlaması gerekmez. Bu, kümeye iletileri besleyebilir ve yalnızca belirtik olarak yönlendirilen yanıtları alabilir ve yayınlanan kuyruklara yanıt verebilir.

In WebSphere MQ for z/OS, a cluster queue manager can be a member of a queue-sharing group. Bu durumda, kuyruk tanımlamalarını aynı kuyruk paylaşım grubundaki diğer kuyruk yöneticileriyle paylaşır.

Küme kuyruğu yöneticileri otonom. Onlar, tanımladıkları kuyruklar ve kanallar üzerinde tam kontrole sahiptir. Tanımları, diğer kuyruk yöneticileri tarafından (aynı kuyruk paylaşım grubundaki kuyruk yöneticileri dışında) değiştiremezler. Havuz kuyruğu yöneticileri, kümedeki diğer kuyruk yöneticilerindeki tanımları denetlemez. Bunlar, gerektiğinde kullanılmak üzere tüm tanımların eksiksiz bir kümesini tutar. Küme, kuyruk yöneticilerinden oluşan bir birleşiktir.

Bir küme kuyruk yöneticisinde bir tanımlama yarattıktan ya da değiştirdikten sonra, bilgiler tam havuz kuyruk yöneticisine gönderilir. Kümedeki diğer havuzlar daha sonra güncelleştirilir.

Tam Havuz Kuyruğu Yöneticisi

Tam havuz kuyruk yöneticisi, kümenin kaynaklarının tam gösterimini tutan bir küme kuyruk yöneticidir. Kullanılabilirliği sağlamak için, her kümede iki ya da daha çok tam havuz kuyruğu yöneticisi ayarlayın. Tam havuz kuyruğu yöneticileri, kümedeki diğer kuyruk yöneticileri tarafından gönderilen bilgileri alır ve havuzlarını günceller. Her ikisinin de küme hakkında yeni bilgiler ile güncel tutulduğundan emin olmak için birbirlerine mesaj gönderirler.

Kuyruk yöneticileri ve havuzlar

Her kümede en az bir (tercihen iki) kuyruk yöneticisi, bir kümedeki kuyruk yöneticilerine, kuyruklara ve kanallara ilişkin bilgileri tam olarak bulunduran bir yöneticiye sahiptir. Bu havuzlar, bilgi güncellemeleri için kümedeki diğer kuyruk yöneticilerinden gelen istekleri de içerir.

Diğer kuyruk yöneticilerinin her biri, iletişim kurmaları gereken kuyruk ve kuyruk yöneticilerinin altkütmesine ilişkin bilgileri içeren kısmi bir havuz tutmasını sağlar. Kuyruk yöneticileri, başka bir kuyruğa ya da kuyruk yöneticisine ilk erişmeleri gerektiğinde sorgular yaparak, kısmi havuzlarını oluşturur. Kuyruk ya da kuyruk yöneticisiyle ilgili herhangi bir yeni bilgi bildirileceğini bildirirler.

Each queue manager stores its repository information in messages on a queue called SYSTEM.CLUSTER.REPOSITORY.QUEUE. Kuyruk yöneticileri, SYSTEM.CLUSTER.COMMAND.QUEUE adlı bir kuyruksa bulunan iletilerde havuz bilgilerini değiş tokuş eder.

Bir kümeye katılan her kuyruk yöneticisi, havuzlardan birine bir küme gönderen CLUSSDR(CLUSSDR), kanal tanımlar. Kümedeki diğer kuyruk yöneticilerinin tam olarak hangi havuzlarda tutulmayı öğrendiğini hemen öğrenir. Bundan sonra kuyruk yöneticisi, havuzlardan herhangi birinden bilgi isteyebilirler. Kuyruk yöneticisi, seçilen havuza bilgi gönderdiğinde, bilgileri başka bir havuza (varsa) gönderir.

Tam bir havuz, kuyruk yöneticisi tarafından barındırıldığında, buna bağlı kuyruk yöneticilerinden birinden yeni bilgiler aldığıda güncelleştirilir. Yeni bilgiler, bir havuz kuyruğu yöneticisinin hizmet dışı olması durumunda gecikme riskini azaltmak için başka bir havuza de gönderilir. Tüm bilgiler iki kez gönderildiğinden, havuzların yinelemeleri atması gerekir. Her bilgi ögesi, havuzların yinelemeleri tanımlamak için kullandığı bir sıra numarası taşır. Tüm havuzlar, ileti alışverişi yaparak adımlarda birbirlerine adım atmakta tutulur.

Küme kuyrukları

Küme kuyruğu, küme kuyruk yöneticisinin barındırdığı ve kümedeki diğer kuyruk yöneticilerine sunulan bir kuyruktır.

Bir küme kuyruğunu, kuyruğun barındırıldığı küme kuyruk yöneticisinde yerel kuyruk olarak tanımlayın. Kuyruğun ait olduğu kümenin adını belirtin. Aşağıdaki örnekte, CLUSTER seçeneğiyle bir küme kuyruğu tanımlamak için bir **runmqsc** komutu gösterilmektedir:

```
DEFINE QLOCAL(Q1) CLUSTER(SALES)
```

Küme kuyruğu tanımlaması, kümedeki diğer kuyruk yöneticilerine duyurulmaktadır. Kümedeki diğer kuyruk yöneticileri, karşılık gelen bir uzak kuyruk tanımlamasına gerek duymadan, iletileri bir küme kuyruğuna yerleştirebilir. Küme kuyruğu, bir küme adı listesi kullanılarak birden çok kümede tanıtılabilir.

Bir kuyruk duyurulduğunda, kümedeki herhangi bir kuyruk yöneticisi iletiye ileti yerleştirebilir. Bir iletiyi koymak için, kuyruk yöneticisinin kuyruğun barındırıldığı tüm havuzlardan öğrenmesi gerekir. Daha sonra, iletiye bazı yöneltme bilgileri ekler ve iletiyi bir küme iletim kuyruğuna yerleştirir.

Küme kuyruğu, IBM WebSphere MQ for z/OS' ta bir kuyruk paylaşım grubunun üyeleri tarafından paylaşılan bir kuyruk olabilir.

Bağ Tanımı

Birden çok kuyruk yöneticisinin aynı küme kuyruğunun bir eşgörünümünü barındırdığı bir küme yaratabilirsiniz. Bir sıra içindeki tüm iletilerin, kuyruğun aynı örneğine gönderildiğinden emin olun. MQOPEN çağrısında MQOO_BIND_ON_OPEN seçeneğini kullanarak bir ileti dizisi için belirli bir kuyruğa bağ tanımlayabilirsiniz.

Küme iletim kuyrukları

z/OS'dışında, bir kuyruk yöneticisi, birden çok iletim kuyruğundaki bir kümedeki diğer kuyruk yöneticilerine ilişkin iletileri saklayabilir. Birden çok küme iletim kuyruğundaki iletileri iki farklı şekilde saklamak için bir kuyruk yöneticisi yapılandırabilirsiniz. If you set the queue manager attribute DEFCLXQ to KANAL, a different cluster transmission queue is created automatically from SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE for each cluster-sender channel. CLCHNAME iletim kuyruğu seçeneğini bir ya da daha çok küme gönderici kanalına eşleştirmek için ayarluyorsanız, kuyruk yöneticisi eşleşen kanallara ilişkin iletileri bu iletim kuyruğunda saklayabilir.



Uyarı: Ürünün önceki bir sürümünden yükseltelen bir kuyruk yöneticisiyle özel olarak ayrılmış SYSTEM . CLUSTER . TRANSMIT . QUEUES kullanıyorsanız, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE ' in SHARE/NOSHARE seçeneğinin **SHARE** olarak ayarlandığından emin olun.

Başka bir kuyruk yöneticisinde bulunan bir küme kuyruğuna ilişkin ileti, gönderilmeden önce bir küme iletim kuyruğuna yerleştirilir. Bir küme-gönderici kanalı, diğer kuyruk yöneticilerindeki bir küme iletim kuyruğundan iletileri küme alıcılı kanallara aktarır. Varsayılan olarak, sistem tarafından tanımlanmış bir küme iletim kuyruğu, diğer küme kuyruğu yöneticilerine aktarılacak tüm iletileri içerir. Kuyruğa SYSTEM . CLUSTER . TRANSMIT . QUEUE adı verilir. Bir kümenin parçası olan bir kuyruk yöneticisi, bu küme iletim kuyruğunda bulunan iletileri aynı kümedeki diğer kuyruk yöneticilerine gönderebilir.

Tek SYSTEM . CLUSTER . TRANSMIT . QUEUE kuyruğu için bir tanım, z/OS'dışındaki her kuyruk yöneticisinde varsayılan olarak yaratılır.

z/OS'dışındaki altyapılarda, birden çok iletim kuyruğu kullanarak, iletileri diğer kümelenmiş kuyruk yöneticilerine aktarmak için bir kuyruk yöneticisi yapılandırabilirsiniz. Ek küme iletim kuyruklarını el ile tanımlayabilir ya da kuyruk yöneticisinin kuyrukları otomatik olarak yaratması gerekir.

Kuyruk yöneticisi tarafından otomatik olarak kuyruklar yaratılmasını istediğiniz için, DEFCLXQ kuyruk yöneticisi özniteliğini SCTQ 'dan KANAL ' a çevirin. Sonuç olarak kuyruk yöneticisi, yaratılan her bir küme gönderici kanalı için tek bir küme iletim kuyruğu yaratır. The transmission queues are created as permanent dynamic queues from the model queue, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. Kalıcı dinamik kuyruğun her birinin adı SYSTEM . CLUSTER . TRANSMIT . *ChannelName*. Her kalıcı dinamik küme iletim kuyruğunun ilişkili olduğu, küme gönderen kanalının adı, CLCHNAME yerel iletim kuyruğu özniteliğinde ayarlanır. Messages for remote clustered queue managers are placed on the permanent dynamic cluster transmission queue for the associated cluster-sender channel, rather than on SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Küme iletim kuyruklarını el ile yaratmak için, USAGE özniteliği XMITQ olarak ayarlanmış bir yerel kuyruk yaratın ve CLCHNAME özniteliği, bir ya da daha çok küme gönderici kanalına çözülen soysal bir kanal adına ayarlanır; bkz. ClusterChannelAd. Küme iletim kuyruklarını el ile oluşturursanız, iletim kuyruğunu tek bir kümeyle gönderici kanalı ya da birden çok kümeli gönderici kanalı ile ilişkilendirme seçeneğiniz vardır. CLCHNAME özniteliği, adda birden çok genel arama karakteri ("*") yerleştirebileceğiniz genel bir addır.

Kuyruk yöneticisini tam bir havuza bağlamak için el ile oluşturduğunuz ilk küme gönderici kanalları dışında, küme gönderici kanalları otomatik olarak yaratılır. Bunlar, bir küme kuyruk yöneticisine aktarma için bir ileti olduğunda otomatik olarak yaratılır. Bunlar, hedef kuyruk yöneticisiyle ilgili belirli bir küme için küme iletilerini alan, küme alıcı kanalının adıyla aynı adı taşıyan bir adla yaratılırlar.

Küme alıcı kanalları için bir adlandırma kuralını izlerseniz, farklı tipte küme iletilerini farklı iletim kuyruklarına süzen CLCHNAME için soysal bir değer tanımlayabilir. For example, if you follow the naming convention for cluster-receiver channels of *ClusterName.QmgrName*, then the generic name *ClusterName.** filters messages for different clusters onto different transmission queues. İletim kuyruklarını el ile tanımlamanız ve her iletim kuyruğunda CLCHNAME değerini *ClusterName.** değerine ayarlamanız gerekir.

Küme iletim kuyruklarının küme gönderici kanallara ilişkisine ilişkin değişiklikler hemen yürürlüğe girmez. Bir küme gönderen kanalının bakım yaptığı şu anda ilişkili iletim kuyruğu, kümeyle gönderici kanalı tarafından aktarılmakta olan iletileri içerebilir. Yalnızca, ilişkili iletim kuyruğunda hiçbir ileti bir kümeyle gönderici kanalı tarafından işlenmezse, kuyruk yöneticisi, kümenin gönderici kanalının farklı bir iletim kuyruğunu ilişkilendirmesini değiştirebilir. Bu durum, iletim kuyruğunda kümes-gönderen kanalı tarafından işlenecek hiçbir ileti kalmadığında ya da iletilerin işlenmesi askıya alındığında ve küme gönderen kanalının "uçuşta" iletileri olmadığında ortaya çıkabilir. Bu gerçekleştiğinde, küme gönderen kanalı için işlenmemiş iletiler yeni ilişkili iletim kuyruğuna aktarılır ve kümeyle gönderici kanal değişikliklerinin ilişkilendirilmesi gerçekleştirilir.

Bir küme iletim kuyruğuna çözülen bir uzak kuyruk tanımlaması yaratabilirsiniz. In the definition, queue manager QMX is in the same cluster as the local queue manager, and there is no transmission queue, QMX.

```
DEFINE QREMOTE(A) RNAME(B) RQMNNAME(QMX)
```

Kuyruk adı çözülmesi sırasında, küme iletim kuyruğu, varsayılan iletim kuyruğundan önceliklidir. A message put to A is stored on the cluster transmission queue and then sent to the remote queue B on QMX.

Kuyruk yöneticileri, bir kümenin parçası olmayan diğer kuyruk yöneticileriyle de iletişim kurabilirler. Diğer kuyruk yöneticisine, dağıtılmış kuyruklama ortamında olduğu gibi, kanalları ve iletim kuyruğunu tanımlamalısınız.

Not: Uygulamalar, küme iletim kuyruğuna çözülen kuyruklara yazmalı ve küme iletim kuyruğuna doğrudan yazmamalıdır.

Uzak kuyrukların otomatik tanımlaması

Bir kümedeki kuyruk yöneticisinin, kümedeki uzak kuyruklar için uzak kuyruk tanımlamasına gerek yoktur. Küme kuyruk yöneticisi, tam havuzdaki bir uzak kuyruğun yerini bulur. İletiyeye yöneltme bilgileri ekler ve bu bilgileri küme iletim kuyruğuna yerleştirir. WebSphere MQ , iletinin gönderilebilmesi için, otomatik olarak uzak kuyruk tanımlamasıyla eşdeğer bir tanım yaratır.

Otomatik olarak yaratılmış bir uzak kuyruk tanımlamasını değiştiremez ya da silemezsiniz. Ancak, DISPLAY QUEUE **runmqsc** komutunu CLUSTINFO öznitelikle birlikte kullanarak, uzak kuyruk yöneticilerindeki küme kuyrukları da içinde olmak üzere, bir kuyruk yöneticisiyle ilgili tüm yerel kuyrukların yanı sıra tüm küme kuyruklarını da görüntüleyebilirsiniz. Örneğin:

```
DISPLAY QUEUE(*) CLUSINFO
```

İlgili başvurular

[ClusterChannelAd \(MQCHAR20\)](#)

Küme kanalları

Kümenizdeki kuyruk yöneticileri için küme alıcısı ve küme gönderici kanalları tanımlamanız gerekir. Tüm havuzlara özel konular uygulanır.

Kümeler içinde, iletiler küme kuyruğu yöneticileri arasında küme alıcı kanal tanımlamalarına ve küme gönderici kanal tanımlamalarına gerek duyarsanız, özel bir kanal tipi üzerinde dağıtılır.

Küme-gönderici kanalı: CLUSSDR

Bir küme gönderici kanalını, kümedeki her kuyruk yöneticisinde tam bir havuza el ile tanımlayın. Küme gönderici tanımlaması kuyruk yöneticisinin kümeyle ilk iletişim kurmasını gerçekleştirmesini sağlar. Bu ad, kuyruk yöneticisinin küme bilgilerini göndermeyi tercih ettiği tam havuz kuyruk yöneticisini adlandırır. Küme gönderici kanalı, havuz yöneticisi' in durumuna ilişkin değişiklikleri havuza bildirmek için kullanılır. Örneğin, bir kuyruk eklenirse ya da kaldırılırsa. İletileri iletmek için de kullanılır.

Tam havuz kuyruğu yöneticilerinin, birbirlerine işaret eden küme gönderici kanalları vardır. Bunlar, küme durumu değişikliklerini birbiriyle iletişim kurmak için kullanılır.

CLUSTSDR kanal tanımının tam olarak hangi havuzda olduğu küçük önem derecesidir. İlk kişi yapıldıktan sonra, daha fazla küme kuyruk yöneticisi nesnelere gerektiği şekilde otomatik olarak tanımlanır. Kuyruk yöneticisi her tam havuza küme bilgilerini ve her kuyruk yöneticisine ileti gönderebilir.

Tam havuz kuyruğu yöneticilerindeki CLUSSDR tanımlamaları özeldir. Tüm havuzlar yalnızca bu kanallarda akış yaparak değiş tokuş edilen tüm güncellemelere. Yönetici, tam havuzların ağını açık bir şekilde denetler. Denetimcinin, her havuz kuyruk yöneticisinden kümedeki diğer tüm havuz kuyruk yöneticilerinden biri için bir CLUSSDR kanalı tanımlaması gerekir. Yönetici, tam havuz kuyruğu yöneticilerindeki CLUSSDR tanımlarını el ile yapmalıdır ve bunları otomatik olarak tanımlanmaz yapmalıdır.

Küme gönderen kanalları, kısmi bir havuzu tam bir havuza bağlamak için ya da iki tam havuzu birbirine bağlamak için tanımlanmalıdır. Kısmi bir havuza ya da kümede olmayan bir kuyruk yöneticisine hitap eden bir CLUSSDR kanalının el ile yapılandırılması, AMQ9427 ve AMQ9428 gibi hata iletilerine yol açar.

Bu durum bazen geçici bir durum (örneğin, bir tam havuzun konumunu değiştirirken) kaçınılmazdır. Yanlış tanımlamalar, bu hataların yayınlanmasına son verilmesi için en kısa sürede silinmelidir.

Küme-alıcı kanalı: CLUSTRVR

Bir küme alıcı kanalı tanımlaması, bir küme kuyruk yöneticisinin kümedeki diğer kuyruk yöneticilerinden ileti alabileceği bir kanalın sonunu tanımlar.

Bir küme-alıcı kanalı, yerel havuza ilişkin kümeyle ilgili bilgileri de taşıyabilir. Kuyruk alıcı kanalı tanımlanarak, kuyruk yöneticisi, ileti almak için uygun olan diğer küme kuyruğu yöneticilerini gösterir. Her küme kuyruk yöneticisi için en az bir küme alıcı kanalına ihtiyacınız vardır.

Bir CLUSRCVR tanımlaması, diğer kuyruk yöneticilerinin karşılık gelen CLUSSDR kanal tanımlamalarını otomatik olarak tanımlamalarını sağlar.

İlgili kavramlar

[“Küme kanallarının otomatik tanımlaması” sayfa 165](#)

Bir kuyruk yöneticisinin uzak bir hedefe ileti gönderebilmesi için, bir küme gönderici kanalı için bir tanımlaması olmalıdır. İlk CLUSTSDR ve CLUSTRCVR tanımlarını yaparak bir kuyruk yöneticisini bir kümeye tanıttıktan sonra, WebSphere MQ gerektiğinde otomatik olarak küme gönderici kanal tanımlarını yapar. Otomatik olarak tanımlanan küme gönderici kanallarını değiştiremezsiniz. Bir kanal otomatik tanımlama çıkışı kullanarak davranışlarını değiştirebilirsiniz.

Küme kanallarının otomatik tanımlaması

Bir kuyruk yöneticisinin uzak bir hedefe ileti gönderebilmesi için, bir küme gönderici kanalı için bir tanımlaması olmalıdır. İlk CLUSTSDR ve CLUSTRCVR tanımlarını yaparak bir kuyruk yöneticisini bir kümeye tanıttıktan sonra, WebSphere MQ gerektiğinde otomatik olarak küme gönderici kanal tanımlarını yapar. Otomatik olarak tanımlanan küme gönderici kanallarını değiştiremezsiniz. Bir kanal otomatik tanımlama çıkışı kullanarak davranışlarını değiştirebilirsiniz.

Hem küme-gönderici ucu hem de bir kanalın küme alıcı ucu tanımlandığında, kanal başlatılır. Otomatik olarak tanımlanmış bir kanal, artık gerek duyulmadıkça etkin kalır ve olağan bağlantı kesme aralığı kuralları kullanılarak kapatılır.

Otomatik olarak tanımlanan küme gönderici kanalları, giriş kuyruğu yöneticilerindeki ilgili küme alıcı kanalı tanımlamasından özniteliklerini alır. El ile tanımlanmış bir küme gönderici kanalı olsa da, öznitelikleri, karşılık gelen küme alıcı tanımlamasıyla eşleştirdiğinden emin olmak için otomatik olarak değiştirilir. Örneğin, CONNAME parametresindeki bir kapı numarası belirtmeden bir CLUSTUSRVR

tanımladığınızı ve kapı numarası belirten bir CLUSSDR ' nin el ile tanımlanmanızı varsayalım. Otomatik olarak tanımlanan CLUSTSDR el ile tanımlanan şekilde yerine geçtiğinde, kapı numarası (CLUSTRVR' den alınır) boş olur. Varsayılan kapı numarası kullanılır ve kanal başarısız olur.

Otomatik olarak tanımlanmış bir küme gönderici tanımlamasını değiştiremezsiniz.

DISPLAY CHANNEL **runmqsc** komutunu kullanarak otomatik olarak tanımlanmış kanalları görmezsiniz. Otomatik tanımlı kanalları görmek için şu komutu kullanın:

```
DISPLAY CLUSQMGR(qMgrName)
```

Yarattığınız CLUSRCVR kanal tanımlamasına karşılık gelen otomatik tanımlı CLUSTSDR kanalının durumunu görüntülemek için şu komutu kullanın:

```
DISPLAY CHSTATUS(channelName)
```

Bir kümeli gönderici kanalı ya da küme alıcı kanalını uyarlamak için bir kullanıcı çıkış programı yazmak istiyorsanız, WebSphere MQ kanalı otomatik tanımlama çıkışını kullanabilirsiniz. Kanal otomatik tanımlama çıkışını bir küme ortamında aşağıdakileri yapmak için kullanabilirsiniz:

- İletişim tanımlamalarının uyarlanması, SNA LU6.2 adları
- Diğer çıkışları ekleyin ya da kaldırın; örneğin, güvenlik çıkışları
- Kanal çıkışlarının adlarını değiştirin. CLUSSDR kanal çıkış adının CLUSRCVR kanal tanımlamasından otomatik olarak oluşturulduğu için, CLUSSDR kanal çıkışının adını değiştirmelisiniz. Otomatik oluşturulan ad yanlış olabilir ve kanalın iki ucu farklı platformlarda yer alıyorsa, neredeyse kesinlikle yanlış. Çıkış adlarının biçimi farklı platformlarda farklıdır. Örneğin, on Pencereler it is, SCYEXIT(' *drive*:\ *path*\ *library*(*secexit*) ').

Exit names on platforms other than z/OS are of the general form *yol/kitaplık(işlev)*. *işlev* varsa, bu en çok sekiz karakter kullanılır. Ters durumda, *kitaplık*, sekiz karakter olarak kısaltılır. Örneğin,

- /var/mqm/exits/myExit.so(MsgExit) , MSGEXITbiçimine dönüştürür
- /var/mqm/exits/myExit , MYEXITbiçimine dönüştürür
- /var/mqm/exits/myExit.so(ExitLongName) , EXITLONGbiçimine dönüştürür

Giden (TCP) bir kanalı belirli bir IP adresi, kapı ya da kapı aralığı kullanacak şekilde etkinleştirmek için, KAPSAY1C1kanal özneliğini kullanın. KAPSAY1C1 is useful if you have more than one network card and you want a channel to use a specific one for outbound communications.

CLUSSDR kanallarında bir sanal IP adresi belirtmek için, el ile tanımlanmış CLUSTSDR'de LOCLADR ' ın IP adresini kullanın. Kapı aralığını belirtmek için CLUSRCVR' den kapı aralığını kullanın.

Bir kümenin giden iletişim kanallarının belirli bir IP adresine bağlanması için LOCLADDR kullanılması gerekiyorsa, LOCLADDR değerini otomatik olarak tanımlanmış CLUSTSDR kanallarından herhangi birine zorlamak için bir Kanal Otomatik Tanımlama Çıkışı yazmalısınız ve bunu el ile tanımlanmış CLUSSDR kanalından belirtmeniz gerekir.

Tüm kuyruk yöneticileri aynı sunucuda değilse, CLUSTRVR kanalının LOCLADR kanalına bir IP adresi koymayın. LOCLADR IP adresi, CLUSTRVR kanalını kullanarak bağlanan tüm kuyruk yöneticilerinin otomatik olarak tanımlanmış CLUSSDR kanallarına dağıtılır.

Bir CLUSTRVR kanalının LOCLADDR kanalına bir kapı numarası ya da kapı aralığı koyun; bir kümedeki tüm kuyruk yöneticilerinin, tüm giden iletişimleri için belirli bir kapı ya da kapı aralığı kullanmasını istiyorsanız,

distributed Dağıtılmış altyapılarda, tanımlı yerel bir adresi olmayan tüm gönderen kanalları için kullanılacak bir varsayılan yerel adres değeri belirlemek mümkündür. Varsayılan değer, kuyruk yöneticisi başlatılmadan önce MQ_LCLADDR ortam değişkeninin ayarlanarak tanımlanır. Değer biçimi, MQSC özneliği LOCLADDR ' ın eşiyile eşleşir.

Otomatik olarak tanımlanan küme-gönderen kanal tanımları gerçek kanal nesnelere değildir. z/OSdışındaki platformlarda, OAM (nesne yetkilisi yöneticisi) varlıklarının farkında değildir. Otomatik tanımlı küme gönderici kanallarında başlatma, durdurma, ping, ilk duruma getirme ya da çözme komutları yayınlamaya

çalışırsanız, OAM, küme için küme alıcı-alıcı kanalında aynı işlemi gerçekleştirme yetkisine sahip olup olmadığınızı denetler.

Bir WebSphere MQ Sürüm 7 kuyruk yöneticisinden WebSphere MQ Sürüm 7 kuyruk yöneticisinden bir kuyruk yöneticisine giden iletilerin RFH2 gibi uygulama üstbilgilerini kaldırmak için, kümenin PROPCTL 'yi kullanması gerekiyorsa, PROPCTL' yi NONE (Yok) değerine zorlayan bir Kanal Otomatik Tanımlama Çıkışı yazmalısınız. Küme gönderici kanallarının tanımlarına karşılık gelen küme alıcı kanallarına dayalı olması nedeniyle çıkış gereklidir. Önceki düzey küme alıcı kanalının PROPCTL özneliği yoksa, öznelik, otomatik küme gönderen kanalı tarafından COMPAT olarak ayarlanır. Bu öznelik, el ile küme gönderen kanalda nelerin ayarlandığından bağımsız olarak COMPAT olarak ayarlanır.

İlgili başvurular

Yerel Adres (LOCLADDR)

Varsayılan küme nesneleri

WebSphere MQ kümeleri kullanılırken varsayılan küme nesneleri yaratır. They are included in the set of default objects automatically created when you define a queue manager.

MQSC ya da PCF komutlarını çalıştırarak, varsayılan kanal tanımlamalarını diğer kanal tanımlamalarıyla aynı şekilde değiştirebilirsiniz.

SYSTEM . CLUSTER . HISTORY . QUEUE dışında varsayılan kuyruk tanımlamalarını değiştirmeyin.

SYSTEM . CLUSTER . COMMAND . QUEUE

Bir kümedeki her kuyruk yöneticisinin, iletileri tam havuza aktarmak için kullanılan SYSTEM . CLUSTER . COMMAND . QUEUE adlı bir yerel kuyruğu vardır. İleti, kuyruk yöneticisiyle ilgili yeni ya da değiştirilmiş bilgileri ya da diğer kuyruk yöneticilerine ilişkin bilgi isteklerini içerir. SYSTEM . CLUSTER . COMMAND . QUEUE olağan durumda boştur.

SYSTEM . CLUSTER . HISTORY . QUEUE

Bir kümedeki her kuyruk yöneticisinin SYSTEM . CLUSTER . HISTORY . QUEUE adlı bir yerel kuyruğu vardır. SYSTEM . CLUSTER . HISTORY . QUEUE , hizmet amacıyla küme durumu bilgilerinin geçmişi saklamak için kullanılır.

Varsayılan nesne ayarlarında SYSTEM . CLUSTER . HISTORY . QUEUE , PUT(ENABLED) olarak ayarlıdır. Geçmiş derlemeyi gizlemek için ayarı PUT(DISABLED) olarak değiştirin.

SYSTEM . CLUSTER . REPOSITORY . QUEUE

Bir kümedeki her kuyruk yöneticisinin SYSTEM . CLUSTER . REPOSITORY . QUEUE adlı bir yerel kuyruğu vardır. Bu kuyruk, tüm havuz bilgilerini saklamak için kullanılır. Bu kuyruk olağan durumda boş değil.

SYSTEM . CLUSTER . TRANSMIT . QUEUE

Her kuyruk yöneticisinin, SYSTEM . CLUSTER . TRANSMIT . QUEUE adlı yerel bir kuyruk için bir tanımlaması vardır. SYSTEM . CLUSTER . TRANSMIT . QUEUE , tüm iletiler için varsayılan iletim kuyruğudur ve kümeler içinde olan tüm kuyruk yöneticilerine ve kuyruk yöneticilerine iletilir. You can change the default transmission queue for each cluster-sender channel to SYSTEM . CLUSTER . TRANSMIT . *ChannelName*, by changing the queue manager attribute DEFXMITQ. SYSTEM . CLUSTER . TRANSMIT . QUEUE ögesini silemezsiniz. Ayrıca, kullanılan varsayılan iletim kuyruğunun SYSTEM . CLUSTER . TRANSMIT . QUEUE ya da SYSTEM . CLUSTER . TRANSMIT . *ChannelName* olup olmadığını denetleyerek yetki denetimini de tanımlamak için kullanılır.

SYSTEM . DEF . CLUSRCVR

Her bir kümenin SYSTEM . DEF . CLUSRCVR adlı varsayılan CLUSRCVR kanal tanımlaması vardır. SYSTEM . DEF . CLUSRCVR , kümedeki bir kuyruk yöneticisinde bir küme alıcı kanalı yarattığınızda belirtmediğiniz öznelikler için varsayılan değerleri sağlamak üzere kullanılır.

SYSTEM . DEF . CLUSSDR

Her bir kümenin SYSTEM . DEF . CLUSSDR adlı varsayılan CLUSSDR kanal tanımlaması vardır. SYSTEM . DEF . CLUSSDR , kümedeki bir kuyruk yöneticisinde bir küme gönderici kanalı yarattığınızda belirtmediğiniz öznelikler için varsayılan değerleri sağlamak üzere kullanılır.

Küme iletim kuyrukları ve küme-gönderici kanalları

Kümelenmiş kuyruk yöneticileri arasındaki iletiler, küme iletim kuyruklarında saklanır ve küme gönderici kanalları tarafından iletilir.

Küme gönderici kanalını görüntülediğinizde, bu kanal bir iletim kuyrukla ilişkili olduğunu görürsünüz. Herhangi bir zamanda, bir küme-gönderici kanalı bir iletim kuyrukla ilişkilendirilir. Kanalın yapısını değiştirirseniz, başlatma işlemi sırasında başka bir iletim kuyruğu da değişebilir. Kümeleme gönderen kanallarının ilişkilendirildiği iletim kuyruklarını görüntülemek için aşağıdaki MQSC komutunu çalıştırın:

```
DISPLAY CHSTATUS(*) WHERE(CHLTYPE EQ CLUSSDR)
```

```
AMQ8417: Display Channel Status details.  
CHANNEL (TO.QM2)                CHLTYPE (CLUSSDR)  
CONNAME (9.146.163.190(1416))   CURRENT  
QMNAME (QM2)                   STATUS (STOPPED)  
SUBSTATE ( )                   XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

Durdurulan bir küme gönderen kanalının kaydedilmiş kanal durumunda gösterilen iletim kuyruğu, kanal yeniden başlatıldığında değişebilir. [“Varsayılan küme gönderen kanallara göre varsayılan iletim kuyruklarının seçilmesi” sayfa 168](#) Varsayılan iletim kuyruğunu seçme işlemi açıklar; [“Küme gönderen kanallara göre el ile tanımlanmış iletim kuyruklarının seçilmesi” sayfa 169](#) , el ile tanımlanmış bir iletim kuyruğunu seçme işlemi açıklar.

Herhangi bir kümeli gönderici kanalı başlatıldığında, bu kanal, ilişkilendirmesini iletim kuyruklarıyla yeniden denetleyebilir. İletim kuyruklarının yapılandırılması ya da kuyruk yöneticisinin varsayılan değeri değişirse, kanal farklı bir iletim kuyruklarıyla yeniden ilişkilendirilebilir. Kanal, konfigürasyon değişikliğinin bir sonucu olarak farklı bir iletim kuyruğuyla yeniden başlatılırsa, yeni ilişkili iletim kuyruğuna ileti aktarma işlemi gerçekleşir. [“Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır” sayfa 170](#) , bir küme-gönderici kanalının bir iletim kuyruğundan diğerine aktarılmasını açıklar.

Küme gönderen kanalların davranışı, gönderen ve sunucu kanallarından farklıdır. They remain associated with the same transmission queue until the channel attribute **XMITQ** is altered. Bir gönderici ya da sunucu kanalındaki iletim kuyruğu özneliğini değiştirir ve yeniden başlatırsanız, iletiler eski iletim kuyruğundan yenisine aktarılmaz.

Küme gönderici kanalları ile gönderen ya da sunucu kanalları arasındaki diğer bir fark, birden çok küme gönderen kanalının bir küme iletim kuyruğunu açabilmesi, ancak yalnızca bir gönderenin ya da sunucu kanalının olağan bir iletim kuyruğunu açabilmesi. Version 7.5 kümesi bağlantıları tek küme iletim kuyruğunu paylaşılincaya kadar SYSTEM.CLUSTER.TRANSMIT.QUEUE. Version 7.5 ' den itibaren, iletim kuyruklarını paylaşmayan, kümeli gönderici kanallarının seçeneğiniz vardır. Ayrıcalık uygulanmaz; bu, yapılandırmanın bir sonucu olur. Bir iletinin, diğer uygulamalar arasında akan iletilerle hiçbir iletim kuyruklarını ya da kanalı paylaşmaması için bir kümede yer alması için gereken yolu yapılandırabilirsiniz. Bkz. [“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281](#) ve [“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202](#).

Varsayılan küme gönderen kanallara göre varsayılan iletim kuyruklarının seçilmesi

Küme iletim kuyruğu, SYSTEM.CLUSTER.TRANSMIT ile başlayan bir adla ya da el ile tanımlanmış bir kuyrukla başlayan varsayılan bir kuyruktur. Küme-gönderici kanalı, bir küme iletim kuyruğundan iki yoldan biriyle ilişkilendirilir: Varsayılan küme iletim kuyruğu mekanizması ya da el ile yapılandırma ile.

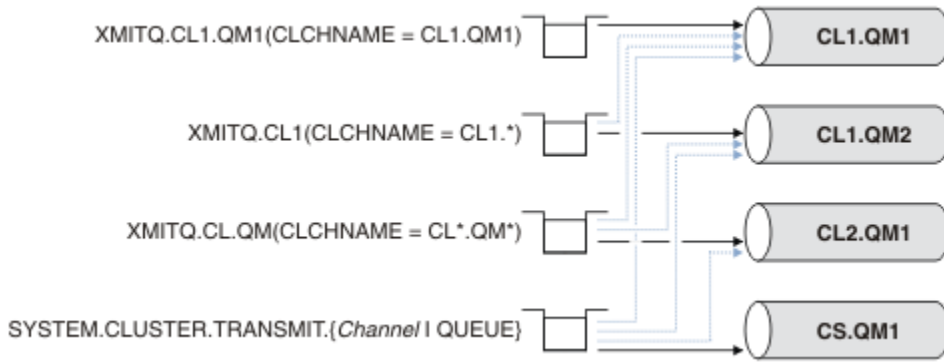
Varsayılan küme iletim kuyruğu, bir kuyruk yöneticisi özneliği (**DEFCLXQ**) olarak ayarlanır. Değeri SCTQ ya da KANAL' dir. Yeni ve geçirilen kuyruk yöneticileri SCTQ değerine ayarlanır. Değeri KANAL olarak değiştirebilirsiniz.

SCTQ ayarlandıysa, varsayılan küme iletim kuyruğu SYSTEM.CLUSTER.TRANSMIT.QUEUE olur. Her küme gönderici kanalı bu kuyruğu açabilir. Kuyruğu açan küme gönderici kanalları, el ile tanımlanmış küme iletim kuyruklarıyla ilişkilendirilmemiş olan kuyruklardır.

KANAL değeri ayarlandıysa, kuyruk yöneticisi her küme gönderici kanalı için ayrı bir kalıcı dinamik iletim kuyruğu yaratabilir. Her bir kuyruk SYSTEM . CLUSTER . TRANSMIT . ChannelName adını taşır ve model kuyruğundan oluşturulur, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. El ile tanımlanmış bir küme iletim kuyrukları ilişkilendirilmemiş olan her bir küme gönderici kanalı, kalıcı dinamik bir küme iletim kuyrukları ilişkilendirilir. Kuyruk, kuyruk yöneticisi tarafından, bu küme gönderen kanal tarafından hizmet verilen küme hedefi için ayrı bir küme iletim kuyruğu gerektirdiğinde ve kuyruk yok olduğunda, kuyruk yöneticisi tarafından yaratılır.

Bazı küme hedefleri, el ile tanımlanmış iletim kuyruklarıyla ilişkili küme gönderen kanallarıyla ve diğer kullanıcılar için varsayılan kuyruk ya da kuyruklar tarafından kullanılabilir. İletim kuyrukları olan kümeli gönderici kanallarında, el ile tanımlanmış iletim kuyrukları her zaman varsayılan iletim kuyruklarına göre önceliklidir.

Küme iletim kuyruklarının önceliği Şekil 24 sayfa 169 içinde gösterilmektedir. Manüel olarak tanımlanmış bir küme iletim kuyrukları ilişkilendirilmemiş tek küme gönderen kanalı CS . QM1. İletim kuyruklarının **CLCHNAME** öznelindeki kanal adlarının hiçbiri CS . QM1 ile eşleşmediği için, el ile tanımlanmış bir iletim kuyruğu ile ilişkilendirilmemiş.



Şekil 24. İletim kuyruğu/küme-gönderici kanal önceliği

Küme gönderen kanallara göre el ile tanımlanmış iletim kuyruklarının seçilmesi

A manually defined queue has the transmission queue attribute **USAGE** attribute set to XMITQ, and the cluster channel name attribute **CLCHNAME** set to a specific or generic channel name.

CLCHNAME kuyruk öznelindeki ad bir kümeyle gönderici kanal adıyla eşleşiyorsa, kanal kuyrukları ilişkilendirilir. Ad, joker karakter içermiyorsa, tam eşleşme ya da ad genel arama karakterleri içeriyorsa, bu ad en iyi eşleşmeye sahip olur.

Birden çok iletim kuyruğundaki **CLCHNAME** tanımlamaları aynı küme-gönderici kanalıyla eşleşirse, tanımlamaların örtüşeceği söylenilir. Belirsizlik ortasını çözmek için, eşleşmeler arasında bir öncelik sırası vardır. Tam eşleşmeler her zaman öncelikli olarak uygulanır. Şekil 24 sayfa 169 , iletim kuyrukları ve kümeli gönderici kanalları arasındaki ilişkileri gösterir. Kara oklar gerçek derneklere, gri oklara, potansiyel derneklere. Şekil 24 sayfa 169 içindeki iletim kuyruklarının öncelik sırası şöyledir:

XMITQ . CL1 . QM1

The transmission queue XMITQ . CL1 . QM1 has its **CLCHNAME** attribute set to CL1 . QM1. The definition of the **CLCHNAME** attribute, CL1 . QM1, has no wildcards, and takes precedence over any other CLCHNAME attributes, defined on other transmission queues, that match with wildcards. The queue manager stores any cluster message that is to be transferred by the CL1 . QM1 cluster-sender channel on the XMITQ . CL1 . QM1 transmission queue. Tek kural dışı durum, birden çok iletim kuyruğunun **CLCHNAME** özneliği CL1 . QM1 olarak ayarlanmış olması olabilir. Bu durumda, kuyruk yöneticisi, bu kuyrukların herhangi birinde CL1 . QM1 küme gönderici kanalına ilişkin iletileri saklar. Kanal başlatıldığında bir kuyruk keyfi olarak seçilir. Kanal yeniden başlatıldığında, farklı bir kuyruk seçebilir.

XMITQ . CL1

The transmission queue XMITQ . CL1 has its **CLCHNAME** attribute set to CL1 . *. The definition of the **CLCHNAME** attribute, CL1 . *, has one trailing wildcard, which matches the name of any cluster-

sender channel that starts with CL1 . . The queue manager stores any cluster message that is to be transferred by any cluster-sender channel whose name begins with CL1 . on the transmission queue XMITQ . CL1, unless there is a transmission queue with a more specific match, such as the queue XMITQ . CL1 . QM1. Sondaki genel arama karakteri, tanımlamayı, genel arama karakteri olmayan bir tanımlamadan daha az, birden çok genel arama karakteri içeren bir tanımlamadan ya da daha çok izleyen karakterlerle biten genel arama karakterlerinden daha belirgin kılar.

XMITQ . CL . QM

XMITQ . CL . QM is the name of the transmission queue with its **CLCHNAME** attribute set to CL* . QM* . CL* . QM* tanımı, CL . ile başlayan herhangi bir küme gönderen kanalının adıyla eşleşen iki genel arama karakterine sahiptir ve QM ile birlikte ya da endsile biter. Eşleşme, bir genel arama karakteriyle eşleşenden daha az belli.

SYSTEM . CLUSTER . TRANSMIT . channelName | QUEUE

Hiçbir iletim kuyruğu, kuyruk yöneticisinin kullanacağı küme gönderen kanalının adıyla eşleşen bir **CLCHNAME** özniteliğine sahip değilse, kuyruk yöneticisi varsayılan küme iletim kuyruğunu kullanır. Varsayılan küme iletim kuyruğu, tek sistem kümesi iletim kuyruğu SYSTEM . CLUSTER . TRANSMIT . QUEUE ya da kuyruk yöneticisinin belirli bir küme gönderici kanalı için oluşturduğu sistem kümesi iletim kuyruğudur, SYSTEM . CLUSTER . TRANSMIT . channelName . Varsayılan değer olan kuyruk, kuyruk yöneticisi **DEFXMITQ** özniteliğinin ayarına bağlıdır.

İpucu: Çakışan tanımlamalara ilişkin açık bir gereksiniminiz yoksa, bunların anlaşılması güç olan karmaşık yapılandırmalara yol açabilecekleri gibi onlardan kaçınınız.

Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır

To change the association of cluster-sender channels with cluster transmission queues, change the **CLCHNAME** parameter of any transmission queue or the queue manager parameter **DEFCLXQ** at any time. Hiçbir şey hemen olmaz. Değişiklikler yalnızca bir kanal başlatıldığında gerçekleşir. Başlatıldığında, aynı iletim kuyruğundan iletileri iletmeye devam edilip etmeyeceğini denetler. Üç tür değişiklik, bir küme gönderici kanalının iletim kuyruğuyla ilişkilendirilmesini değiştirir.

1. Küme gönderici kanalının şu anda daha az özel ya da boş olması ile ilişkilendirildiği iletim kuyruğunun **CLCHNAME** parametresini yeniden tanımlamak ya da kanal durdurulduğunda küme iletim kuyruğunu silme işlemi.

Diğer bir küme iletim kuyruğu artık kanal adı için daha iyi bir eşleşme olabilir. Ya da, başka bir iletim kuyruğu, kümeyle gönderici kanalın adıyla eşleşmiyorsa, ilişkilendirmenin varsayılan iletim kuyruğuna geri çevrilmesi gerekir.

2. Diğer herhangi bir küme iletim kuyruğunun **CLCHNAME** parametresini yeniden tanımlama ya da bir küme iletim kuyruğu ekleme.

Başka bir iletim kuyruğunun **CLCHNAME** parametresi, küme gönderen kanalı için şu anda ilişkilendirilmiş olan iletim kuyruğundan daha iyi bir küme göndericisi kanalı olabilir. Küme gönderici kanalı varsayılan bir küme iletim kuyruğuyla ilişkilendirilmişse, el ile tanımlanmış bir küme iletim kuyruğuyla ilişkilendirilebilir.

3. Küme gönderici kanalı şu anda varsayılan bir küme iletim kuyruğuyla ilişkilendirilmişse, **DEFCLXQ** kuyruk yöneticisi değiştirilmesinin değiştirilmesi gerekir.

Bir küme gönderici kanalı değişirse, kanal başlatıldığında ilişkilendirmesini yeni iletim kuyruğuna geçirir. Anahtar sırasında, iletilerin kaybolmamasını sağlar. İletiler, yeni iletim kuyruğuna, kanalın iletileri uzak kuyruk yöneticisine aktardığı sırayla aktarılır.

Unutmayın: Bir kümedeki iletilerin herhangi bir iletimiyle ortak olarak, teslim edilmesi gereken iletilerin sırayla teslim edilmesini sağlamak için iletileri gruplara yerleştirmeniz gerekir. Nadiren, iletiler bir kümede sıradan çıkabilirler.

Anahtar işlemi aşağıdaki işlem adımlarından geçer. Anahtar işlemi kesintiye uğratılırsa, kanal yeniden başlatıldığında yürürlükteki işlem adımı sürdürülür.

1. Adım-İletilerin özgün iletim kuyruğundan işlenmesini sağlar

Küme gönderici kanalı, yeni iletim kuyruğuyla ilişkilendirilir ve bu kanal diğer kümeyle gönderenler kanallarıyla paylaşılabilir. Küme gönderen kanalına ilişkin iletiler, özgün iletim kuyruğuna yerleştirmeye devam eder. Geçiş anahtarı işlemi, iletilerin özgün iletim kuyruğundan yeni iletim kuyruğuna aktarımlarını aktarır. Küme gönderen kanalı, yeni iletim kuyruğundan iletileri küme alıcı kanalına iletir. Kanal durumu, küme gönderici kanalının eski iletim kuyruğuyla ilişkili olmaya devam ediyor olduğunu gösterir.

switch işlemi yeni gelen mesajları da aktarmaya devam ediyor. Bu adım, anahtar işlemi tarafından iletilecek geri kalan iletilerin sayısı sifıra ulaşıncaya kadar devam eder. İletilerin sayısı sifıra ulaştığında, yordam 2. adıma geçer.

Adım 1 'de, kanala ilişkin disk etkinliği artar. Kalıcı iletiler, ilk iletim kuyruğunda ve ikinci iletim kuyruğunda kesinleştirilir. Bu disk etkinliği, iletilerin olağan şekilde aktarılması bir parçası olarak iletim kuyruğundan yerleştirildiğinde ve kaldırıldığında kesinleştirilmekte olan iletilere ek nitelidir. İdeal olarak, geçiş işlemi sırasında hiçbir ileti gelmez, böylece geçiş mümkün olan en hızlı şekilde gerçekleşebilir. İleti geldiye, bunlar anahtar işlemi tarafından işlenir.

Adım 2-Yeni iletim kuyruğundan ileti işleme

Küme gönderici kanalı için özgün iletim kuyruğunda hiçbir ileti kalmadığı anda, yeni iletiler doğrudan yeni iletim kuyruğuna yerleştirilir. Kanal durumu, küme gönderici kanalının yeni iletim kuyruğuyla ilişkili olduğunu gösterir. Kuyruk yöneticisi hata günlüğüne aşağıdaki ileti yazıldı: "AMQ7341 The transmission queue for channel *ChannelName* is *QueueName*."

Birden çok küme iletim kuyruğu ve küme iletim kuyruğu öznitelikleri

Küme iletilerini, iletileri tek bir küme iletim kuyruğunda ya da birden çok kuyrukta saklayan farklı kuyruk yöneticilerine iletme seçeneğiniz vardır. Bir kuyrukla, birden çok kuyrukla birden çok kümeniz olacak şekilde ayarlamak ve sorgulamak için tek bir küme iletim kuyruğu özniteliği takımınız vardır. Bazı öznitelikler için, birden çok küme olması bir avantajdır: örneğin, kuyruk derinliğini sorgulamak, size kaç iletinin bir ya da bir dizi kanal tarafından iletileceğini, tüm kanalların yerine, bir ya da daha çok kanal tarafından iltılmeyi beklediğini gösterir. Diğer öznitelikler için, birden çok küme olması bir dezavantaj olabilir: Örneğin, her küme iletim kuyruğu için aynı erişim izinlerini yapılandırmak istemeyenlersiniz. Bu nedenle, erişim izinleri her zaman, belirli bir küme iletim kuyruğuna ilişkin tanıtlara karşı değil, SYSTEM . CLUSTER . TRANSMIT . QUEUEprofiline yönelik olarak denetlenir. Daha ayrıntılı güvenlik denetimleri uygulamak istiyorsanız, bkz. "Erişim denetimi ve birden çok küme iletim kuyruğu" sayfa 157.

Birden çok küme-gönderici kanalı ve birden çok iletim kuyruğu

Kuyruk yöneticisi, bir iletiyi küme gönderici kanalına iletmeden önce, bir küme iletim kuyruğunda saklar. İleti için hedefe bağlı bir küme gönderici kanalı seçer. Tümü aynı hedefe bağlanan, kümeli gönderici kanallardan bir seçeneği olabilir. Hedef, tek bir kuyruk yöneticisine birden çok kümeyle gönderen kanalla bağlanmış aynı fiziksel kuyruksa olabilir. Hedef, aynı kümedeki farklı kuyruk yöneticilerinde bulunan aynı kuyruk adına sahip birçok fiziksel kuyruklar da olabilir. Bir hedefe bağlanan, kümeli gönderici kanallarının seçimi, iş yükü dengeleme algoritması bir hedef seçiyor. Seçenek, bazı etkenlere bağlıdır; bkz. Küme iş yükü yönetimi algoritması.

Şekil 25 sayfa 172, CL1 . QM1, CL1 . QM2 ve CS . QM1 ' de, aynı hedefe yönlendirebilecek tüm kanallardır. For example, if you define Q1 in CL1 on QM1 and QM2 then CL1 . QM1 and CL1 . QM2 both provide routes to the same destination, Q1, on two different queue managers. If the channel CS . QM1 is also in CL1, it too is a channel that a message for Q1 can take. CS . QM1 küme üyeliği bir küme adı listesi tarafından tanımlanabilir; bu nedenle, kanal adı, yapımına bir küme adı eklemes. Depending on the workload balancing parameters, and the sending application, some messages for Q1 might be placed on each of the transmission queues, XMITQ . CL1 . QM1, XMITQ . CL1 and SYSTEM . CLUSTER . TRANSMIT . CS . QM1.

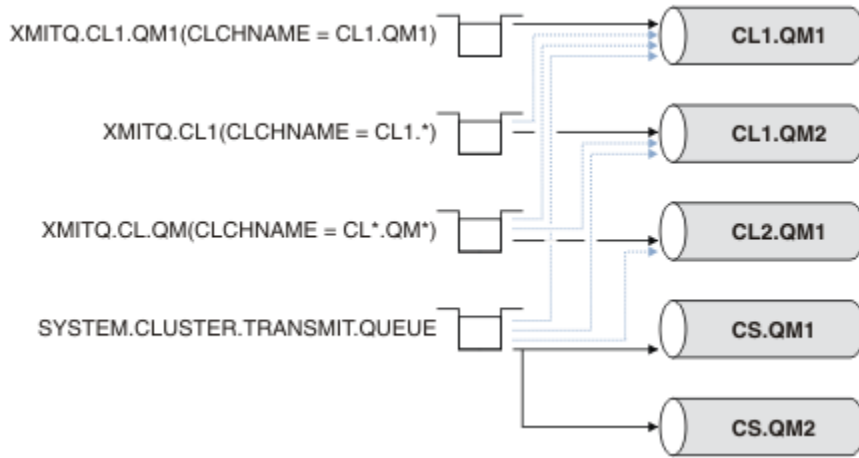
İleti trafiğini ayırmak istiyorsanız, aynı hedefe ilişkin iletiler kuyrukları ya da kanalları farklı hedefler için ileti ile paylaşmazsa, önce trafiği farklı küme gönderen kanallarına nasıl bölebildiğinizi ve daha sonra, belirli bir kanala ilişkin iletileri farklı bir iletim kuyruğunda nasıl ayracağını göz önünde bulundurabilmelisiniz. Aynı kümedeki küme kuyrukları, aynı kuyruk yöneticisinde, olağan durumda aynı küme kanallarını paylaşır. Tek başına birden çok küme iletim kuyruklarının tanımlanması, küme ileti

trafiğini farklı kuyruklara ayırmak için yeterli değildir. Farklı hedef kuyrukları için iletileri farklı kanallara ayırmazsanız, iletiler aynı küme iletim kuyruğunu paylaşır.

İletilerin üstlendiği kanalları birbirinden ayırmak için basit bir yol, birden çok küme yaratmaktan başka bir şey değildir. Her bir kümedeki herhangi bir kuyruk yöneticisinde, yalnızca bir küme kuyruğu tanımlayın. Daha sonra, her küme/kuyruk yöneticisi birleşimi için farklı bir küme alıcılı kanal tanımlıyorsanız, her küme kuyruğuna ilişkin iletiler, diğer küme kuyruklarına ilişkin iletileri içeren bir küme kanalını paylaşmaz. Küme kanalları için ayrı iletim kuyrukları tanımlarsanız, gönderme kuyruğu yöneticisi iletileri her iletim kuyruğunda yalnızca tek bir küme kuyruğu için saklar. Örneğin, iki küme kuyruklarının kaynak paylaşılmasını istiyorsanız, bunları aynı kuyruk yöneticisinde farklı kümelere yerleştirebilir ya da aynı kümedeki farklı kuyruk yöneticilerine yerleştirebilirsiniz.

Küme iletim kuyruğu seçimi, iş yükü dengeleme algoritmasını etkilemez. İş yükü dengeleme algoritması, iletiyi iletmek için hangi küme gönderen kanalını seçer. Bu ileti, iletiyi, o kanala göre hizmet verilen iletim kuyruğuna yerleştirir. İş yükü dengeleme algoritması yeniden seçim yapmak için çağrılırsa, örneğin kanal durduğunda, iletiyi iletmek için farklı bir kanal seçmiş olabilir. Farklı bir kanal seçerse ve yeni kanal iletileri farklı bir küme iletim kuyruğundan iletirse, iş yükü dengeleme algoritması iletiyi diğer iletim kuyruğuna aktarır.

Şekil 25 sayfa 172' ta, iki kümeli gönderici kanalı, CS .QM1 ve CS .QM2 varsayılan sistem iletim kuyrukları ilişkilendirilir. İş yükü dengeleme algoritması SYSTEM . CLUSTER . TRANSMIT . QUEUE' de ya da başka bir küme iletim kuyruğunda bir ileti saklarken, iletiyi iletmek için kullanılan küme gönderen kanalının adı, iletinin ilinti tanıtıcısında saklanır. Her kanal, yalnızca kanal adı ile ilinti tanıtıcısıyla eşleşen iletileri iletir.



Şekil 25. Birden çok küme gönderen kanalı

CS . QM1 durduğunda, o küme gönderen kanalına ilişkin iletim kuyruğunda bulunan iletiler incelenir. Başka bir kanal tarafından iletilebilecek bu iletiler, iş yükü dengeleme algoritması tarafından yeniden işlenir. İlintilendirme tanıtıcısı, alternatif bir küme gönderen kanal adına sıfırlanıyor. Alternatif kümeli gönderici kanalı CS . QM2 ise, ileti SYSTEM . CLUSTER . TRANSMIT . QUEUE üzerinde kalır. Alternatif kanal CL1 . QM1 ise, iş yükü dengeleme algoritması iletiyi XMITQ . CL1 . QM1'e aktarır. Küme gönderen kanal yeniden başlatıldığında, yeni iletiler ve farklı bir küme gönderici kanalı için işaretlenmemiş iletiler, kanal tarafından yeniden aktarılır.

Çalışmakta olan bir sistemdeki iletim kuyrukları ve küme gönderici kanalları arasındaki ilişkiyi değiştirebilirsiniz. Bir iletim kuyruğunda bir CLCHNAME parametresini değiştirebilir ya da **DEFCLXQ** kuyruk yöneticisi değiştirgesini değiştirebilirsiniz. Değişiklik yeniden başlatmalarından etkilenen bir kanal, iletim kuyruğu geçişi işlemini başlatır; bkz. [“Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır” sayfa 170.](#)

Kanal yeniden başlatıldığında iletim kuyruğunu değiştirme işlemi başlar. İş yükü yeniden dengeleme işlemi, kanal durdurulduğunda başlar. İki işlem koşut olarak çalışabilir.

Basit vaka, bir küme gönderici kanalının durdurulurken, yeniden dengeleme işleminin, kuyruklardaki iletileri iletemeyecek olan küme gönderen kanalını değiştirmesine neden olmaz. Bu durumda, başka bir küme-gönderici kanalı iletileri doğru hedefe iletmez. İletileri hedefine iletilebilmek için hiçbir alternatif

kümes-gönderici kanalı olmadan, bu iletiler, küme gönderen kanal durduktan sonra aynı küme gönderici kanalı için işaretlenir. Kanal başlatıldığında, bir anahtar beklemede olduğunda, geçiş işlemleri iletileri, aynı küme gönderici kanalı tarafından işlendikleri farklı bir iletim kuyruğuna taşır.

Daha karmaşık bir durum ise, birden çok küme gönderen kanalının bazı iletileri aynı hedefe işleyebildikleri. İletim kuyruğu anahtarını tetiklemek için, küme gönderici kanalını durdurup yeniden başlatın. Birçok durumda, kanalı yeniden başlattığınızda, iş yükü dengeleme algoritması, özgün iletim kuyruğundan farklı küme gönderen kanallarının sunduğu iletileri farklı iletim kuyruklarına taşımıştır. Yeni iletim kuyruğuna yalnızca, farklı bir kümeyle gönderici kanal tarafından iletilmeyecek iletiler aktarılabilir. Bazı durumlarda, kanal hızlı bir şekilde yeniden başlatılırsa, iş yükü dengeleme algoritmasıyla aktarılabilen bazı iletiler kalır. Bu durumda, geri kalan bazı iletiler, iş yükü dengeleme işlemi tarafından ve bazıları iletim kuyruğunu değiştirme işlemi tarafından değiştirilir.

İlgili kavramlar

“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

“Günlüğün büyüklüğünün hesaplanması” sayfa 395

Bir kuyruk yöneticisi için günlük kaydı gereksinmesi büyüklüğünün hesaplanması.

İlgili görevler

“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'deki altyapılar için geçerlidir.

“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

“Bir kümeyle kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194

Yarattığınız kümeyle kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra, ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra, ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

Tüm havuzları tutmak için küme kuyruğu yöneticilerinin nasıl seçileceği

Her bir kümede en az bir adet ve tercihen iki kuyruk yöneticisi tam olarak havuz tutmasını tercih etmelisiniz. İki tam havuz, en istisnai durumlar için yeterlidir. Mümkünse, sağlam ve kalıcı olarak bağlantılı platformlarda barındırılan, çakışan kesintiler olmayan ve coğrafi olarak merkezi bir konumda olan kuyruk yöneticilerini seçin. Ayrıca, sistemleri tam havuz anasistemleri olarak ayırmanın yanı sıra, bu sistemleri başka görevler için kullanmamayı da göz önünde bulundurun.

Tam havuzlar , kümenin durumunu tam olarak koruyan kuyruk yöneticileridir. Bu bilgileri paylaşmak için, her tam havuz CLUSSDR kanalları (ve karşılık gelen CLUSRCVR tanımlamaları) tarafından kümedeki diğer tüm tam havuzlarla bağlantılıdır. Bu kanalları el ile tanımlamanız gerekir.



Şekil 26. İki bağlantılı tam havuz.

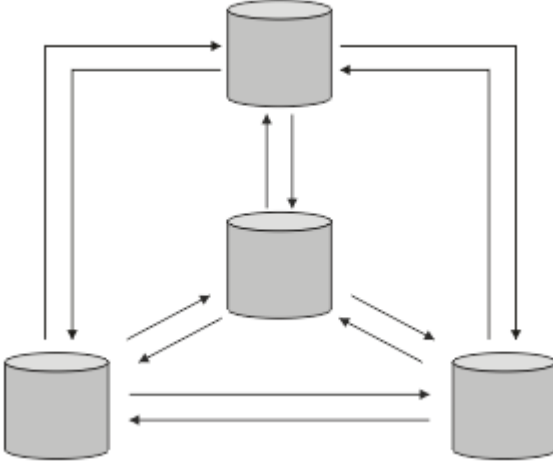
Kümedeki diğer her kuyruk yöneticisi, bir *kısmi havuz* içindeki kümenin durumu hakkında şu anda ne bilse de bir resmi saklar. Bu kuyruk yöneticileri, kendilerine ilişkin bilgileri yayınlamaya ve kullanılabilir iki tam havuzu kullanarak diğer kuyruk yöneticilerine ilişkin bilgi ister. Seçilen bir tam havuz kullanılamıyorsa, başka bir havuz kullanılır. Seçilen tam havuz yeniden kullanılabilir duruma geldiğinde, en son yeni ve değiştirilmiş bilgileri diğer kullanıcıların adımlarına devam etmek üzere toplar. Tüm havuzların tümü hizmet dışına çıkıyorsa, diğer kuyruk yöneticileri, kısmi havuzlarındaki bilgileri kullanır. Ancak, bunlar sahip oldukları bilgileri kullanmakta sınırlanmıştır; yeni bilgiler ve güncellemeler için istekler işlenemez. Tüm havuzlar ağa yeniden bağlandığında, tüm havuzları (hem tam hem de kısmi) güncel olarak getirmek için iletiler değiştirilir.

Tüm havuzların tahsisini planlarken aşağıdaki noktaları göz önünde bulundurulması gerekir:

- Tüm havuzları tutmak için seçilen kuyruk yöneticilerinin güvenilir ve yönetimli olması gerekir. Sağlam ve kalıcı olarak bağlantılı bir platformda barındırılan kuyruk yöneticilerini seçin.
- Tüm havuzlarınıza ev sahipliği yapan sistemler için planlanmış kesintileri göz önünde bulundurun ve bu kesintilerin denk gelmesi durumunda olmamalarını sağlayın.
- Ağ performansını göz önünde bulundurun: Coğrafi konumda bulunan ya da kümedeki diğer kuyruk yöneticileriyle aynı sistemi paylaşan kuyruk yöneticilerini seçin.
- Kuyruk yöneticisinin birden çok küme üyesi olup olmadığını dikkate alın. Bu avantajın, kuyruk yöneticisinin ne kadar meşgul olmasını beklediğiniz konusunda dengeli olması koşuluyla, aynı kuyruk yöneticisini birden çok küme için tüm havuzları barındırmak üzere kullanması için uygun bir şekilde uygun bir yönetim olabilir.
- Bazı sistemleri yalnızca tam havuzları içerecek şekilde ayırmanın ve bu sistemleri başka herhangi bir görev için kullanılmayalı olarak değerlendirin. Bu, bu sistemlerin yalnızca kuyruk yöneticisi yapılandırması için bakım gerektirmesini ve diğer iş uygulamalarının bakımına ilişkin hizmetten kaldırılmamasını sağlar. Ayrıca, havuzu koruma görevinin sistem kaynakları için uygulamalarla rekabet etmemesini sağlar. Bu özellikle, tam havuzların küme durumunu korumada çok daha yüksek bir iş yüküne sahip olduğu büyük kümelerde (örneğin, binden fazla kuyruk yöneticisi kümelerine) yararlı olabilir.

İki tam havuzdan daha fazla havuz olması mümkün, ancak nadiren önerilir. Nesne tanımlamaları (yani, kuyruklar, konular ve kanallar) kullanılabilir tüm tüm havuzlara akışı olsa da, kısmi bir havuzdan en çok iki tam havuza doğru akış sağlar. Bu, iki tam havuzun tanımlandığı ve iki tam havuzun kullanılamaz duruma geldiği anlamına gelir; bazı kısmi havuzlar bekledikleri güncellemeleri alamayabilir. Bkz. [MQ Cluster ürünleri: Neden yalnızca iki Tam Havuz?](#)

Var olan tam havuzları, yeni donanıma ya da yeni kuyruk yöneticilerine geçirirken, ikiden fazla tam havuz tanımlamayı yararlı bulabileceğiniz bir durum. Bu durumda, önceki tam havuzları kaldırmadan önce, yeni tam havuzları tanıtmalı ve tam olarak doldurulduklarını onaylamanız gerekir. Tam bir havuz eklediğinizde, bunu CLUSSDR kanallarıyla diğer tüm havuzlara doğrudan bağlamanız gerektiğini unutmayın.



Şekil 27. İki bağlantılı tam havuz sayısı

İlgili bilgiler

MQ Cluster ürünleri: Neden yalnızca iki Tam Havuz?

MQ Kümesi ne kadar büyük olabilir?

Küme düzenleme

Hangi kuyruk yöneticilerinin hangi tam havuza bağlanacağı seçin. Performans etkisini, kuyruk yöneticisi sürümünü ve birden çok CLUSSDR kanalının istenip istenmediğini göz önünde bulundurun.

Tam havuzları tutmak için kuyruk yöneticilerini seçmiş olması, hangi kuyruk yöneticilerinin hangi tam havuza bağlanmaya karar vereceğine karar vermeniz gerekir. CLUSSDR kanal tanımlaması, bir kuyruk yöneticisini kümedeki diğer tam havuzlarla ilgili olarak bulunduğu tam havuza bağlar. Bundan sonra, kuyruk yöneticisi iki tam havuza ileti gönderir. Her zaman önce bir CLUSSDR kanal tanımlamasına sahip olduğu bir kanal tanımlamasını kullanmayı dener. Bir kuyruk yöneticisini tüm havuzla bağlantıyla ilişkilendirmeyi seçebilirsiniz. Seçim yaparken, yapılandırmanın topolojisini ve kuyruk yöneticilerinin fiziksel ya da coğrafi konumunu göz önünde bulundurun.

Tüm küme bilgileri iki tam havuza gönderildiğinden, ikinci bir CLUSTSDR kanal tanımlaması yapmak istediğiniz durumlar olabilir. Geniş bir alana yayılmış birçok tam havuzu olan bir kümede ikinci bir CLUSSDR kanalı tanımlayabilirsiniz. Daha sonra, bilgilerinizin gönderildiği iki tam havuzun hangilerinin gönderileceğini denetleyebilirsiniz.

Küme adlandırma kuralları

Kuyruk yöneticisinin ait olduğu kümeyi tanıtan bir adlandırma kuralı kullanarak aynı kümedeki kuyruk yöneticilerini adlandırmayı düşünün. Kanal adları için benzer bir adlandırma kuralı kullanın ve kanal özelliklerini açıklayacak şekilde genişletin.

MQ Cluster ürünlerini adlandırırken en iyi uygulamalar

Küme adları en çok 48 karakter olabilese de, diğer nesnelere adlandırma kuralları uygulanırken görelilik olarak kısa küme adları yararlı olur. Bkz. “Küme kanalı adlarını seçerken en iyi uygulamalar” sayfa 176.

Bir küme adı seçerken, genellikle 'içerik' yerine kümenin 'amacını' (uzun ömürlü olması muhtemeldir) temsil etmek yararlı olur. Örneğin, 'QM1_QM2_QM3_CLUS' yerine 'B2BPROD' veya 'ACTTEST'.

Küme Kuyruğu Yöneticisi adlarını seçerken en iyi uygulamalar

Yeni bir küme ve üyeleri sıfırdan oluşturuyorsanız, kuyruk yöneticileri için küme kullanımını yansıtan bir adlandırma kuralı düşünün. Her kuyruk yöneticisinin farklı bir adı olmalıdır. Ancak, bir kümedeki kuyruk

yöneticilerine mantıksal gruplamaların (örneğin, 'ACTTQM1, ACTTQM2) tanımlanmasına ve anımsamasına yardımcı olması için benzer adlar verebilirsiniz.

Görece kısa kuyruk yöneticisi adları (örneğin, 8 karakterden kısa), kanal adları için sonraki bölümde açıklanan kuralı ya da benzeri bir şeyi kullanmayı seçerseniz yardımcı olur.

Küme kanalı adlarını seçerken en iyi uygulamalar

Kuyruk yöneticileri ve kümelerin adları en çok 48 karakterden oluşabildiğinden ve bir kanal adı 20 karakterle sınırlandırıldığından, bir projenin ortasında adlandırma kuralını değiştirmek zorunda kalmamak için ilk adlandırma nesnelere dikkatli olun (önceki bölüme bakın).

Kanalları tanımlarken, kümedeki herhangi bir kuyruk yöneticisinde otomatik olarak yaratılan küme gönderen kanallarının adlarını, kümedeki alıcı kuyruk yöneticisinde yapılandırılan karşılık gelen küme-alıcı kanalından aldığı ve bu nedenle bu kanalların benzersiz olması ve kümedeki uzak kuyruk yöneticilerine anlamlı olması gerektiğini unutmayın.

Yaygın bir yaklaşım, küme adının önünde bulunan kuyruk yöneticisi adını kullanmaktır. Örneğin, küme adı CLUSTER1 ve kuyruk yöneticileri QM1, QM2 ise, küme alıcı kanalları CLUSTER1.QM1, CLUSTER1.QM2 olur.

Kanalların farklı öncelikleri varsa ya da farklı iletişim kuralları kullanıyorsa, bu kuralı genişletebilirsiniz. Örneğin:

- CLUSTER1.QM1.S1
- CLUSTER1.QM1.N3
- CLUSTER1.QM1.T4

Bu örnekte, S1 ilk SNA kanalı, N3 üç ağ önceliğine sahip NetBIOS kanalı ve T4 IPV4 ağı kullanan TCP/IP olabilir.

Paylaşılan kanal tanımlarını adlandırma

Tek bir kanal tanımı birden çok küme arasında paylaşılabilir; bu durumda, burada önerilen adlandırma kurallarının değiştirilmesi gerekir. Ancak, [Kanal tanımlarının yönetilmesi](#) konusunda açıklandığı gibi, her durumda her küme için ayrı kanallar tanımlanması tercih edilir.

Eski kanal adlandırma kuralları

Küme ortamlarının dışında geçmişte 'FROMQM.TO.TARGETQM' adlandırma kuralını kullanmak yaygındır, bu nedenle var olan kümelerin benzer bir şey kullandığını bulabilirsiniz (CLUSTER.TO.TARGET gibi). Yeni bir küme adlandırma şemasının parçası olarak bu önerilmez; kanal adınız içinde 'yararlı' bilgileri iletmek için kullanılacak karakterleri daha da azaltır.

Çakışan kümeler

Çakışan kümeler, ek yönetim yetenekleri sağlar. Çakışan kümeleri denetlemek için gereken komut sayısını azaltmak için ad listelerini kullanın.

Çakışan kümeleri oluşturabilirsiniz. Çakışan kümeleri tanımlayabileceğiniz bir dizi nedeni vardır; örneğin:

- Farklı organizasyonların kendi yönetimlerine sahip olmasını sağlamak.
- Bağımsız uygulamaların ayrı olarak denetlenmesine izin vermek için.
- Hizmet sınıfları yaratmak için.

Şekil 28 sayfa 177'inde, kuyruk yöneticisi STF2 hem kümelerin bir üyesidir. Kuyruk yöneticisi birden çok kümenin üyesi olduğunda, gereksinim duyduğunuz tanımlama sayısını azaltmak için ad listelerinden yararlanabilirsiniz. Ad listeleri, ad listesi (örneğin, küme adları) içerir. Kümeleri adlandırma adına bir ad listesi oluşturabilirsiniz. Specify the namelist on the ALTER QMGR command for STF2 to make it a full repository queue manager for both clusters.

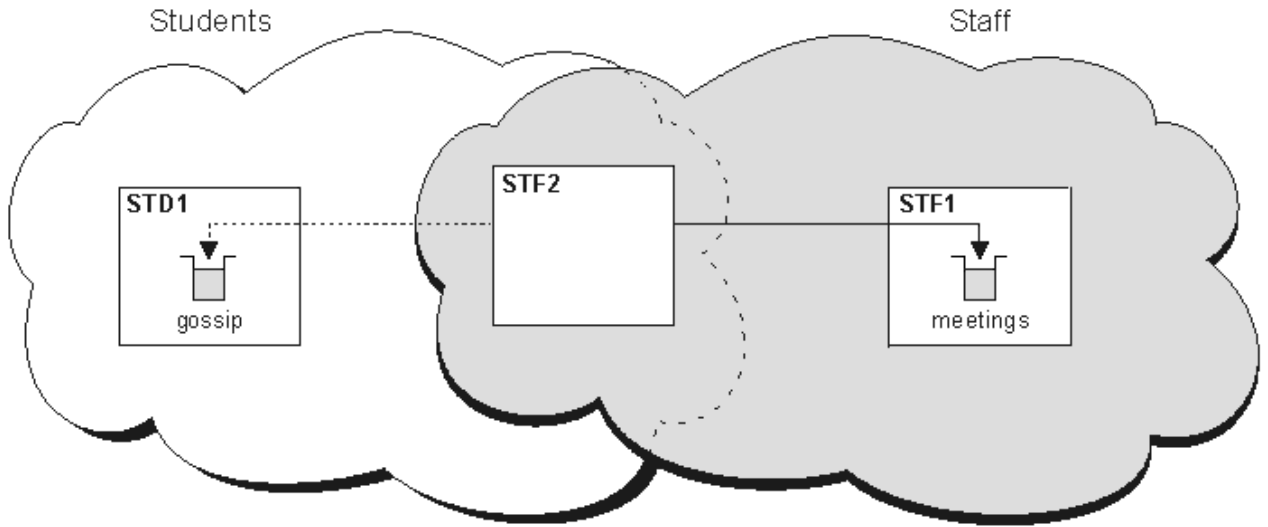
Ağınızda birden çok kümeniz varsa, bu kümeden farklı adlar vermelisiniz. Aynı adı taşıyan iki küme hiç birleştirilirse, bunları yeniden ayırmak mümkün değildir. Ayrıca, kümelere ve kanallara farklı adlar vermek

de iyi bir fikir. Çıkışa DISPLAY komutlarından baktığınızda daha kolay ayırt edilirler. Kuyruk yöneticisi adlarının düzgün çalışabilmek için bir kümede benzersiz olması gerekir.

Hizmet sınıflarının tanımlanması

Her bir öğrenci üyesi ve her öğrenci için kuyruk yöneticisi olan bir üniversite düşünün. personel üyeleri arasındaki mesajlar yüksek öncelikli ve yüksek bant genişliğine sahip kanallarda seyahat etmek. öğrenciler arasındaki mesajlar daha ucuz, daha yavaş kanallara seyahat etmek. Bu ağı geleneksel dağıtılmış kuyruklama tekniklerini kullanarak ayarlayabilirsiniz. WebSphere MQ , hedef kuyruk adı ve kuyruk yöneticisi adına bakarak hangi kanalların kullanılacağını seçer.

Çalışanları ve öğrencileri ayırt etmek için, kuyruk yöneticilerini Şekil 28 sayfa 177' ta gösterildiği gibi iki küme halinde gruplayabilirsiniz. WebSphere MQ , iletileri yalnızca o kümede tanımlı olan kanalların üzerinde bulunan personel kümesindeki toplantı kuyruğuna taşır. Öğrenci kümesindeki dedikodu kuyruğuna ilişkin iletiler, o kümede tanımlı olan kanalların üzerinden gider ve uygun hizmet sınıfını alır.



Şekil 28. Hizmet sınıfları

Kümeleme ipuçları

Kümelemeyi kullanmadan önce sistemlerinizde ya da uygulamalarınızda bazı değişiklikler yapmanız gerekebilir. dağıtık kuyruklama davranışından hem benzerlikler hem de farklar var.

- The WebSphere MQ Explorer cannot directly administer WebSphere MQ for z/OS queue managers at versions earlier than Version 6.0.
- Küme kuyruklarına erişmeleri için, bir kümenin dışındaki kuyruk yöneticilerine el ile yapılandırma tanımlamaları eklemelisiniz.
- İki kümeyi aynı adla birleştirirseniz, bunları yeniden ayıramazsınız. Bu nedenle, tüm kümelere benzersiz bir ad verebilmek önerilir.
- Kuyruk yöneticisinde bir ileti gelirse, ancak bu iletiyi alacak bir kuyruk yoksa, ileti, ölü-mektup kuyruğuna konadır. Herhangi bir ölü harf kuyruğu yoksa, kanal başarısız olur ve yeniden dener. Ölü harf kuyruğun kullanımı, dağıtılmış kuyruğa alma ile aynıdır.
- Kalıcı iletilerin bütünlüğü sürdürülüdür. İletiler, kümelerin kullanılmasının sonucu olarak çoğaltılmaz ya da kaybolmaz.
- Kümelerin kullanılması sistem yönetimini azaltır. Kümeler, dağıtılmış kuyruklama kullanarak tasarlamayacaklarından çok daha fazla kuyruk yöneticisi ile daha büyük ağların birbirine bağlanmasını kolaylaştırır. Bir kümedeki her kuyruk yöneticisi arasında iletişimi etkinleştirmeye çalışırsanız, aşırı ağ kaynakları tüketebileceğiniz bir risk vardır.
- Kuyruk yöneticilerini ağaç yapısında sunan WebSphere MQ Explorer kullanıyorsanız, büyük kümeler için görünüm hantal olabilir.

- WebSphere MQ Explorer can administer a cluster with repository queue managers on WebSphere MQ for z/OS Version 6 or later. Aynı bir sistemde ek bir havuz göstermemeniz gerekir. For earlier versions of WebSphere MQ on z/OS, the WebSphere MQ Explorer cannot administer a cluster with repository queue managers. WebSphere MQ Gezgini 'nin denetleyebileceği bir sistemde ek bir havuz göstermeniz gerekir.
- Dağıtım listelerinin amacı, birden çok hedefe aynı iletiyi göndermek için tek bir MQPUT komutu kullanmaktır. Dağıtım listeleri WebSphere MQ for AIX, IBM i, HP-UX, Solaris, Linux ve Windows için desteklenir. Dağıtım listelerini kuyruk yöneticisi kümelerine sahip olarak kullanabilirsiniz. Bir kümede, tüm iletiler MQPUT saatinde genişletilir. Bu avantaj, ağ trafiği açısından, kümeleme dışı bir ortamda olduğu kadar büyük değildir. Dağıtım listelerinin avantajı, çok sayıda kanal ve iletim kuyruklarının el ile tanımlanması gerekmemesine neden olur.
- İş yükünüzü dengelemek için kümeleri kullanabildiğinizde, uygulamalarınızı inceler. İletilerin belirli bir kuyruk yöneticisi tarafından mı, yoksa belirli bir sırada mı işlenmesini zorunlu kılacağını belirleyin. Bu tür uygulamaların mesaj afetlerine sahip olduğu söyleniyor. Karmaşık kümelerde bunları kullanabilmeniz için uygulamalarınızı değiştirmeniz gerekebilir.
- İletilerin belirli bir hedefe gönderilmesini sağlamak için MQOPEN üzerinde MQOO_BIND_ON_OPEN seçeneğini kullanmayı seçebilirsiniz. Hedef kuyruk yöneticisi kullanılmıyorsa, kuyruk yöneticisi yeniden kullanılabilir duruma gelinceye kadar iletiler teslim edilmez. Yinelenme riski nedeniyle iletiler başka bir kuyruk yöneticisine yönlendirilmedi.
- Kuyruk yöneticisi bir küme havuzunu barıntacaksa, anasistem adını ya da IP adresini bilmeniz gerekir. You have to specify this information in the AD1 parameter when you make the CLUSSDR definition on other queue managers joining the cluster. DHCP kullanıyorsanız, bir sistemi her yeniden başlattığınızda DHCP yeni bir IP adresi ayarlayabileceği için, IP adresi değişebilir. Bu nedenle, CLUSSDR tanımlamalarında IP adresini belirtmemelisiniz. Tüm CLUSTSDR tanımlarında IP adresi yerine anasistem adı belirtilse bile, tanımlamalar yine de güvenilir olmaz. DHCP, anasisteme ilişkin DNS izin girişini yeni adresle güncellemez. Kuyruk yöneticilerini, DHCP kullanan sistemlerde tam havuz olarak aday göstermeniz gerekiyorsa, DNS dizinini güncel tutmayı garanti eden yazılımı kurun.
- Sosyal adlar kullanmayın; örneğin, VTAM sosyal kaynakları ya da Dynamic Domain Name Server (GDNS) sosyal adları, kanallarınız için bağlantı adları olarak adlandırır. Bunu yapabilirseniz, kanallarınız beklenenden farklı bir kuyruk yöneticisine bağlanabilirler.
- Bir iletiyi yalnızca yerel bir küme kuyruğundan alabilirsiniz, ancak bir kümedeki herhangi bir kuyruğa ileti koyabilirsiniz. If you open a queue to use the MQGET command, the queue manager opens the local queue.
- Basit bir WebSphere MQ kümesi ayarladıysanız, uygulamalarınızdan herhangi birini değiştirmenize gerek yoktur. Uygulama, MQOPEN çağrısında hedef kuyruğun adını verebilir ve kuyruk yöneticisinin yerini bilmek zorunda değildir. İş yükü yönetimi için bir küme ayarlarsanız, uygulamalarınızı gözden geçirmeniz ve bunları gerektiği gibi değiştirmelisiniz.
- DISPLAY CHSTATUS ve DISPLAY QSTATUS **runmqsc** komutlarını kullanarak bir kanala ya da kuyruğa ilişkin yürürlükteki izleme ve durum verilerini görüntüleyebilirsiniz. İzleme bilgileri, sistemin performansını ve durumunu göstermenize yardımcı olmak için kullanılabilir. İzleme, kuyruk yöneticisi, kuyruk ve kanal öznitelikleri tarafından denetlenir. Monitoring of auto-defined cluster-sender channels is possible with the MONACLS queue manager attribute.

İlgili kavramlar

Kümeleler

Kümeleler nasıl çalışır

“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

“Bir kümenin bileşenleri” sayfa 161

Kümeleler, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

“IBM WebSphere MQ kümelelerini yönetme” sayfa 181

IBM WebSphere MQ kümelelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

İlgili görevler

“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

“Yeni bir küme ayarlanıyor” sayfa 181

Örnek kümeyi ayarlamak için bu yönergeleri izleyin. Ayrı yönergeler, TCP/IP, LU 6.2 üzerinde ve tek bir iletim kuyruğu ya da birden çok iletim kuyruğundaki kümeyi ayarlayıp tanımlamalarını açıklar. Bir kuyruk yöneticisinden diğerine ileti göndererek, kümenin çalıştığını test edin.

Bir kümede iletişim kurulması

Teslim edilecek bir ileti olduğunda bir iletişim kanalı başlatmak için kanal başlatıcısı gerekir. Bir kanal dinleyicisi, iletiyi almak için bir kanalın diğer ucunu başlatmak için bekler.

Başlamadan önce

Bir kümedeki kuyruk yöneticileri arasında iletişim kurmak için, desteklenen iletişim protokollerinden birini kullanarak bir bağlantı yapılandırın. Desteklenen iletişim kuralları, herhangi bir platform üzerinde TCP ya da LU 6.2 ve Windows sistemlerinde NetBIOS ya da SPX. Bu yapılandırmanın bir parçası olarak, dağıtım kuyruğa alma işlemi sırasında yapmanız gereken, kanal başlatıcılarına ve kanal dinleyicilerine de ihtiyacınız vardır.

Bu görev hakkında

Sistem tanımlı başlatma kuyruğunu (SYSTEM . CHANNEL . INITQ) izlemek için tüm küme kuyruğu yöneticilerinin bir kanal başlatıcısına ihtiyacı vardır. SYSTEM . CHANNEL . INITQ , küme iletim kuyruğu da içinde olmak üzere, tüm iletim kuyruklarına ilişkin başlatma kuyruğudur.

Her kuyruk yöneticisinin bir kanal dinleyicisi olması gerekir. Bir kanal dinleyici programı, gelen ağ isteklerini bekler ve gerektiğinde uygun alıcı kanalını başlatır. Kanal dinleyicilerinin uygulanması platforma özgü, ancak bazı ortak özellikler de vardır. Tüm WebSphere MQ altyapılarında, dinleyici START LISTENER komutu kullanılarak başlatılabilir. WebSphere MQ for IBM i, Pencereler, UNIX and Linux sistemlerinde, dinleyici adını kuyruk yöneticisiyle aynı anda otomatik olarak başlatabilirsiniz. İletişimci otomatik olarak başlatmak için, LISTENER nesnesinin CONTROL özneliğini QMGR ya da STARTONLY olarak ayarlayın.

Yordam

1. Kanal başlatıcısı başlatın.

- 

IBM WebSphere MQ for Pencereler, UNIX and Linux systems

When you start a queue manager, if the queue manager attribute SCHINIT is set to MMGR, a channel initiator is automatically started. Otherwise it can be started using the **runmqsc** START CHINIT command or the **runmqchi** control command.

2. Kanal dinleyicisi başlat.

- 

Pencereler için IBM WebSphere MQ

WebSphere MQ tarafından sağlanan kanal dinleyici programını ya da işletim sistemi tarafından sağlanan olanakları kullanın.

WebSphere MQ kanalı dinleyicisini başlatmak için RUNQLSR komutunu kullanın. Örneğin:

```
RUNQLSR -t tcp -p 1414 -m QM1
```

- 

UNIX and Linux sistemlerinde IBM WebSphere MQ

Use either the channel listener program provided by WebSphere MQ, or the facilities provided by the operating system; for example, **inetd** for TCP communications.

WebSphere MQ kanalı dinleyicisini başlatmak için **runmqclsr** komutunu kullanın. Örneğin:

```
runmqclsr -t tcp -p 1414 -m QM1
```

Kanalları başlatmak için **inetd** ' u kullanmak için iki dosya yapılandırın:

- /etc/services dosyasını düzenleyin. Ayrıcalıklı kullanıcı ya da kök olarak oturum açmış olmanız gerekir. Aşağıdaki satır dosyada yoksa, aşağıdaki satırı gösterildiği gibi ekleyin:

```
MQSeries      1414/tcp      # Websphere MQ channel listener
```

Burada 1414 , IBM WebSphere MQ için gereken kapı numarasıdır. Kapı numarasını değiştirebilirsiniz, ancak bu, gönderme bitişindeki belirtilen kapı numarasıyla eşleşmelidir.

- /etc/inetd.conf dosyasını düzenleyin. Bu dosyada aşağıdaki satırı yoksa, aşağıdaki satırı gösterildiği gibi ekleyin:

```
MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta  
-m queue.manager.name
```

Burada `MQ_INSTALLATION_PATH` , WebSphere MQ ' un kurulu olduğu üst düzey dizinle değiştirilir.

The updates become active after **inetd** has reread the configuration files. Kök kullanıcı kimliğinden aşağıdaki komutları verin:

AIX üzerinde:

```
refresh -s inetd
```

HP-UX üzerinde:

```
inetd -c
```

Solaris ya da Linux üzerinde:

- Find the process ID of the **inetd** with the command:

```
ps -ef | grep inetd
```

- Uygun komutu aşağıdaki gibi çalıştırın:

- Solaris 9 ve Linux için:

```
kill -1 inetd processid
```

- Solaris 10 ya da sonraki sürümleri için:

```
inetconv
```

Kuyruk yöneticisi havuzları bilgileri ne kadar süreyle alıkoymuyor?

Kuyruk yöneticisi havuzları bilgileri 30 gün boyunca saklar. Otomatik işlem, kullanılmakta olan bilgileri verimli bir şekilde yeniler.

Bir kuyruk yöneticisi kendisi hakkında bazı bilgiler gönderdiğinde, tam ve kısmi havuz kuyruğu yöneticileri bilgileri 30 gün boyunca saklar. Örneğin, bir kuyruk yöneticisi yeni bir kuyruğun yaratılmasına duyurulduğunda, bilgi dışarı gönderilir. Bu bilgilerin süresi dolmasına engel olmak için kuyruk yöneticileri, 27 gün sonra kendi hakkındaki tüm bilgileri otomatik olarak yeniden göndermektedir. Kısmi bir havuz, 30 günlük yaşam süresi boyunca bilgi için yeni bir istek gönderirse, son 30 gün süre bitimi kalır.

Bilgilerin süresi dolduğunda, havuzdan hemen kaldırılmaz. Bunun yerine 60 günlük bir ek süre için tutulmaktadır. Ek süre içinde güncelleme alınmazsa, bilgiler kaldırılır. Ek süre, bir kuyruk yöneticisinin süre bitimi tarihinde geçici olarak hizmet dışı kalabileceği gerçemesine olanak tanır. Kuyruk yöneticisi 90 günden uzun bir süre kümeden kesilirse, kümenin bir parçası olmayı durdurur. Ancak, ağla yeniden bağlanırsa, yeniden kümenin bir parçası olur. Tüm havuzlar, diğer kuyruk yöneticilerinden gelen yeni istekleri karşılamak için süresi dolmuş bilgileri kullanmaz.

Benzer şekilde, bir kuyruk yöneticisi dolu bir havuzdan güncel bilgiler için bir istek gönderdiğinde, istek 30 gün sürer. After 27 days IBM WebSphere MQ checks the request. 27 gün içinde başvurulduysa, otomatik olarak yenilenir. Yoksa, kullanım süresi sona erer ve kuyruk yöneticisi tarafından yeniden güncellenir ve yeniden gereklidir. Süresi dolan istekler, atıl kuyruk yöneticilerinden bilgi için isteklerin birikmesini önler.

Not: Büyük kümeler için, birçok kuyruk yöneticisi aynı anda kendileriyle ilgili tüm bilgileri otomatik olarak yeniden gönderirse kesintiye yol açar. Bkz. [“Büyük bir kümede yenileme, kümenin performansını ve kullanılabilirliğini olumsuz etkileyebilir” sayfa 300.](#)

İlgili kavramlar

[“Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma” sayfa 299](#)

Bir kümeye ilişkin olarak yerel olarak tutulan tüm bilgileri atmak ve bu bilgileri kümedeki tüm havuzlardan yeniden oluşturmak için **REFRESH CLUSTER** komutunu kullanıyorsunuz. Bu komutu, istisnai koşullar dışında, kullanmanıza gerek yoktur. Bunu kullanmanız gerekiyorsa, bunu nasıl kullanmanız gerektiği konusunda dikkat edilmesi gereken noktalar vardır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuzdur.

IBM WebSphere MQ kümelerini yönetme

IBM WebSphere MQ kümelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

IBM WebSphere MQ kümelerinizi nasıl yönetebilmeye ilişkin ayrıntılar için aşağıdaki alt konulara bakın:

İlgili kavramlar

Kümeler

[Kümeler nasıl çalışır](#)

[“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158](#)

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

[“Bir kümenin bileşenleri” sayfa 161](#)

Kümeler, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

İlgili görevler

[“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156](#)

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

[“Yeni bir küme ayarlanıyor” sayfa 181](#)

Örnek kümeyi ayarlamak için bu yönergeleri izleyin. Ayrı yönergeler, TCP/IP, LU 6.2.üzerinde ve tek bir iletim kuyruğu ya da birden çok iletim kuyruğundaki kümeyi ayarlayıp tanımlamalarını açıklar. Bir kuyruk yöneticisinden diğerine ileti göndererek, kümenin çalıştığını test edin.

Yeni bir küme ayarlanıyor

Örnek kümeyi ayarlamak için bu yönergeleri izleyin. Ayrı yönergeler, TCP/IP, LU 6.2.üzerinde ve tek bir iletim kuyruğu ya da birden çok iletim kuyruğundaki kümeyi ayarlayıp tanımlamalarını açıklar. Bir kuyruk yöneticisinden diğerine ileti göndererek, kümenin çalıştığını test edin.

Başlamadan önce

- Bu yönergeleri izleyerek, bu görevin yarattığı gibi bir küme yaratmak için IBM WebSphere MQ Explorer ile birlikte verilen sihirbazlardan birini kullanabilirsiniz. Kuyruk Yöneticisi Kümeleri klasörünü farenin

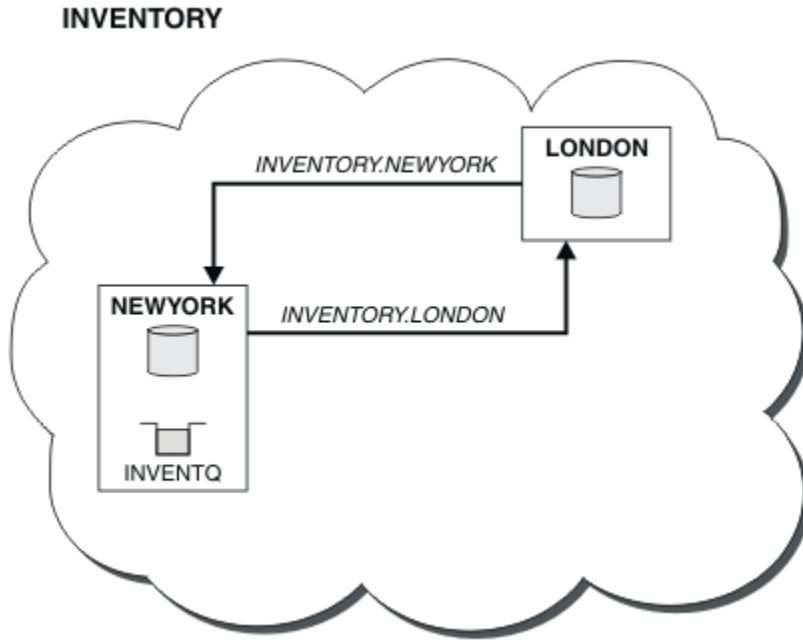
sağ düğmesiyle tıklattın ve **Yeni > Kuyruk Yöneticisi Kümesi** öğelerini seçin ve sihirbazda belirtilen yönergeleri izleyin.

- Bir küme oluşturmak için atılan adımlarla ilgili anlayışınıza yardımcı olabilecek arka plan bilgileri için bkz. “Küme kuyrukları” sayfa 162, [Kanallar](#) ve [Dinleyiciler](#).

Bu görev hakkında

Bir zincir mağaza için yeni bir IBM WebSphere MQ ağı oluşturuyorsunuz. Mağazanda iki tane şubesi var, biri Londra 'da, diğeri de New York 'ta. Her bir mağazana ilişkin veriler ve uygulamalar, ayrı kuyruk yöneticileri çalıştıran sistemler tarafından barındırılır. The two queue managers are called LONDON and NEWYORK. The inventory application runs on the system in New York, connected to queue manager NEWYORK. The application is driven by the arrival of messages on the INVENTQ queue, hosted by NEWYORK. The two queue managers, LONDON and NEWYORK, are to be linked in a cluster called INVENTORY so that they can both put messages to the INVENTQ.

Şekil 29 sayfa 182 içinde bu kümenin nasıl görüldüğünü gösterir.



Şekil 29. İki kuyruk yöneticisine sahip INVENTORY kümesi

Farklı küme iletim kuyrukları kullanarak kümedeki diğer kuyruk yöneticilerine ileti göndermek için, z/OS üzerinde olmayan her kuyruk yöneticisini yapılandırabilirsiniz.

Kümeyi ayarlamaya ilişkin yönergeler, iletim protokollerine, iletim kuyruklarına ya da platforma göre biraz değişiklik gösterir. Üç kombinasyon seçeneğiniz var. Doğrulama yordamı, tüm birleşimler için aynı kalır.

Yordam

- “Kuyruk yöneticisi başına tek bir iletim kuyruğuyla TCP/IP kullanarak bir küme kurulması” sayfa 183
- “Kuyruk yöneticisi başına birden çok iletim kuyruğu kullanarak TCP/IP ' de bir küme kurulması” sayfa 186
- “z/OSüzerinde LU 6.2 kullanan bir küme oluşturulması” sayfa 188
- “Kümenin doğrulanması” sayfa 190

Sonuçlar

Şekil 29 sayfa 182 , bu görevle INVENTORY kümesi kuruluşunu gösterir.

Clearly, INVENTORY is a small cluster. Ancak, bu kavramın bir kanıtı olarak kullanışlıdır. Bu küme hakkında anlamanız gereken önemli nokta, gelecekteki geliştirmeniz için sunduğu kapsamdır.

İlgili kavramlar

Kümeler

Kümeler nasıl çalışır

“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

“Bir kümenin bileşenleri” sayfa 161

Kümeler, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

“IBM WebSphere MQ kümelerini yönetme” sayfa 181

IBM WebSphere MQ kümelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

İlgili görevler

“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

Kuyruk yöneticisi başına tek bir iletim kuyruğuyla TCP/IP kullanarak bir küme kurulması

Başlamadan önce

- AIX, HP-UX, IBM i, Linux, Solaris, and Windows üzerinde, kuyruk yöneticisi özniteliği (**DEFCLXQ**), varsayılan değeri olan SCTQ olarak bırakılmalıdır.

Bu görev hakkında

İletim protokolü TCP/IP ' yi kullanarak AIX, HP-UX, IBM i, Linux, Solaris, and Windows üzerinde bir küme ayarlamak için bu adımları izleyin.

Yordam

1. Kümenin kuruluşuna ve adına karar verin.

İki kuyruk yöneticisini, LONDON ve NEWYORK' yi bir kümeye bağlamaya karar verdiğinizde. Yalnızca iki kuyruk yöneticisi olan bir küme, dağıtılmış kuyruğa alma kullanan bir ağ üzerinde yalnızca marjinal avantaj sağlar. Bu, başlangıç için iyi bir yoldur ve gelecekteki genişlemenin kapsamını sağlar. Mağazanız yeni dallarını açtığınızda, yeni kuyruk yöneticilerini kümeye kolayca ekleyebilirsiniz. Yeni kuyruk yöneticilerinin eklenmesi var olan ağı bozmaz; bkz. “Küme için kuyruk yöneticisi eklenmesi” sayfa 191.

Bu süre için, çalıştırdığınız tek uygulama stok uygulamasıdır. Küme adı INVENTORY.

2. Hangi kuyruk yöneticilerinin tüm havuzları tutacağına karar verin.

Herhangi bir kümede, tam havuzları tutmak için en az bir kuyruk yöneticisi ya da tercihen iki kişi aday göstermeniz gerekir. Bu örnekte, her ikisi de tam havuz tutan yalnızca iki kuyruk yöneticisi (LONDON ve NEWYORK) vardır.

- a. Geri kalan adımları herhangi bir sırada gerçekleştirebilirsiniz.
- b. Adımlarda ilerledikçe, uyarı iletileri kuyruk yöneticisi günlüğüne yazılabilir. İletiler, henüz ekleyediğiniz eksik tanımların sonudur.

Examples of the responses to the commands are shown in a box like this after each step in this task. These examples show the responses returned by WebSphere MQ for AIX. The responses vary on other platforms.

- c. Bu adımlara devam etmeden önce kuyruk yöneticilerinin başlatıldığından emin olun.
3. Havuz tanımlamalarını eklemek için kuyruk yöneticisi tanımlamalarını değiştirin.

Tam havuzu tutacak her kuyruk yöneticisinde, ALTER QMGR komutunu kullanarak ve REPOS özniteliğini belirterek, yerel kuyruk yöneticisi tanımlamasını değiştirin.

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: Websphere MQ queue manager changed.
```

Örneğin, aşağıdaki bilgileri girin:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON , tam bir havuz olarak değiştirildi.

4. Dinleyicileri tanımlayın.

Kümedeki her kuyruk yöneticisi için diğer kuyruk yöneticilerinden gelen ağ isteklerini kabul eden bir dinleyici tanımlayın. LONDON kuyruk yöneticilerindeki aşağıdaki komutu verin:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

Not: Bir dinleyici tanımladığınızda, CONNAME alanında IP adresleri kullanıyorsanız ve kapı numarası varsayılan kapı (1414) değilse, bir kapı numarası tanımlanmalıdır. Örneğin:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(1415)
```

CONTROL özniteliği, dinleyicinin kuyruk yöneticisi tarafından başlatılıp başlatılmasını ve durduğunu doğrular.

Dinleyici, tanımlandığında başlatılmaz; bu nedenle, aşağıdaki MQSC komutu ile ilk kez el ile başlatılmalıdır:

```
START LISTENER(LONDON_LS)
```

Kümedeki diğer tüm kuyruk yöneticileri için de benzer komutlar yayınlayın ve dinleyici adını her bir kuyruk için değiştirin.

Bu dinleyicileri tanımlamak için birkaç yol vardır: [Dinleyicilerinde](#) gösterildiği gibi.

5. LONDON kuyruk yöneticisi için CLUSRCVR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinde, kuyruk yöneticisinin iletileri alabileceği bir küme alıcılı kanalı tanımlayın. CLUSTVR , kuyruk yöneticisinin bağlantı adını tanımlar. Bağlantı adı, diğer kuyruk yöneticilerinin başvuruda bulunduğu havuzlarda depolanır. CLUSTER anahtar sözcüğü, kuyrukta bulunan diğer kuyruk yöneticilerinden ileti almak için kuyruk yöneticisinin kullanılabilirliğini gösterir.

Bu örnekte kanal adı INVENTORY . LONDON ve bağlantı adı (CONNAME) , kuyruk yöneticisinin bulunduğu makinenin ağ adresidir; LONDON .CHSTORE .COM ise. Ağ adresi, alfasayısal bir DNS anasistem adı olarak ya da IPv4 noktalı onlu biçimindeki bir IP adresi olarak girilebilir. Örneğin, 192 . 0 . 2 . 0 ya da IPv6 onaltılı biçimi; örneğin, 2001 : DB8 : 0204 : acff : fe97 : 2c34 : fde0 : 3485. Kapı numarası belirtilmedi, bu nedenle varsayılan kapı (1414) kullanılır.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```



```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: Websphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. NEWYORK kuyruk yöneticisi için CLUSRCVR kanalını tanımlayın.

Kanal dinleyicisi varsayılan kapıyı kullanıyorsa, genellikle 1414 ve küme z/OSüzerinde bir kuyruk yöneticisi içermiyorsa, CONNNAME' ı atlayabilirsiniz.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
```

7. LONDON kuyruk yöneticisinde CLUSSDR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinde, bir küme gönderici kanalı tanımlayın. Kuyruk yöneticisi, iletileri, küme gönderen kanalındaki tam havuz kuyruğu yöneticilerinden birine gönderir. Bu durumda, her ikisi de tam havuz tutan yalnızca iki kuyruk yöneticisi vardır. Bunların her birinde, diğer kuyruk yöneticisinde tanımlanan CLUSRCVR kanalını gösteren bir CLUSSDR tanımlaması olmalıdır. CLUSSDR tanımlamalarında verilen kanal adlarının, karşılık gelen CLUSTRVR tanımlamalarındaki kanal adlarıyla eşleşmesi gerekir. Kuyruk yöneticisi, aynı kümedeki bir küme alıcı kanalı ve bir küme gönderici kanalı için tanımlamalar varsa, kümeli gönderici kanalı başlatılır.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: Websphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

8. NEWYORK kuyruk yöneticisinde CLUSSDR kanalını tanımlayın.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. Define the cluster queue INVENTQ

NEWYORK kuyruk yöneticisinde, CLUSTER anahtar sözcüğünü belirterek INVENTQ kuyruğunu tanımlayın.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: Websphere MQ queue created.
```

CLUSTER anahtar sözcüğü, kuyruğun kümeye tanıtılmasına neden olur. Kuyruk tanımlanır tanımlanmaz, kümedeki diğer kuyruk yöneticilerine de kullanılabilir olur. Bunun için bir uzak kuyruk tanımlaması yapmadan, bu ileti için ileti gönderebilirler.

Tüm tanımlamalar tamamlandı. Tüm altyapılarda, her kuyruk yöneticisinde bir dinleyici programı başlatın. Dinleyici programı, gelen ağ isteklerini bekler ve gerektiğinde küme alıcı kanalını başlatır.

Kuyruk yöneticisi başına birden çok iletim kuyruğu kullanarak TCP/IP ' de bir küme kurulması

Bu görev hakkında

İletim protokolü TCP/IP ' yi kullanarak AIX, HP-UX, IBM i, Linux, Solaris, and Windows üzerinde bir küme ayarlamak için bu adımları izleyin. Havuz kuyruğu yöneticileri, iletileri birbirlerine ve kümedeki diğer kuyruk yöneticilerine göndermek için farklı bir küme iletim kuyruğu kullanacak şekilde yapılandırılır. Farklı iletim kuyrukları kullanmak için de küme yöneticilerini kümeye eklerseniz, görevi izleyin [“Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194](#). Ayrı küme iletim kuyrukları kullanmak için z/OS üzerinde bir kuyruk yöneticisi ayarlayamazsınız.

Yordam

1. Kümenin kuruluşuna ve adına karar verin.

İki kuyruk yöneticisini, LONDON ve NEWYORK' yi bir kümeye bağlamaya karar verdiniz. Yalnızca iki kuyruk yöneticisi olan bir küme, dağıtılmış kuyruğa alma kullanan bir ağ üzerinde yalnızca marjinal avantaj sağlar. Bu, başlangıç için iyi bir yoldur ve gelecekteki genişlemenin kapsamını sağlar. Mağazanız yeni dallarını açtığınızda, yeni kuyruk yöneticilerini kümeye kolayca ekleyebildiniz. Yeni kuyruk yöneticilerinin eklenmesi var olan ağı bozmadı; bkz. [“Küme için kuyruk yöneticisi eklenmesi” sayfa 191](#).

Bu süre için, çalıştırdığınız tek uygulama stok uygulamasıdır. Küme adı INVENTORY.

2. Hangi kuyruk yöneticilerinin tüm havuzları tutacağına karar verin.

Herhangi bir kümede, tam havuzları tutmak için en az bir kuyruk yöneticisi ya da tercihen iki kişi aday göstermeniz gerekir. Bu örnekte, her ikisi de tam havuz tutan yalnızca iki kuyruk yöneticisi (LONDON ve NEWYORK) vardır.

- a. Geri kalan adımları herhangi bir sırada gerçekleştirebilirsiniz.
- b. Adımlarda ilerledikçe, uyarı iletileri kuyruk yöneticisi günlüğüne yazılabilir. İletiler, henüz ekleyemediğiniz eksik tanımların sonudur.

Examples of the responses to the commands are shown in a box like this after each step in this task. These examples show the responses returned by WebSphere MQ for AIX. The responses vary on other platforms.

- c. Bu adımlara devam etmeden önce kuyruk yöneticilerinin başlatıldığından emin olun.

3. Havuz tanımlamalarını eklemek için kuyruk yöneticisi tanımlamalarını değiştirin.

Tam havuzu tutacak her kuyruk yöneticisinde, ALTER QMGR komutunu kullanarak ve REPOS özniteliğini belirterek, yerel kuyruk yöneticisi tanımlamasını değiştirin.

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: Websphere MQ queue manager changed.
```

Örneğin, aşağıdaki bilgileri girin:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON , tam bir havuz olarak değiştirildi.

4. Her hedef için ayrı küme iletim kuyrukları yaratmak üzere kuyruk yöneticisi tanımlamalarını değiştirin.

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Kümeye eklediğiniz her kuyruk yöneticisinde ayrı iletim kuyruklarının kullanılmasının gerekip gerekmediğini belirleyin. “Küme için kuyruk yöneticisi eklenmesi” sayfa 191 ve “Bir küme için kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194 konularına bakın.

5. Dinleyicileri tanımlayın.

Kümedeki her kuyruk yöneticisi için diğer kuyruk yöneticilerinden gelen ağ isteklerini kabul eden bir dinleyici tanımlayın. LONDON kuyruk yöneticilerindeki aşağıdaki komutu verin:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

Not: Bir dinleyici tanımladığınızda, CONNAME alanında IP adresleri kullanıyorsanız ve kapı numarası varsayılan kapı (1414) değilse, bir kapı numarası tanımlanmalıdır. Örneğin:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR) PORT(1415)
```

CONTROL özneliği, dinleyicinin kuyruk yöneticisi tarafından başlatılıp başlatılmasını ve durduğunu doğrular.

Dinleyici, tanımlandığında başlatılmaz; bu nedenle, aşağıdaki MQSC komutu ile ilk kez el ile başlatılmalıdır:

```
START LISTENER(LONDON_LS)
```

Kümedeki diğer tüm kuyruk yöneticileri için de benzer komutlar yayınlayın ve dinleyici adını her bir kuyruk için değiştirin.

Bu dinleyicileri tanımlamak için birkaç yol vardır: [Dinleyicileri](#) içinde gösterildiği gibi.

6. LONDON kuyruk yöneticisi için CLUSRCVR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinde, kuyruk yöneticisinin iletileri alabileceği bir küme alıcılı kanalı tanımlayın. CLUSTVR , kuyruk yöneticisinin bağlantı adını tanımlar. Bağlantı adı, diğer kuyruk yöneticilerinin başvuruda bulunduğu havuzlarda depolanır. CLUSTER anahtar sözcüğü, kuyrukta bulunan diğer kuyruk yöneticilerinden ileti almak için kuyruk yöneticisinin kullanılabilirliğini gösterir.

Bu örnekte kanal adı INVENTORY . LONDON ve bağlantı adı (CONNAME) , kuyruk yöneticisinin bulunduğu makinenin ağ adresidir; LONDON . CHSTORE . COM ise. Ağ adresi, alfasayısal bir DNS anasistem adı olarak ya da IPv4 noktalı onlu biçimindeki bir IP adresi olarak girilebilir. Örneğin, 192 . 0 . 2 . 0 ya da IPv6 onaltılı biçimi; örneğin, 2001 : DB8 : 0204 : acff : fe97 : 2c34 : fde0 : 3485 . Kapı numarası belirtilmedi, bu nedenle varsayılan kapı (1414) kullanılır.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: Websphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

7. NEWYORK kuyruk yöneticisi için CLUSRCVR kanalını tanımlayın.

Kanal dinleyicisi varsayılan kapıyı kullanıyorsa, genellikle 1414 ve küme z/OS üzerinde bir kuyruk yöneticisi içermiyorsa, CONNAME' ı atlayabilirsiniz.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
```

8. LONDON kuyruk yöneticisinde CLUSSDR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinde, bir küme gönderici kanalı tanımlayın. Kuyruk yöneticisi, iletileri, küme gönderen kanalındaki tam havuz kuyruğu yöneticilerinden birine gönderir. Bu durumda, her ikisi de tam havuz tutan yalnızca iki kuyruk yöneticisi vardır. Bunların her birinde, diğer kuyruk yöneticisinde tanımlanan CLUSRCVR kanalını gösteren bir CLUSSDR tanımlaması olmalıdır. CLUSSDR

tanımlamalarında verilen kanal adlarının, karşılık gelen CLUSTRVR tanımlamalarındaki kanal adlarıyla eşleşmesi gerekir. Kuyruk yöneticisi, aynı kümedeki bir küme alıcı kanalı ve bir küme gönderici kanalı için tanımlamalar varsa, kümeli gönderici kanalı başlatılır.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: Websphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

9. NEWYORK kuyruk yöneticisinde CLUSSDR kanalını tanımlayın.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

10. Define the cluster queue INVENTQ

NEWYORK kuyruk yöneticisinde, CLUSTER anahtar sözcüğünü belirterek INVENTQ kuyruğunu tanımlayın.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: Websphere MQ queue created.
```

CLUSTER anahtar sözcüğü, kuyruğun kümeye tanıtılmasına neden olur. Kuyruk tanımlanır tanımlanmaz, kümedeki diğer kuyruk yöneticilerine de kullanılabilir olur. Bunun için bir uzak kuyruk tanımlaması yapmadan, bu ileti için ileti gönderebilirler.

Tüm tanımlamalar tamamlandı. Tüm altyapılarda, her kuyruk yöneticisinde bir dinleyici programı başlatın. Dinleyici programı, gelen ağ isteklerini bekler ve gerektiğinde küme alıcı kanalını başlatır.

z/OSüzerinde LU 6.2 kullanan bir küme oluşturulması

Yordam

1. Kümenin kuruluşuna ve adına karar verin.

İki kuyruk yöneticisini, LONDON ve NEWYORK' yi bir kümeye bağlamaya karar verdiğinizde. Yalnızca iki kuyruk yöneticisi olan bir küme, dağıtılmış kuyruğa alma kullanan bir ağ üzerinde yalnızca marjinal avantaj sağlar. Bu, başlangıç için iyi bir yoldur ve gelecekteki genişlemenin kapsamını sağlar. Mağazanız yeni dallarını açtığınızda, yeni kuyruk yöneticilerini kümeye kolayca ekleyebilirsiniz. Yeni kuyruk yöneticilerinin eklenmesi var olan ağı bozmaz; bkz. [“Küme için kuyruk yöneticisi eklenmesi” sayfa 191.](#)

Bu süre için, çalıştırdığınız tek uygulama stok uygulamasıdır. Küme adı INVENTORY.

2. Hangi kuyruk yöneticilerinin tüm havuzları tutacağına karar verin.

Herhangi bir kümede, tam havuzları tutmak için en az bir kuyruk yöneticisi ya da tercihen iki kişi aday göstermeniz gerekir. Bu örnekte, her ikisi de tam havuz tutan yalnızca iki kuyruk yöneticisi (LONDON ve NEWYORK) vardır.

a. Geri kalan adımları herhangi bir sırada gerçekleştirebilirsiniz.

b. Adımlarda ilerledikçe, uyarı iletileri z/OS sistem konsolunu yazılabilir. İletiler, henüz ekleyediğiniz eksik tanımların sonudur.

c. Bu adımlara devam etmeden önce kuyruk yöneticilerinin başlatıldığından emin olun.

3. Havuz tanımlamalarını eklemek için kuyruk yöneticisi tanımlamalarını değiştirin.

Tam havuzu tutacak her kuyruk yöneticisinde, ALTER QMGR komutunu kullanarak ve REPOS özneliğini belirterek, yerel kuyruk yöneticisi tanımlamasını değiştirin.

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: Websphere MQ queue manager changed.
```

Örneğin, aşağıdaki bilgileri girin:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON , tam bir havuz olarak değiştirildi.

4. Dinleyicileri tanımlayın.

Dinleyici, tanımlandığında başlatılmaz; bu nedenle, aşağıdaki MQSC komutu ile ilk kez el ile başlatılmalıdır:

```
START LISTENER(LONDON_LS)
```

Kümedeki diğer tüm kuyruk yöneticileri için de benzer komutlar yayınlayın ve dinleyici adını her bir kuyruk için değiştirin.

5. LONDON kuyruk yöneticisi için CLUSRCVR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinde, kuyruk yöneticisinin iletileri alabileceği bir küme alıcılı kanalı tanımlayın. CLUSTVR , kuyruk yöneticisinin bağlantı adını tanımlar. Bağlantı adı, diğer kuyruk yöneticilerinin başvuruda bulunduğu havuzlarda depolanır. CLUSTER anahtar sözcüğü, kuyrukta bulunan diğer kuyruk yöneticilerinden ileti almak için kuyruk yöneticisinin kullanılabilirliğini gösterir.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
AMQ8014: Websphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. NEWYORK kuyruk yöneticisi için CLUSRCVR kanalını tanımlayın.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager NEWYORK')
```

7. LONDON kuyruk yöneticisinde CLUSSDR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinde, bir küme gönderici kanalı tanımlayın. Kuyruk yöneticisi, iletileri, küme gönderen kanalındaki tam havuz kuyruğu yöneticilerinden birine gönderir. Bu durumda, her ikisi de tam havuz tutan yalnızca iki kuyruk yöneticisi vardır. Bunların her birinde, diğer kuyruk yöneticisinde tanımlanan CLUSRCVR kanalını gösteren bir CLUSSDR tanımlaması olmalıdır. CLUSSDR tanımlamalarında verilen kanal adlarının, karşılık gelen CLUSTVR tanımlamalarındaki kanal adlarıyla eşleşmesi gerekir. Kuyruk yöneticisi, aynı kümedeki bir küme alıcı kanalı ve bir küme gönderici kanalı için tanımlamalar varsa, kümeli gönderici kanalı başlatılır.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(CPIC) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: Websphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

8. NEWYORK kuyruk yöneticisinde CLUSSDR kanalını tanımlayın.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. Define the cluster queue INVENTQ

NEWYORK kuyruk yöneticisinde, CLUSTER anahtar sözcüğünü belirterek INVENTQ kuyruğunu tanımlayın.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: Websphere MQ queue created.
```

CLUSTER anahtar sözcüğü, kuyruğun kümeye tanıtılmasına neden olur. Kuyruk tanımları tanımlanmaz, kümedeki diğer kuyruk yöneticilerine de kullanılabilir olur. Bunun için bir uzak kuyruk tanımlaması yapmadan, bu ileti için ileti gönderebilirler.

Tüm tanımlamalar tamamlandı. Tüm altyapılarda, her kuyruk yöneticisinde bir dinleyici programı başlatın. Dinleyici programı, gelen ağ isteklerini bekler ve gerektiğinde küme alıcı kanalını başlatır.

Kümenin doğrulanması

Bu görev hakkında

Kümeyi aşağıdaki yollardan birini ya da birkaçını kullanarak doğrulayabilirsiniz:

1. Küme ve kanal özniteliklerini görüntülemek için yönetim komutları çalıştırılıyor.
2. Bir küme kuyruğunda ileti göndermek ve almak için örnek programları çalıştırın.
3. Bir istek iletisini bir küme kuyruğuna göndermek ve kümeli olmayan bir yanıt kuyruğuna yanıt iletilerini yanıtlamak için kendi programlarınızı yazın.

Yordam

Kümeyi doğrulamak için DISPLAY **runmqsc** komutlarını yayınlayın.

Gördüğünüz yanıtlar, takip eden adımlardaki yanıtlar gibi olmalıdır.

1. NEWYORK kuyruk yöneticisinden **DISPLAY CLUSQMGR** komutunu çalıştırın:

```
dis clusqmgr(*)
```

```
1 : dis clusqmgr(*)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(NEWYORK) CLUSTER(INVENTORY)
CHANNEL(INVENTORY.NEWYORK)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(LONDON) CLUSTER(INVENTORY)
CHANNEL(INVENTORY.LONDON)
```

2. NEWYORK kuyruk yöneticisinden **DISPLAY CHANNEL STATUS** komutunu çalıştırın:

```
dis chstatus(*)
```

```
1 : dis chstatus(*)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.NEWYORK) XMITQ( )
CONNAME(192.0.2.0) CURRENT
CHLTYPE(CLUSRCVR) STATUS(RUNNING)
RQMNAME(LONDON)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.LONDON) XMITQ(SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON)
CONNAME(192.0.2.1) CURRENT
CHLTYPE(CLUSSDR) STATUS(RUNNING)
RQMNAME(LONDON)
```

amqsput komutunu kullanarak, iki kuyruk yöneticisi arasında ileti gönderin.

3. LONDON üzerinde **amqsput INVENTQ LONDON** komutunu çalıştırın.

Bazı iletileri yazın ve ardından boş bir satır yazın.

4. NEWYORK üzerinde **amqsget INVENTQ NEWYORK** komutunu çalıştırın.

Şimdi LONDON' ta girdiğiniz iletileri görürsünüz. 15 saniye sonra program sona eriyor.

Kendi programlarınızı kullanarak iki kuyruk yöneticisi arasında ileti gönderin.

Aşağıdaki adımlarda, LONDON , NEWYORK saatinde INVENTQ ' e bir ileti koyar ve LONDON_reply kuyruğuna yanıt alır.

5. LONDON üzerinde, küme kuyruğuna bir ileti yerleştirin.

- Define a local queue called LONDON_reply.
- MQOPEN seçeneklerini MQOO_OUTPUT olarak ayarlayın.
- Issue the MQOPEN call to open the queue INVENTQ.
- İleti açıklayıcısındaki *ReplyToQ* adını LONDON_reply olarak ayarlayın.
- İletiyi yerleştirmek için MQPUT çağrısını yayınlayın.
- İletiyi kesinleştirin.

6. NEWYORK ' da ileti kümesi kuyruğunda iletiyi alın ve yanıt kuyruğuna bir yanıt yazın.

- MQOPEN seçeneklerini MQOO_BROWSE olarak ayarlayın.
- Issue the MQOPEN call to open the queue INVENTQ.
- İletiyi INVENTQ' den almak için MQGET çağrısını yayınlayın.
- İleti tanımlayıcısından *ReplyToQ* adını alın.
- Nesne tanımlayıcısının *ObjectName* alanına *ReplyToQ* adını yazın.
- MQOPEN seçeneklerini MQOO_OUTPUT olarak ayarlayın.
- Issue the MQOPEN call to open LONDON_reply at queue manager LONDON.
- Issue the MQPUT call to put the message to LONDON_reply.

7. On LONDON receive the reply.

- MQOPEN seçeneklerini MQOO_BROWSE olarak ayarlayın.
- Issue the MQOPEN call to open the queue LONDON_reply.
- Issue the MQGET call to get the message from LONDON_reply.

Küme için kuyruk yöneticisi eklenmesi

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Küme kuyruklarına ve konulara ilişkin iletiler tek küme iletim kuyruğu SYSTEM.CLUSTER.TRANSMIT.QUEUE kullanılarak aktarılır.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Yeni bir küme ayarlanıyor” sayfa 181’inde açıklandığı şekilde ayarlanır. İki kuyruk yöneticisi (LONDON ve NEWYORK) içerir; bu da, her ikisinin de tam havuz tutması sağlar.
- Kuyruk yöneticisi PARIS birincil kuruluş tarafından iyelidir. If it is not, you must run the **setmqenv** command to set up the command environment for the installation that PARIS belongs to.
- Tüm üç sistem arasında TCP bağlantılığı vardır ve kuyruk yöneticisi, kuyruk yöneticisinin denetimi altında başlayan bir TCP dinleyicisiyle yapılandırılır.

Bu görev hakkında

1. Paris 'te bir zincir mağazasının yeni bir şubesi hazırlanıyor ve kümeden PARIS adlı bir kuyruk yöneticisi eklemek istiyorsunuz.
2. Kuyruk yöneticisi PARIS , INVENTORY kuyruğuna ileti koyarak, sistemde New York 'ta çalışan uygulamaya döküm güncellemeleri gönderir.

Bir kümeye kuyruk yöneticisi eklemek için aşağıdaki adımları izleyin.

Yordam

1. PARIS hangi tam havuzunun ilk önce başvurmaya karar vereceğine karar verin.

Bir kümedeki her kuyruk yöneticisinin, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Tam havuzdaki kümeyle ilgili bilgileri toplar ve bu nedenle kendi kısmi havuzunu oluşturur. Havuzlardan birini tam havuz olarak seçin. Kümeye yeni bir kuyruk yöneticisi eklendiği anda, diğer havuzu da hemen öğrenir. Kuyruk yöneticisindeki değişikliklerle ilgili bilgiler doğrudan iki havuza gönderilir. Bu örnekte, yalnızca coğrafi nedenlerden dolayı PARIS kuyruk yöneticisine LONDONbağlanmasını sağlar.

Not: Kuyruk yöneticisi PARIS başlatıldıktan sonra, kalan adımları herhangi bir sırada gerçekleştirin.

2. Define a CLUSRCVR channel on queue manager PARIS.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcı kanalı tanımlamalıdır. PARISüzerinde şunları tanımlayın:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

The cluster-receiver channel advertises the availability of the queue manager to receive messages from other queue managers in the cluster INVENTORY. Bir gönderme sonunun küme alıcı kanalı INVENTORY . PARIS' e gönderilmesi için diğer kuyruk yöneticilerine tanım yapmaya gerek yoktur. Diğer tanımlar gerektiğinde otomatik olarak yapılır.

3. Define a CLUSSDR channel on queue manager PARIS.

Bir kümedeki her kuyruk yöneticisinin, ilk tam havuzlarına ileti gönderebileceği bir küme gönderici kanalı tanımlamalıdır.

On PARIS, make the following definition for a channel called INVENTORY . LONDON to the queue manager with the network address LONDON . CHSTORE . COM.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. İsteğe bağlı: Bu kuyruk yöneticisi bir kümeye yeniden katılıyorsa, bazı ek adımları tamamlayın.

a) Aynı kümeden önceden kaldırılmış bir kümeye kuyruk yöneticisi ekliyorsanız, bunun artık bir küme üyesi olarak gösterildiğini doğrulayın. Aşağıdaki ek adımları tamamlamazsa, aşağıdaki adımları tamamlayın:

i) Eklemeli olduğunuz kuyruk yöneticinde **REFRESH CLUSTER** komutunu verin. Bu adım, küme kanallarını durdurur ve yerel küme önbelleğinizi, kümenin geri kalan sınırları içinde güncel olacak şekilde güvenli bir dizi sıra numarası kümesinden verir.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

Not: Büyük kümeler için, **REFRESH CLUSTER** komutunun kullanımı devam ederken kümeyi kesintiye uğratabilir ve bundan sonra 27 gün aralıklarla küme nesnelere, ilgili tüm kuyruk yöneticilerine otomatik olarak durum güncellemeleri gönderdiğinde, bu işlem yine 27 gün aralıklarla kesintiye uğrayabilir. Bkz. [Büyük bir kümede yenilenme, kümenin performansını ve kullanılabilirliğini etkileyebilir.](#)

ii) CLUSSDR kanalını yeniden başlatın (örneğin, [START CHANNEL](#) komutunu kullanarak).

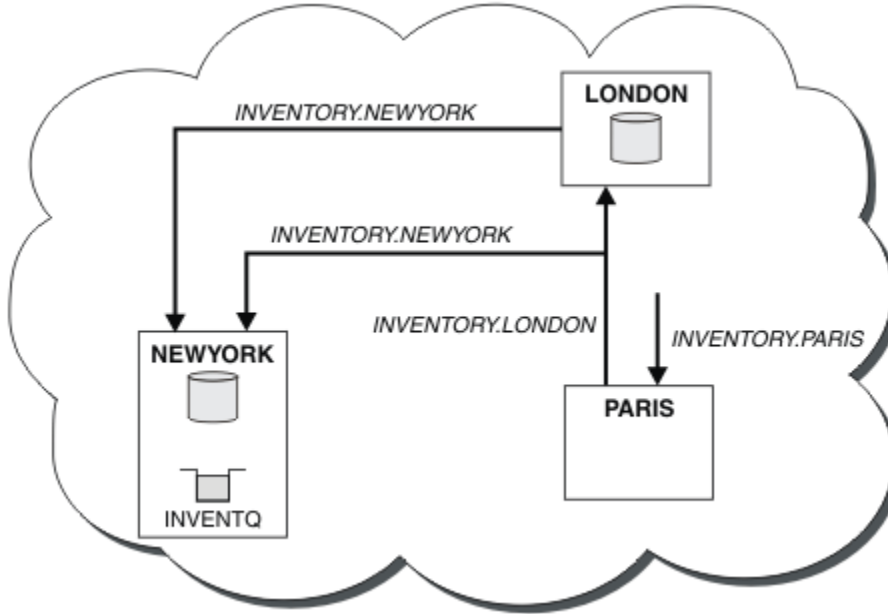
iii) CLUSTRVR kanalını yeniden başlatın.

b) Küme bir yayınlama/abone olma kümesiye ve yeniden birleştirme kuyruk yöneticisinin abonelikleri varsa, yetkili sunucu aboneliklerinin küme genelinde doğru bir şekilde uyumlaştırılmasını sağlamak için aşağıdaki komutu verin:

```
REFRESH QMGR TYPE(PROXYSUB)
```

Sonuçlar

Aşağıdaki şekil, bu görev tarafından ayarlanan kümeyi göstermektedir.



Şekil 30. Üç kuyruk yöneticisine sahip INVENTORY kümesi

Yalnızca iki tanımlama, bir CLUSTRVR tanımlaması ve bir CLUSSDR tanımlaması yaparak, PARIS kuyruk yöneticisini kümeye ekledik.

Now the PARIS queue manager learns, from the full repository at LONDON, that the INVENTQ queue is hosted by queue manager NEWYORK. When an application hosted by the system in Paris tries to put messages to the INVENTQ, PARIS automatically defines a cluster-sender channel to connect to the cluster-receiver channel INVENTORY . NEWYORK. Hedef kuyruk yöneticisi olarak kuyruk yöneticisi adı belirtildiğinde ve bir yanıtlama kuyruğu sağlandığı zaman, uygulama yanıt alabilir.

Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

Başlamadan önce

- Kuyruk yöneticisi, z/OS'dışındaki bir altyapıda tanımlıdır.
- Kuyruk yöneticisi herhangi bir kümenin üyesi değil.
- Küme var; bu kuyruk yöneticisinin doğrudan bağlanabileceği ve havuz kullanılabilir durumda olduğu tam bir havuz var. Kümeyi oluşturmaya ilişkin adımlar için bkz. [“Yeni bir küme ayarlanıyor” sayfa 181](#).

Bu görev hakkında

Bu görev, küme iletilerini tek bir iletim kuyruğunda yer alan bir kümeye kuyruk yöneticisi eklediğiniz [“Küme için kuyruk yöneticisi eklenmesi” sayfa 191](#) 'e bir alternatiftir.

Bu görevde, her bir küme gönderici kanalı için otomatik olarak ayrı küme iletim kuyrukları oluşturan bir kümeye kuyruk yöneticisi ekliyorsunuz.

Kuyrukların tanımlarının sayısını küçük tutmak için, varsayılan değer tek bir iletim kuyruğu kullanmaktadır. Farklı kuyruk yöneticilerine ve farklı kümelere giden trafiği izlemek istiyorsanız, ayrı iletim kuyrukları kullanılması avantajlıdır. Yalıtma ya da performans hedeflerine ulaşmak için trafiği farklı hedeflere ayırmak da isteyebilirsiniz.

Yordam

1. Varsayılan küme kanalı iletim kuyruğu tipini değiştirin.

Alter the queue manager PARIS:

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Kuyruk yöneticisi, kuyruk yöneticisine ileti göndermek için bir küme gönderici kanalı yarattığınız her zaman, bir küme iletim kuyruğu yaratır. İletim kuyruğu yalnızca bu küme gönderici kanalı tarafından kullanılır. İletim kuyruğu kalıcı-dinamik. It is created from the model queue, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE, with the name SYSTEM.CLUSTER.TRANSMIT.ChannelName.



Uyarı: Ürünün önceki bir sürümünden yükseltile bir kuyruk yöneticisiyle özel olarak ayrılmış SYSTEM.CLUSTER.TRANSMIT.QUEUES kullanıyorsanız, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE ' in [SHARE/NOSHARE](#) seçeneğinin **SHARE** olarak ayarlandığından emin olun.

2. PARIS hangi tam havuzunun ilk önce başvurmaya karar vereceğine karar verin.

Bir kümedeki her kuyruk yöneticisinin, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Tam havuzdaki kümeyle ilgili bilgileri toplar ve bu nedenle kendi kısmi havuzunu oluşturur. Havuzlardan birini tam havuz olarak seçin. Kümeye yeni bir kuyruk yöneticisi eklendiği anda, diğer havuzu da hemen öğrenir. Kuyruk yöneticisindeki değişikliklerle ilgili bilgiler doğrudan iki havuza gönderilir. Bu örnekte, yalnızca coğrafi nedenlerden dolayı PARIS kuyruk yöneticisine LONDONbağlanmanızı sağlar.

Not: Kuyruk yöneticisi PARIS başlatıldıktan sonra, kalan adımları herhangi bir sırada gerçekleştirin.

3. Define a CLUSRCVR channel on queue manager PARIS.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcı kanalı tanımlamalıdır. PARISüzerinde şunları tanımlayın:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)  
CONNNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)  
DESCR('Cluster-receiver channel for queue manager PARIS')
```

The cluster-receiver channel advertises the availability of the queue manager to receive messages from other queue managers in the cluster INVENTORY. Bir gönderme sonunun küme alıcı kanalı INVENTORY . PARIS' e gönderilmesi için diğer kuyruk yöneticilerine tanım yapmaya gerek yoktur. Diğer tanımlar gerektiğinde otomatik olarak yapılır.

4. Define a CLUSSDR channel on queue manager PARIS.

Bir kümedeki her kuyruk yöneticisinin, ilk tam havuzlarına ileti gönderebileceği bir küme gönderici kanalı tanımlamalıdır.

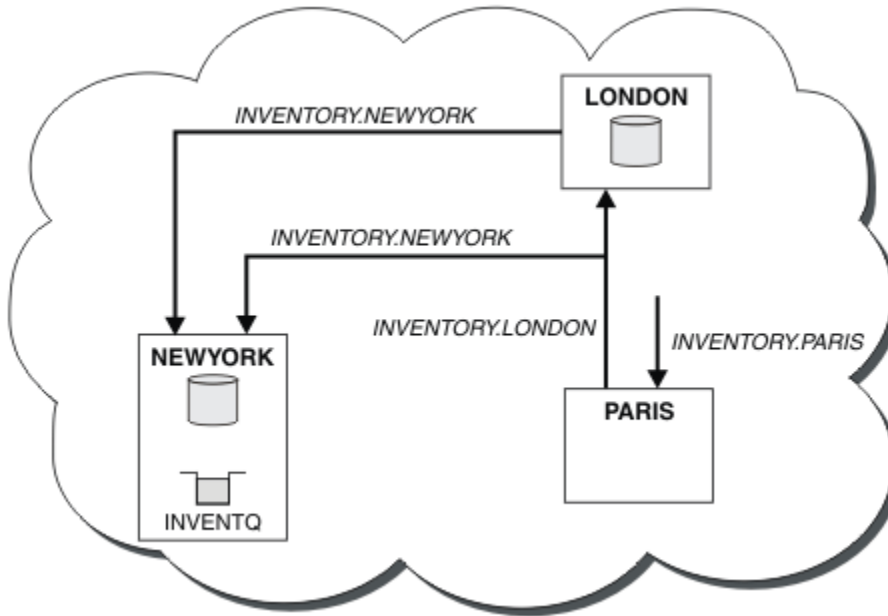
On PARIS, make the following definition for a channel called INVENTORY . LONDON to the queue manager with the network address LONDON . CHSTORE . COM.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

The queue manager automatically creates the permanent dynamic cluster transmission queue SYSTEM . CLUSTER . TRANSMIT . INVENTORY . LONDON from the model queue SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. Bu, iletim kuyruğunun CLCHNAME özneliğini INVENTORY . LONDONdeğerine ayarlar.

Sonuçlar

Aşağıdaki şekil, bu görev tarafından ayarlanan kümeyi göstermektedir.



Şekil 31. Üç kuyruk yöneticisine sahip INVENTORY kümesi

Yalnızca iki tanımlama, bir CLUSTRVR tanımlaması ve bir CLUSSDR tanımlaması yaparak, PARIS kuyruk yöneticisini kümeye ekledik.

Now the PARIS queue manager learns, from the full repository at LONDON, that the INVENTQ queue is hosted by queue manager NEWYORK. When an application hosted by the system in Paris tries to put messages to the INVENTQ, PARIS automatically defines a cluster-sender channel to connect to the cluster-receiver channel INVENTORY . NEWYORK. Hedef kuyruk yöneticisi olarak kuyruk yöneticisi adı belirtildiğinde ve bir yanıtlama kuyruğu sağlandığı zaman, uygulama yanıt alabilir.

Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtım için uzak kuyruk tanımlaması eklenmesi

Ağ geçidi kuyruk yöneticisini kullanan çıkışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelendi bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

Başlamadan önce

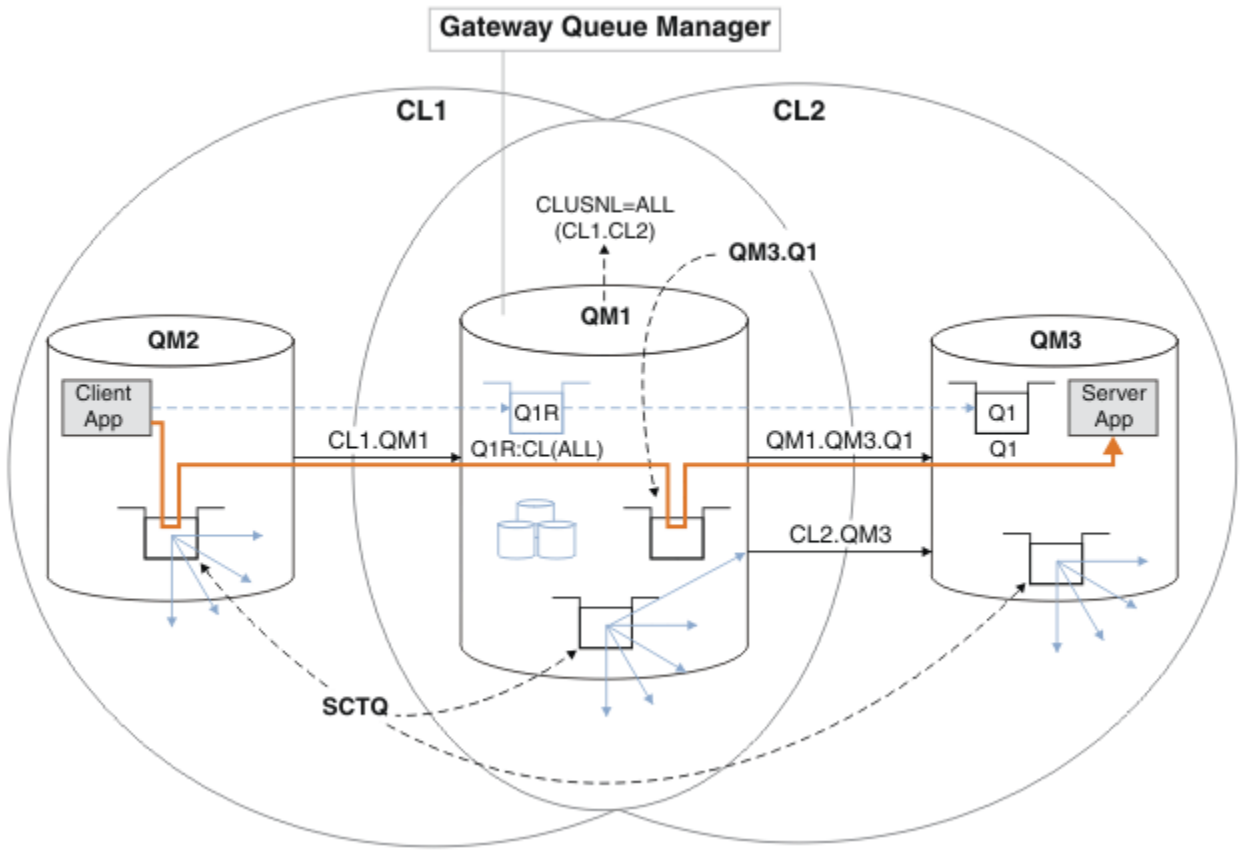
Construct the overlapping clusters shown in Şekil 37 sayfa 214 in [“Ağ geçidi kuyruk yöneticisiyle çıkışan iki küme oluşturma” sayfa 213](#) by following the steps in that task.

Bu görev hakkında

The solution uses distributed queueing to separate the messages for the Server App application from other message traffic on the gateway queue manager. İletileri farklı bir iletim kuyruğuna ve farklı bir kanala yönlendirmek için, QM1 üzerinde kümelendi bir uzak kuyruk tanımlaması tanımlamanız gerekir. Uzak kuyruk tanımlamasının, iletileri yalnızca QM3 üzerindeki Q1 için saklayan belirli iletim kuyruğuna bir başvuru içermesi gerekir. Şekil 32 sayfa 197 içinde, küme kuyruğu diğer adı Q1A uzak kuyruk tanımı Q1Rve iletim kuyruğu ve gönderen kanalı eklenerek tamamlanır.

Bu çözümde, ortak SYSTEM . CLUSTER . TRANSMIT . QUEUE kullanılarak herhangi bir yanıt iletisi döndürülür.

Bu çözümün avantajı, aynı küme içinde aynı kuyruk yöneticisinde birden çok hedef kuyruğun trafiği ayrı ayrı olması. Çözümün dezavantajı, farklı kuyruk yöneticilerindeki birden çok Q1 kopyası arasında küme iş yükü dengelemesini kullanamayamandır. Bu dezavantajı aşmak için bkz. [“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme iletim kuyruğu ekleme” sayfa 199](#). Ayrıca, anahtarı bir iletim kuyruğundan diğerine de yönetmeniz gerekir.



Şekil 32. İstemci sunucusu uygulaması göbekte konuşlandırıldı ve uzak kuyruk tanımlamalarını kullanarak küme mimarisini konuştu

Yordam

1. Ağ geçidi kuyruk yöneticisinden Q1 için ileti trafiğini ayırmak için bir kanal yaratın.
 - a) Ağ geçidi yöneticisi QM1' de hedef kuyruk yöneticisine (QM3) bir gönderen kanalı yaratın.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(SDR) CONNAME(QM3HostName(1413)) XMITQ(QM3.Q1) REPLACE
```

- b) Create a receiver channel on the target queue manager, QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(RCVR) REPLACE
```

2. Create a transmission queue on the gateway queue manager for message traffic to Q1

```
DEFINE QLOCAL(QM3.Q1) USAGE(XMITQ) REPLACE
START CHANNEL(QM1.QM3.Q1)
```

İletim kuyruğuyla ilişkili kanalı başlatma, iletim kuyruğunu kanalla ilişkilendirir. Kanal, aktarım kuyruğu kanalla ilişkilendirildikten sonra otomatik olarak başlatılır.

3. Ağ geçidi kuyruk yöneticisiyle ilgili olarak Q1 için kümelenmiş kuyruk diğer adı tanımlamasını kümelenmiş bir uzak kuyruk tanımlamasıyla tamamla.

```
DEFINE QREMOTE CLUSNL(ALL) RNAME(Q1) RQMNAME(QM3) XMITQ(QM3.Q1) REPLACE
```

Sonraki adım

Test the configuration by sending a message to Q1 on QM3 from QM2 using the clustered queue remote definition Q1R on the gateway queue manager QM1.

1. Run the sample program **amqspu**t on QM2 to put a message.

```
C:\IBM\MQ>amqspu t Q1R QM2
Sample AMQSPUT0 start
target queue is Q1R
Sample request message from QM2 to Q1 using Q1R
```

```
Sample AMQSPUT0 end
```

2. Run the sample program **amqsget** to get the message from Q1 on QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1R>
no more messages
Sample AMQSGET0 end
```

İlgili kavramlar

[“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157](#)

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, SYSTEM . CLUSTER . TRANSMIT . QUEUE' a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanıtlara karşı denetlenir.

[“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278](#)

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

İlgili görevler

[“Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194](#)

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

[“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213](#)

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

[“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelenecek bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

[“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219](#)

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenecek bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanız için bir yol sağlar.

[“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

[“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'dışındaki altyapılar için geçerlidir.

Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme iletim kuyruğu ekleme

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra, ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

Başlamadan önce

1. The gateway queue manager must be on Version 7.5, or later, and on a platform other than z/OS.
2. Construct the overlapping clusters shown in Şekil 37 sayfa 214 in “Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213 by following the steps in that task.

Bu görev hakkında

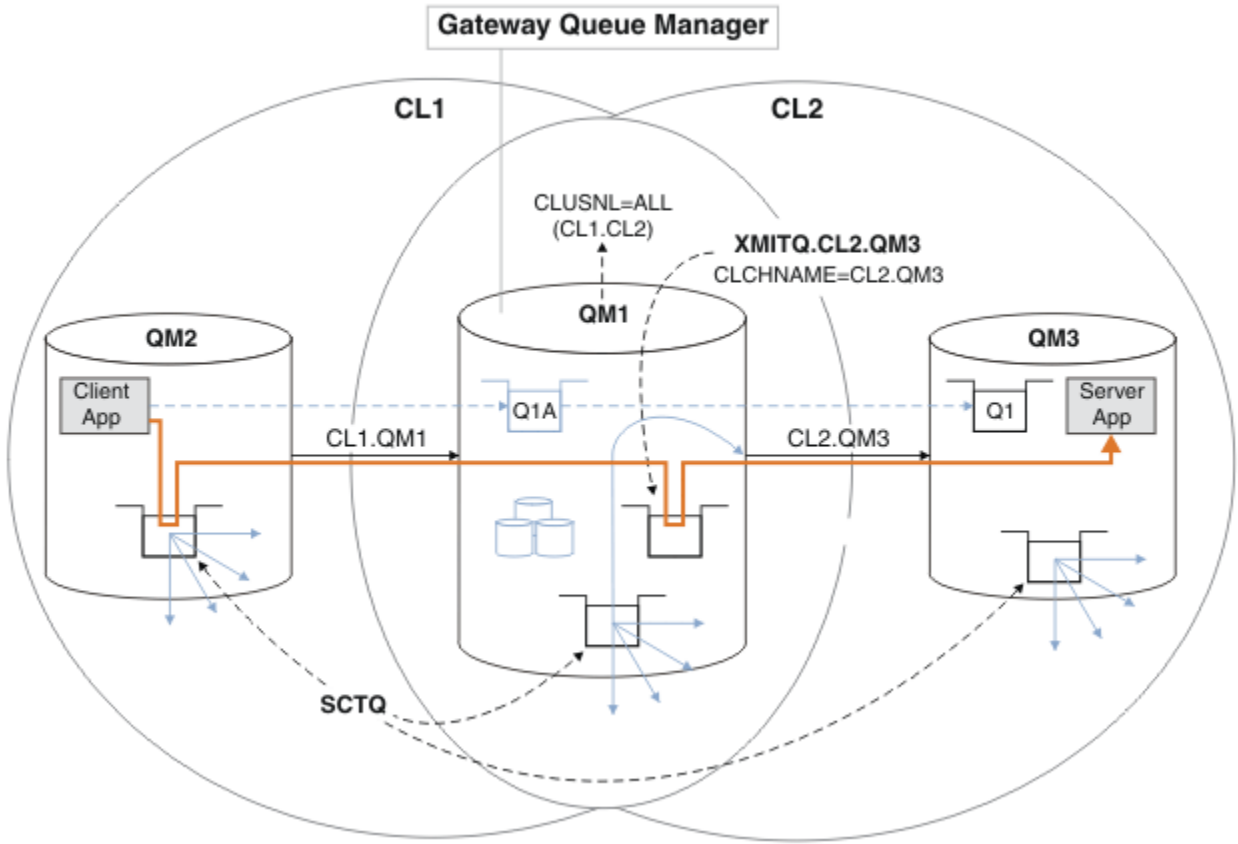
Ağ geçidi kuyruk yöneticisinde QM1, bir iletim kuyruğu ekleyin ve CLCHNAMEkuyruk özneliğini ayarlayın. Set CLCHNAME to the name of the cluster-receiver channel on QM3; see Şekil 33 sayfa 200.

This solution has a number of advantages over the solution described in “Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtım için uzak kuyruk tanımlaması eklenmesi” sayfa 196:

- Daha az tanımlama gerektirir.
- It supports workload balancing between multiple copies of the target queue, Q1, on different queue managers in the same cluster, CL2.
- Kanal yeniden başlatıldığında, kanal yeniden başlatıldığında, ağ geçidi kuyruk yöneticisi otomatik olarak yeni yapılandırmaya geçiş yapar.
- Ağ geçidi kuyruk yöneticisi, iletileri aldığı sırayla iletmeye devam eder. It does so, even if the switch takes place with messages for the queue Q1 at QM3 still on SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Şekil 33 sayfa 200 içindeki küme ileti trafiğini yalıtım için kullanılan yapılandırma, “Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtım için uzak kuyruk tanımlaması eklenmesi” sayfa 196 içindeki uzak kuyrukları kullanan yapılandırma kadar trafik yalıtılması ile sonuçlanmaz. If the queue manager QM3 in CL2 is hosting a number of different cluster queues and server applications, all those queues share the cluster channel, CL2 . QM3, connecting QM1 to QM3. The additional flows are illustrated in Şekil 33 sayfa 200 by the gray arrow representing potential cluster message traffic from the SYSTEM . CLUSTER . TRANSMIT . QUEUE to the cluster-sender channel CL2 . QM3.

Çözüm, kuyruk yöneticisini belirli bir kümede tek bir küme kuyruğunu barındırması için sınırlamanızı sağlar. Kuyruk yöneticisi zaten bir dizi küme kuyruğu barındırıyorsa, bu kısıtlamayı karşılamak için başka bir kuyruk yöneticisi oluşturmalı ya da başka bir küme oluşturmalısınız; bkz. “Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202.



Şekil 33. İstemci sunucu uygulaması, ek bir küme iletim kuyruğu kullanarak göbeğe konuşlandırıldı ve mimari konuştu.

Yordam

1. Create an additional cluster transmission queue for the cluster-sender channel CL2 .QM3 on the gateway queue manager, QM1.

```
*... on QM1
DEFINE QLOCAL(XMITQ.CL2.QM3) USAGE(XMITQ) CLCHNAME(CL2.QM3)
```

2. İletim kuyruğunu kullanmaya geçin, XMITQ .CL2 .QM3.

- a) Stop the cluster-sender channel CL2 .QM3.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

Yanıt, komutun kabul edildiğinden emin olun:

AMQ8019: Stop WebSphere MQ channel accepted.

- b) CL2 .QM3 kanalının durdurulduğunu doğrulayın.

Kanal durdurulamazsa, **STOP CHANNEL** komutunu **FORCE** seçeneğiyle yeniden çalıştırabilirsiniz. Kanal durdurulamazsa ve kanalı uyumlulaştırmak için diğer kuyruk yöneticisini yeniden başlatamazsınız; **FORCE** seçeneğini ayarlamak için bir örnek de olur.

```
*... On QM1
start
```

Yanıt, kanal durumunun bir özettir


```
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3)           CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3)              STATUS (STOPPED)
SUBSTATE (MQGET)           XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

c) Kanalı başlatın, CL2.QM3.

```
*... On QM1
START CHANNEL (CL2.QM3)
```

Yanıt, komutun kabul edildiğinden emin olun:

```
AMQ8018: Start WebSphere MQ channel accepted.
```

d) Kanala bakın.

```
*... On QM1
DISPLAY CHSTATUS (CL2.QM3)
```

Yanıt, kanal durumunun bir özetidir:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3)           CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3)              STATUS (RUNNING)
SUBSTATE (MQGET)           XMITQ (XMITQ.CL2.QM3)
```

e) İletim kuyruğunun anahtarlamalı olarak değiştirildiğini denetleyin.

Monitor the gateway queue manager error log for the message "AMQ7341 The transmission queue for channel CL2.QM3 is XMITQ.CL2.QM3".

Sonraki adım

Test the separate transmission queue by sending a message from QM2 to Q1 on QM3 using the queue alias definition Q1A

1. Bir ileti koymak için QM2 üzerinde **amqspout** örnek programını çalıştırın.

```
C:\IBM\MQ>amqspout Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

2. İletiyi şu adresten almak için **amqsget** örnek programını çalıştırın: Q1 on QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

İlgili kavramlar

[“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157](#)

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, SYSTEM.CLUSTER.TRANSMIT.QUEUE' a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanımlara karşı denetlenir.

[“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278](#)

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir

kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

“Küme iletim kuyrukları ve küme-gönderici kanalları” sayfa 168

Kümelenmiş kuyruk yöneticileri arasındaki iletiler, küme iletim kuyruklarında saklanır ve küme gönderici kanalları tarafından iletilir.

İlgili görevler

“Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelenmiş bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanız için bir yol sağlar.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'deki altyapılar için geçerlidir.

Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

Başlamadan önce

Görevdeki adımlar, Şekil 33 sayfa 200 içindeki yapılandırmayı değiştirmek için yazılır.

1. The gateway queue manager must be on Version 7.5, or later, and on a platform other than z/OS.

2. Construct the overlapping clusters shown in Şekil 37 sayfa 214 in “Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213 by following the steps in that task.
3. Do the steps in Şekil 33 sayfa 200in “Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme iletim kuyruğu ekleme” sayfa 199 to create the solution without the additional cluster. Bu görevi, bu görevdeki adımlar için bir temel olarak kullanın.

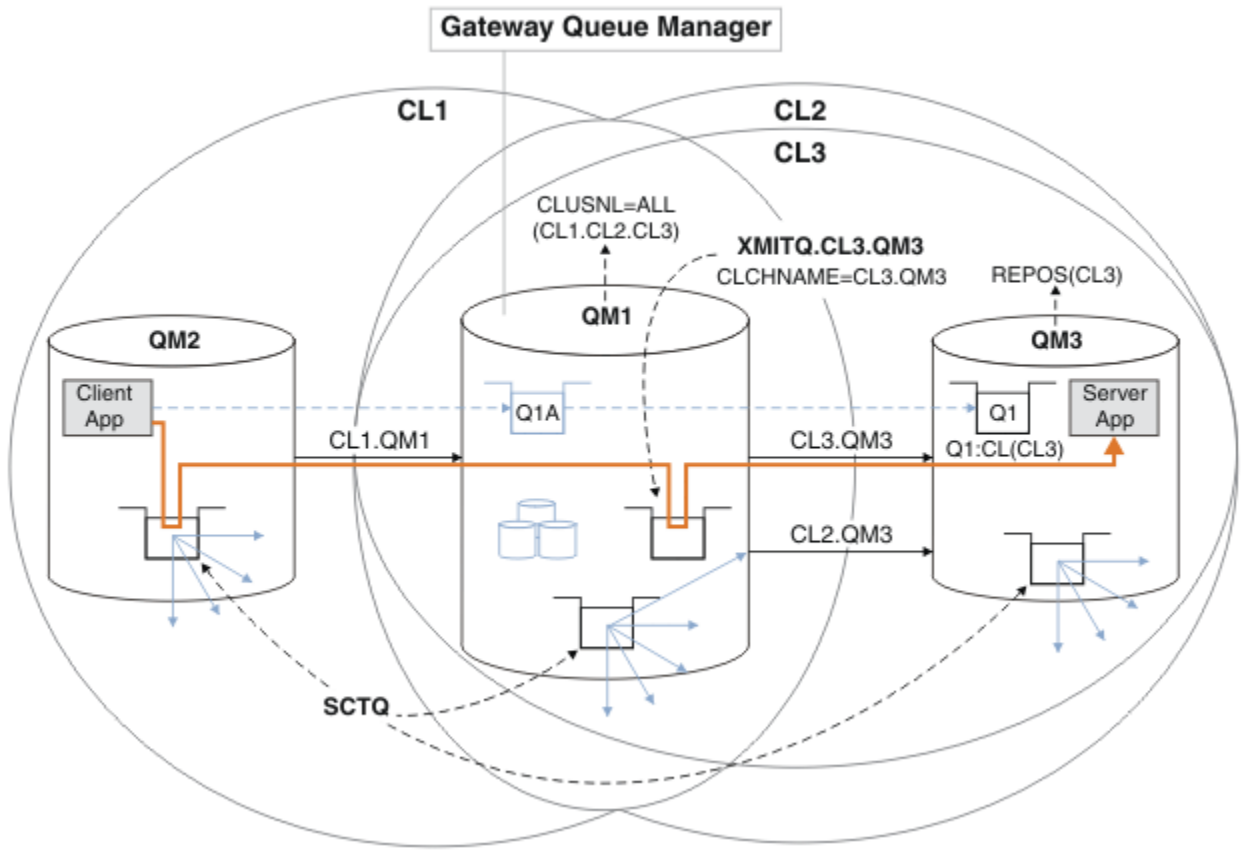
Bu görev hakkında

The solution to isolating message traffic to a single application in “Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme iletim kuyruğu ekleme” sayfa 199 works if the target cluster queue is the only cluster queue on a queue manager. Eğer değilse, iki seçeneğiniz vardır. Kuyruğu farklı bir kuyruk yöneticisine taşıyın ya da kuyruk yöneticisinde bulunan diğer küme kuyruklarından kuyruğu yalıtım bir küme yaratın.

Bu görev, hedef kuyruğun yalıtılması için bir küme ekleme adımları boyunca size yol göstermenizi sağlar. Küme yalnızca bu amaç için eklenir. Uygulamada, kümeleri ve küme adlandırma şemalarını tasarlar, belirli uygulamaları sistematik olarak yalıtmanın görevine yaklaşımları. Her kuyruk için bir küme eklenmesi için her defasında bir küme eklenmesi, yönetilecek birçok küme ile sona erebilir. In this task, you change the configuration in “Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme iletim kuyruğu ekleme” sayfa 199 by adding a cluster CL3 to isolate Q1 on QM3. Uygulamalar, değişiklik boyunca çalışmaya devam eder.

Yeni ve değiştirilen tanımlamalar Şekil 34 sayfa 204 içinde vurgulanır. Değişikliklerin özeti şu şekildedir: Bir küme yaratın; başka bir deyişle, yeni bir tam küme havuzu da yaratmalısınız. Örnekte, QM3 , CL3 için tam havuzlardan biri haline getirilmektedir. Ağ geçidi kuyruk yöneticisini yeni kümeye eklemek için, QM1 için küme-gönderici ve küme alıcı kanalları yaratın. Change the definition of Q1 to switch it to CL3. Ağ geçidi kuyruk yöneticisinde küme adı listesini değiştirin ve yeni küme kanalını kullanmak için bir küme iletim kuyruğu ekleyin. Son olarak, Q1A kuyruk diğer adını yeni küme adı listesine geçin.

IBM WebSphere MQ cannot transfer messages from the transmission queue XMITQ . CL2 . QM3 that you added in “Ağ geçidi kuyruk yöneticisinden gönderilen küme iletisi trafiğini yalıtım için bir küme iletim kuyruğu ekleme” sayfa 199 to the new transmission queue XMITQ . CL3 . QM3, automatically. İleti, yalnızca her iki iletim kuyruğu da aynı küme gönderici kanalı tarafından hizmet sunulursa otomatik olarak iletileri aktarabilir. Bunun yerine, anahtar el ile gerçekleştirilmenin bir yolu açıklanır, bu da sizin için uygun olabilir. Aktarma işlemi tamamlandığında, QM3 üzerindeki diğer CL2 küme kuyrukları için varsayılan küme iletim kuyruğunu kullanma seçeneğine geri çevirme seçeneğiniz vardır. Ya da XMITQ . CL2 . QM3 seçeneğini kullanmaya devam edebilirsiniz. Varsayılan bir küme iletim kuyruğuna geri dönmeye karar verirseniz, ağ geçidi kuyruk yöneticisi anahtar sizin için otomatik olarak yönetir.



Şekil 34. Aynı kuyruk yöneticisinde bulunan küme kuyruklarından birine giden ağ geçidi kuyruk yöneticisinde ileti trafiğini ayırmak için ek bir küme kullanılması

Yordam

1. Alter the queue managers QM3 and QM5 to make them repositories for both CL2 and CL3.

Kuyruk yöneticisi birden çok küme üyesi yapmak için, üyesi olduğu kümeleri tanımlamak için bir küme adı listesi kullanılmalıdır.

```
*... On QM3 and QM5
DEFINE NAMELIST(CL23) NAMES(CL2, CL3) REPLACE
ALTER QMGR REPOS(' ') REPOSNL(CL23)
```

2. Define the channels between the queue managers QM3 and QM5 for CL3.

```
*... On QM3
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSRCVR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE

*... On QM5
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
```

3. Ağ geçidi kuyruk yöneticisini CL3' e ekleyin.

Add the gateway queue manager by adding QM1 to CL3 as a partial repository. Küme göndereni ve küme alıcı kanalları QM1' e ekleyerek kısmi bir havuz yaratın.

Ayrıca, ağ geçidi kuyruk yöneticisine bağlı tüm kümelerin ad listesine CL3 ekleyin.

```
*... On QM1
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL3) REPLACE
ALTER NAMELIST(ALL) NAMES(CL1, CL2, CL3)
```

4. Add a cluster transmission queue to the gateway queue manager, QM1, for messages going to CL3 on QM3.

Başlangıçta, iletim kuyruklarını değiştirmeye hazır oluncaya kadar, küme gönderen kanalının iletim kuyruğundan aktarma iletilerini aktarmasını durdurun.

```
*... On QM1
DEFINE QLOCAL(XMITQ.CL3.QM3) USAGE(XMITQ) CLCHNAME(CL3.QM3) GET(DISABLED) REPLACE
```

5. Drain messages from the existing cluster transmission queue XMITQ.CL2.QM3.

Bu alt yordam, Q1 içindeki iletilerin, ağ geçidi kuyruk yöneticisine vardıkları sırayla eşleştirmek için kullanılması amaçlanır. Kümeler, ileti sıralaması tam olarak garanti edilmez, ancak büyük olasılıkla. Garantili ileti sıralaması gerekliyse, uygulamaların ileti sırasını belirlemesi gerekir; bkz. [Kuyruktan iletilerin alındığı sıralama](#).

- a) Change the target queue Q1 on QM3 from CL2 to CL3.

```
*... On QM3
ALTER QLOCAL(Q1) CLUSTER(CL3)
```

- b) İletiler teslim edilmeye başlayıncaya kadar XMITQ.CL3.QM3 programını izleyin.

Messages start to be delivered to XMITQ.CL3.QM3 when the switch of Q1 to CL3 is propagated to the gateway queue manager.

```
*... On QM1
DISPLAY QUEUE(XMITQ.CL3.QM3) CURDEPTH
```

- c) Monitor XMITQ.CL2.QM3 until it has no messages waiting to be delivered to Q1 on QM3.

Not: XMITQ.CL2.QM3, QM3 'daki diğer kuyruklar için iletileri saklarken, derinlik sıfıra gidemeyebileceği için CL2' in üyeleri olan iletiler depolanabilir.

```
*... On QM1
DISPLAY QUEUE(XMITQ.CL2.QM3) CURDEPTH
```

- d) Yeni küme iletim kuyruğundan alma işlemini geçerli kıl XMITQ.CL3.QM3

```
*... On QM1
ALTER QLOCAL(XMITQ.CL3.QM3) GET(ENABLED)
```

6. Eski küme iletim kuyruğunu (XMITQ.CL2.QM3) artık gerekmiyorsa kaldırın.

Messages for cluster queues in CL2 on QM3 revert to using the default cluster transmission queue on the gateway queue manager, QM1. The default cluster transmission queue is either SYSTEM.CLUSTER.TRANSMIT.QUEUE, or SYSTEM.CLUSTER.TRANSMIT.CL2.QM3. Which one depends on whether the value of the queue manager attribute **DEFCLXQ** on QM1 is SCTQ or KANAL. The queue manager transfers messages from XMITQ.CL2.QM3 automatically when the cluster-sender channel CL2.QM3 next starts.

- a) İletim kuyruğunu (XMITQ.CL2.QM3), küme iletim kuyruğundan olağan iletim kuyruğuna (olağan iletim kuyruğu) olarak değiştirin.

Bu işlem, herhangi bir küme gönderici kanalına sahip iletim kuyruğunun ilişkilendirmesini keser. Yanıt olarak, IBM WebSphere MQ, küme gönderici kanalı bir sonraki başlatıldığında XMITQ.CL2.QM3 iletilerini varsayılan küme iletim kuyruğuna otomatik olarak aktarır. O zamana kadar, QM3 üzerinde CL2 için iletiler XMITQ.CL2.QM3 üzerine yerleştirmeye devam eder.

```
*... On QM1
ALTER QLOCAL(XMITQ.CL2.QM3) CLCHNAME(' ')
```

- b) Stop the cluster-sender channel CL2.QM3.

Küme gönderici kanalının durdurulması ve yeniden başlatılması, iletilerin XMITQ.CL2.QM3 ' den varsayılan küme iletim kuyruğuna aktarılmasını başlatır. Genellikle, aktarma işlemini başlatmak için kanalı el ile durdurup başlatmanız. Kanal, bağlantı kesme aralığının süre bitimini sona erdirdikten sonra yeniden başlatılırsa, aktarım otomatik olarak başlatılır.

```
*... On QM1
STOP CHANNEL (CL2.QM3)
```

Yanıt, komutun kabul edildiğinden emin olun:

```
AMQ8019: Stop WebSphere MQ channel accepted.
```

c) CL2.QM3 kanalının durdurulduğunu doğrulayın.

Kanal durdurulamazsa, **STOP CHANNEL** komutunu FORCE seçeneğiyle yeniden çalıştırabilirsiniz. Kanal durdurulamazsa ve kanalı uyumlulaştırmak için diğer kuyruk yöneticisini yeniden başlatamazsınız; FORCE seçeneğini ayarlamak için bir örnek de olur.

```
*... On QM1
DISPLAY CHSTATUS (CL2.QM3)
```

Yanıt, kanal durumunun bir özettir

```
AMQ8417: Display Channel Status details.
```

CHANNEL (CL2.QM3)	CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413))	CURRENT
RQMNAME (QM3)	STATUS (STOPPED)
SUBSTATE (MQGET)	XMITQ (XMITQ.CL2.QM3)

d) Kanalı başlatın, CL2.QM3.

```
*... On QM1
START CHANNEL (CL2.QM3)
```

Yanıt, komutun kabul edildiğinden emin olun:

```
AMQ8018: Start WebSphere MQ channel accepted.
```

e) Kanala bakın.

```
*... On QM1
DISPLAY CHSTATUS (CL2.QM3)
```

Yanıt, kanal durumunun bir özettir:

```
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3)             STATUS (RUNNING)
SUBSTATE (MQGET)          XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE | CL2.QM3)
```

f) Monitor the gateway queue manager error log for the message "AMQ7341 CL2.QM3 kanalına ilişkin iletim kuyruğu SYSTEM.CLUSTER.TRANSMIT.QUEUE | CL2.QM3 olur."

g) Küme iletim kuyruğunu silin, XMITQ.CL2.QM3.

```
*... On QM1
DELETE QLOCAL (XMITQ.CL2.QM3)
```

Sonraki adım

Test the separately clustered queue by sending a message from QM2 to Q1 on QM3 using the queue alias definition Q1A

1. Bir ileti koymak için QM2 üzerinde **amqspu**t örnek programını çalıştırın.

```
C:\IBM\MQ>amqspu t Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

2. İletiyi şu adresten almak için **amqsget** örnek programını çalıştırın: Q1 on QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

İlgili kavramlar

“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, SYSTEM . CLUSTER . TRANSMIT . QUEUE' a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanıtlara karşı denetlenir.

“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

“Küme iletim kuyrukları ve küme-gönderici kanalları” sayfa 168

Kümelenmiş kuyruk yöneticileri arasındaki iletiler, küme iletim kuyruklarında saklanır ve küme gönderici kanalları tarafından iletilir.

İlgili görevler

“Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelenmiş bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanız için bir yol sağlar.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra, ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtım için ek bir küme kullanır.

“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'dışındaki altyapılar için geçerlidir.

DHCP kullanarak bir kümeye kuyruk yöneticisi eklenmesi

DHCP kullanarak bir kümeye kuyruk yöneticisi ekleyin. Görev, bir CLUSRCVR tanımlamasında AD1 değerini atlamayı gösterir.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Görev, iki özel özelliği gösterir:

- CLUSRCVR tanımındaki AD1 değerini atlamanın yeteneği.
- CLUSSDR tanımı üzerinde +QMNAME+ kullanma yeteneği.

Neither feature is provided on z/OS.

Senaryo:

- INVENTORY kümesi, “Yeni bir küme ayarlanıyor” sayfa 181 içinde açıklandığı şekilde ayarlanmıştır. İki kuyruk yöneticisi (LONDON ve NEWYORK) içerir; bu da, her ikisinin de tam havuz tutması sağlar.
- Paris 'te bir zincir mağazasının yeni bir şubesi hazırlanıyor ve kümeden PARIS adlı bir kuyruk yöneticisi eklemek istiyorsunuz.
- Kuyruk yöneticisi PARIS , INVENTQ kuyruğuna ileti koyarak, sistemde New York 'ta çalışan uygulamaya döküm güncellemeleri gönderir.
- Tüm üç sistem arasında ağ bağlantısı vardır.
- Ağ protokolü TCP 'dir.
- PARIS kuyruk yöneticisi sistemi, DHCP ' yi kullanır. Bu, IP adreslerinin sistem yeniden başlatıldığında değiştirebileceği anlamına gelir.
- The channels between the PARIS and LONDON systems are named according to a defined naming convention. Kural, LONDON üzerinde tam havuz kuyruk yöneticisinin kuyruk yöneticisi adını kullanır.
- PARIS kuyruk yöneticisinin denetimcileri, LONDON havuzundaki kuyruk yöneticisinin adı hakkında bilgi içermiyor. LONDON havuzundaki kuyruk yöneticisinin adı değiştirilebilir.

Bu görev hakkında

DHCP ' yi kullanarak bir kümeye kuyruk yöneticisi eklemek için aşağıdaki adımları izleyin.

Yordam

1. PARIS hangi tam havuzunun ilk önce başvurmasına karar vereceğine karar verin.

Bir kümedeki her kuyruk yöneticisinin, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Tam havuzdaki kümeyle ilgili bilgileri toplar ve bu nedenle kendi kısmi havuzunu oluşturur. Havuzlardan birini tam havuz olarak seçin. Kümeye yeni bir kuyruk yöneticisi eklendiği anda, diğer havuzu da hemen öğrenir. Kuyruk yöneticisindeki değişikliklerle ilgili bilgiler doğrudan iki havuza gönderilir. In this example we choose to link PARIS to the queue manager LONDON, purely for geographical reasons.

Not: Kuyruk yöneticisi PARIS başlatıldıktan sonra, kalan adımları herhangi bir sırada gerçekleştirin.

2. Define a CLUSRCVR channel on queue manager PARIS.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcı kanalı tanımlamaları gerekir. PARIS üzerinde şunları tanımlayın:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR)
TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

The cluster-receiver channel advertises the availability of the queue manager to receive messages from other queue managers in the cluster INVENTORY. Küme alıcılı kanalda CONNAME belirtmenize gerek yoktur. You can request IBM WebSphere MQ to find out the connection name from the system, either by omitting AD1, or by specifying CONNAME(' '). IBM WebSphere MQ, sistemin geçerli IP adresini kullanarak CONNAME değerini oluşturur; bkz. CONNAME. Bir gönderme sonunun küme alıcı kanalı INVENTORY.PARIS'e gönderilmesi için diğer kuyruk yöneticilerine tanım yapmaya gerek yoktur. Diğer tanımlar gerektiğinde otomatik olarak yapılır.

3. Define a CLUSSDR channel on queue manager PARIS.

Bir kümedeki her kuyruk yöneticisinin, ilk tam havuzlarına ileti gönderebileceği bir küme gönderici kanalı tanımlanması gerekir. On PARIS, make the following definition for a channel called INVENTORY.+QMNAME+ to the queue manager with the network address LONDON.CHSTORE.COM.

```
DEFINE CHANNEL(INVENTORY.+QMNAME+) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. İsteğe bağlı: Bu kuyruk yöneticisi bir kümeye yeniden katılıyorsa, bazı ek adımları tamamlayın.

a) Aynı kümeden önceden kaldırılmış bir kümeye kuyruk yöneticisi ekliyorsanız, bunun artık bir küme üyesi olarak gösterildiğini doğrulayın. Aşağıdaki ek adımları tamamlamazsa, aşağıdaki adımları tamamlayın:

i) Eklemeli olduğunuz kuyruk yöneticinde **REFRESH CLUSTER** komutunu verin. Bu adım, küme kanallarını durdurur ve yerel küme önbelleğinizi, kümenin geri kalan sınırları içinde güncel olacak şekilde güvenli bir dizi sıra numarası kümesinden verir.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

Not: Büyük kümeler için, **REFRESH CLUSTER** komutunun kullanımı devam ederken kümeyi kesintiye uğratabilir ve bundan sonra 27 gün aralıklarla küme nesnelere, ilgili tüm kuyruk yöneticilerine otomatik olarak durum güncellemeleri gönderdiğinde, bu işlem yine 27 gün aralıklarla kesintiye uğrayabilir. Bkz. [Büyük bir kümede yenilenme, kümenin performansını ve kullanılabilirliğini etkileyebilir.](#)

ii) CLUSSDR kanalını yeniden başlatın (örneğin, [START CHANNEL](#) komutunu kullanarak).

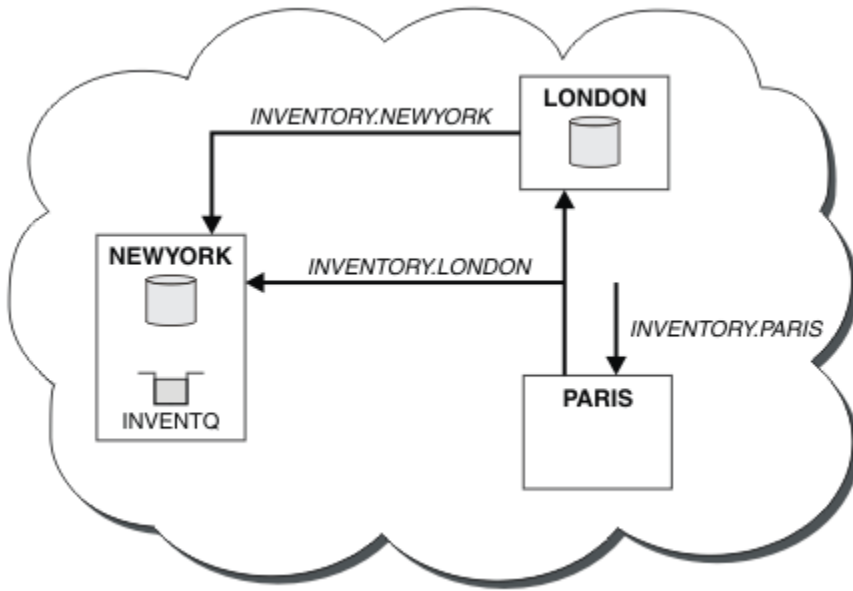
iii) CLUSTVR kanalını yeniden başlatın.

b) Küme bir yayınlama/abone olma kümesiye ve yeniden birleştirme kuyruk yöneticisinin abonelikleri varsa, yetkili sunucu aboneliklerinin küme genelinde doğru bir şekilde uyumlaştırılmasını sağlamak için aşağıdaki komutu verin:

```
REFRESH QMGR TYPE(PROXYSUB)
```

Sonuçlar

Bu görev tarafından ayarlanan küme, [“Küme için kuyruk yöneticisi eklenmesi” sayfa 191](#) ile aynı şekilde ayarlanır:



Şekil 35. Üç kuyruk yöneticisine sahip INVENTORY kümesi

By making only two definitions, a CLUSRCVR definition and a CLUSSDR definition, we have added the queue manager PARIS to the cluster.

PARIS kuyruk yöneticisinde, +QMNAME+ dizgisini içeren CLUSSDR başlatılır. On the LONDON system IBM WebSphere MQ resolves the +QMNAME+ to the queue manager name (LONDON). IBM WebSphere MQ then matches the definition for a channel called INVENTORY . LONDON to the corresponding CLUSRCVR definition.

WebSphere MQ , çözümlenen kanal adını PARIS kuyruk yöneticisine geri gönderir. At PARIS, the CLUSSDR channel definition for the channel called INVENTORY . +QMNAME+ is replaced by an internally generated CLUSSDR definition for INVENTORY . LONDON. Bu tanım çözümlenen kanal adını içerir; ancak, yaptığınız +QMNAME+ tanımlamalarıyla aynı adı alır. Küme havuzları, yeni çözümlenen kanal adı ile kanal tanımlamasıyla birlikte güncel olarak da getirilir.

Not:

1. +QMNAME+ adıyla oluşturulan kanal, hemen devre dışı duruma gelir. Veri aktarmaya hiç alışık değildir.
2. Kanal çıkışları, kanal adının bir başlatma ve sonraki bir başlatma arasında değişmesini görebilir.

Now the PARIS queue manager learns, from the repository at LONDON, that the INVENTQ queue is hosted by queue manager NEWYORK. When an application hosted by the system in Paris tries to put messages to the INVENTQ, PARIS automatically defines a cluster-sender channel to connect to the cluster-receiver channel INVENTORY . NEWYORK. Hedef kuyruk yöneticisi olarak kuyruk yöneticisi adı belirtildiğinde ve bir yanıtla kuyruğu sağlandığı zaman, uygulama yanıt alabilir.

İlgili başvurular

[KANAL TANIMLA](#)

Kuyruk barındıran bir kuyruk yöneticisi eklenmesi

Başka bir INVENTQ kuyruğunu barındırmak için kümeye başka bir kuyruk yöneticisi ekleyin. İstekler, her kuyruk yöneticisindeki kuyruklara dönüşümlü olarak gönderilir. Var olan INVENTQ anasisteminde hiçbir değişiklik yapılmaması gerekmez.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Küme için kuyruk yöneticisi eklenmesi” sayfa 191’inde açıklandığı şekilde ayarlanmıştır. Bu, üç kuyruk yöneticisi içerir; LONDON ve NEWYORK her ikisi de tam havuzları barındırıyor, PARIS kısmi bir havuz barındırıyor. Döküm uygulaması, NEWYORK kuyruk yöneticisine bağlı olarak, sistemde New York 'ta çalışır. The application is driven by the arrival of messages on the INVENTQ queue.
- Toronto 'da yeni bir mağaza hazırlanıyor. Ek kapasite sağlamak için Toronto 'daki sistemde bulunan envanter uygulamasını ve New York 'u da çalıştırmak isteyebilirsiniz.
- Tüm dört sistem arasında ağ bağlantısı vardır.
- Ağ protokolü TCP 'dir.

Not: The queue manager TORONTO contains only a partial repository. Bir kümeye tam havuz kuyruk yöneticisi eklemek istiyorsanız, “Tüm havuzu başka bir kuyruk yöneticisine taşıma” sayfa 226 dosyasına bakın.

Bu görev hakkında

Kuyruğu barındıran bir kuyruk yöneticisi eklemek için aşağıdaki adımları izleyin.

Yordam

1. TORONTO hangi tam havuzunun ilk önce başvurmayaya karar vereceğine karar verin.

Bir kümedeki her kuyruk yöneticisinin, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Tam havuzdaki kümeyle ilgili bilgileri toplar ve bu nedenle kendi kısmi havuzunu oluşturur. Hangi havuzu seçtiğiniz belli bir öneme sahip değildir. Bu örnekte, NEWYORKöğesini seçiyoruz. Yeni kuyruk yöneticisi, her iki havuzla iletişim kurduğu kümeye yeni bir kuyruk yöneticisi katıldıktan sonra.

2. CLUSTRVR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcı kanalı tanımlamaları gerekir. TORONTO üzerinde bir CLUSTRVR kanalı tanımlayın:

```
DEFINE CHANNEL(INVENTORY.TORONTO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(TORONTO.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for TORONTO')
```

TORONTO kuyruk yöneticisi, küme alıcı kanalını kullanarak INVENTORY kümesindeki diğer kuyruk yöneticilerinden ileti almak için kullanılabilirliğini bildirir.

3. Define a CLUSSDR channel on queue manager TORONTO.

Bir kümedeki her kuyruk yöneticisinin, ilk tam havuzuna ileti gönderebileceği bir küme gönderici kanalı tanımlanması gerekir. Bu durumda NEWYORKöğesini seçin. TORONTO ' in aşağıdaki tanıma gereksinimi vardır:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from TORONTO to repository at NEWYORK')
```

4. İsteğe bağlı: Bu kuyruk yöneticisi bir kümeye yeniden katılıyorsa, bazı ek adımları tamamlayın.

- a) Aynı kümeden önceden kaldırılmış bir kümeye kuyruk yöneticisi ekliyorsanız, bunun artık bir küme üyesi olarak gösterildiğini doğrulayın. Aşağıdaki ek adımları tamamlamazsa, aşağıdaki adımları tamamlayın:

- i) Eklemeli olduğunuz kuyruk yöneticinde **REFRESH CLUSTER** komutunu verin. Bu adım, küme kanallarını durdurur ve yerel küme önbelleğinizi, kümenin geri kalan sınırları içinde güncel olacak şekilde güvenli bir dizi sıra numarası kümesinden verir.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

Not: Büyük kümeler için, **REFRESH CLUSTER** komutunun kullanımı devam ederken kümeyi kesintiye uğratabilir ve bundan sonra 27 gün aralıklarla küme nesnelere, ilgili tüm kuyruk yöneticilerine otomatik olarak durum güncellemeleri gönderdiğinde, bu işlem yine 27 gün aralıklarla kesintiye uğrayabilir. Bkz. [Büyük bir kümede yenilenme, kümenin performansını ve kullanılabilirliğini etkileyebilir.](#)

- ii) CLUSSDR kanalını yeniden başlatın (örneğin, START CHANNEL komutunu kullanarak).

- iii) CLUSTRVR kanalını yeniden başlatın.

- b) Küme bir yayınlama/abone olma kümesiye ve yeniden birleştirme kuyruk yöneticisinin abonelikleri varsa, yetkili sunucu aboneliklerinin küme genelinde doğru bir şekilde uyumlaştırılmasını sağlamak için aşağıdaki komutu verin:

```
REFRESH QMGR TYPE(PROXYSUB)
```

5. İleti zenginlikleri için envanter uygulamasını gözden geçirin.

Devam etmeden önce, döküm uygulamasının iletilerin işlenmesi sırasında herhangi bir bağımlılıkları olmadığından emin olun ve uygulamayı Toronto 'da sisteme kurun.

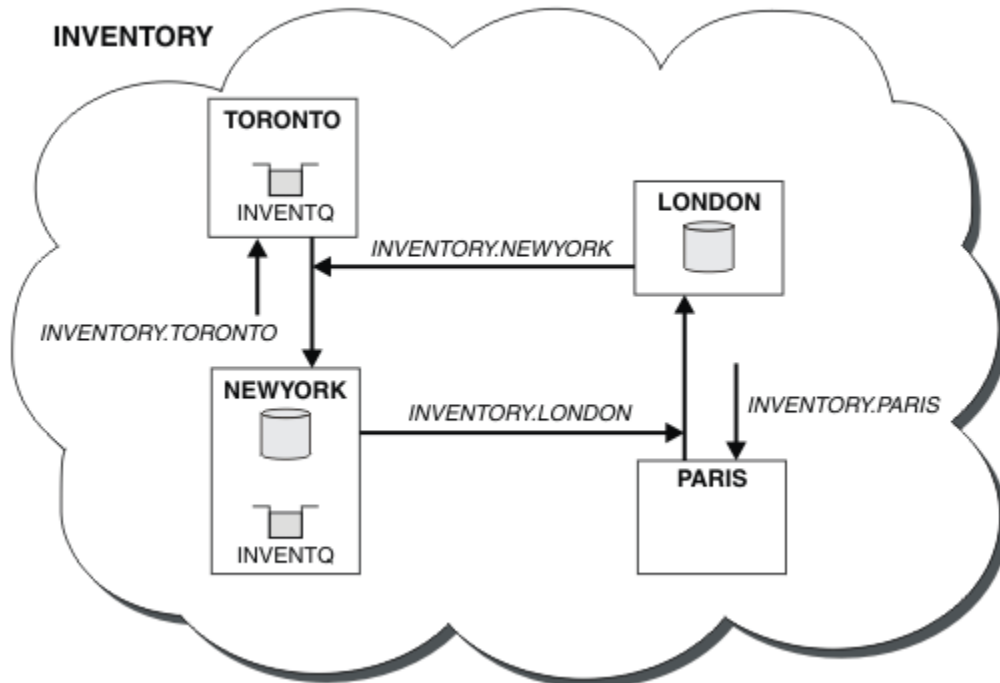
6. Define the cluster queue INVENTQ.

The INVENTQ queue, which is already hosted by the NEWYORK queue manager, is also to be hosted by TORONTO. Bunu TORONTO kuyruk yöneticisine aşağıdaki şekilde tanımlayın:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

Sonuçlar

Şekil 36 sayfa 212 , bu görev tarafından ayarlanan INVENTORY kümesini gösterir.



Şekil 36. Dört kuyruk yöneticisine sahip INVENTORY kümesi

INVENTQ kuyruğu ve döküm uygulaması, kümedeki iki kuyruk yöneticisinde artık barındırılır. Bu, onların kullanılabilirliğini artırır, ileti verimini hızlandırır ve iş yükünün iki kuyruk yöneticisi arasında dağıtılmasını sağlar. Messages put to INVENTQ by either TORONTO or NEWYORK are handled by the instance on the local queue manager whenever possible. LONDON ya da PARIS tarafından gönderilen iletiler, iş yükünün dengeli olması için dönüşümlü olarak TORONTO ya da NEWYORK' e yönlendirilir.

This modification to the cluster was accomplished without you having to alter the definitions on queue managers NEWYORK, LONDON, and PARIS. The full repositories in these queue managers are updated automatically with the information they need to be able to send messages to INVENTQ at TORONTO. NEWYORK ya da TORONTO kuyruk yöneticilerinden biri kullanılamaz duruma gelirse, döküm uygulaması çalışmaya devam eder ve yeterli kapasiteye sahiptir. Her iki yerde de barındırılırsa, döküm uygulamasının doğru çalışabilmesi gerekir.

Bu görevin sonucundan da görebileceğiniz gibi, aynı uygulamanın birden çok kuyruk yöneticisine sahip olması da olabilir. Dağılım iş yüküne eşit olarak kümeleyebilirsiniz.

Bir uygulama her iki konumdaki kayıtları işleyemeyebilir. Örneğin, LONDON ve NEWYORK'ta çalışan bir müşteri hesabı sorgusu ve güncelleme uygulaması eklemeye karar verdiğiniz varsayın. Bir hesap kaydı yalnızca bir yerde tutulabilir. Bir veri bölümlenme tekniğini kullanarak, isteklerin dağılımını denetleyebilmeye karar verebilirsiniz. Kayıtların dağılımını bölebilirsiniz. Örneğin, kayıtların yarısını (örneğin, 00000-49999 hesap numaraları gibi), LONDON' ta düzenleyebilirsiniz. Diğer yarısı, 50000-99999 aralığında, NEWYORK'ta tutulur. Daha sonra, tüm iletilerde hesap alanını incelemek için bir küme iş yükü çıkış programı yazabilir ve iletileri uygun kuyruk yöneticisine yönlerebilirsiniz.

Sonraki adım

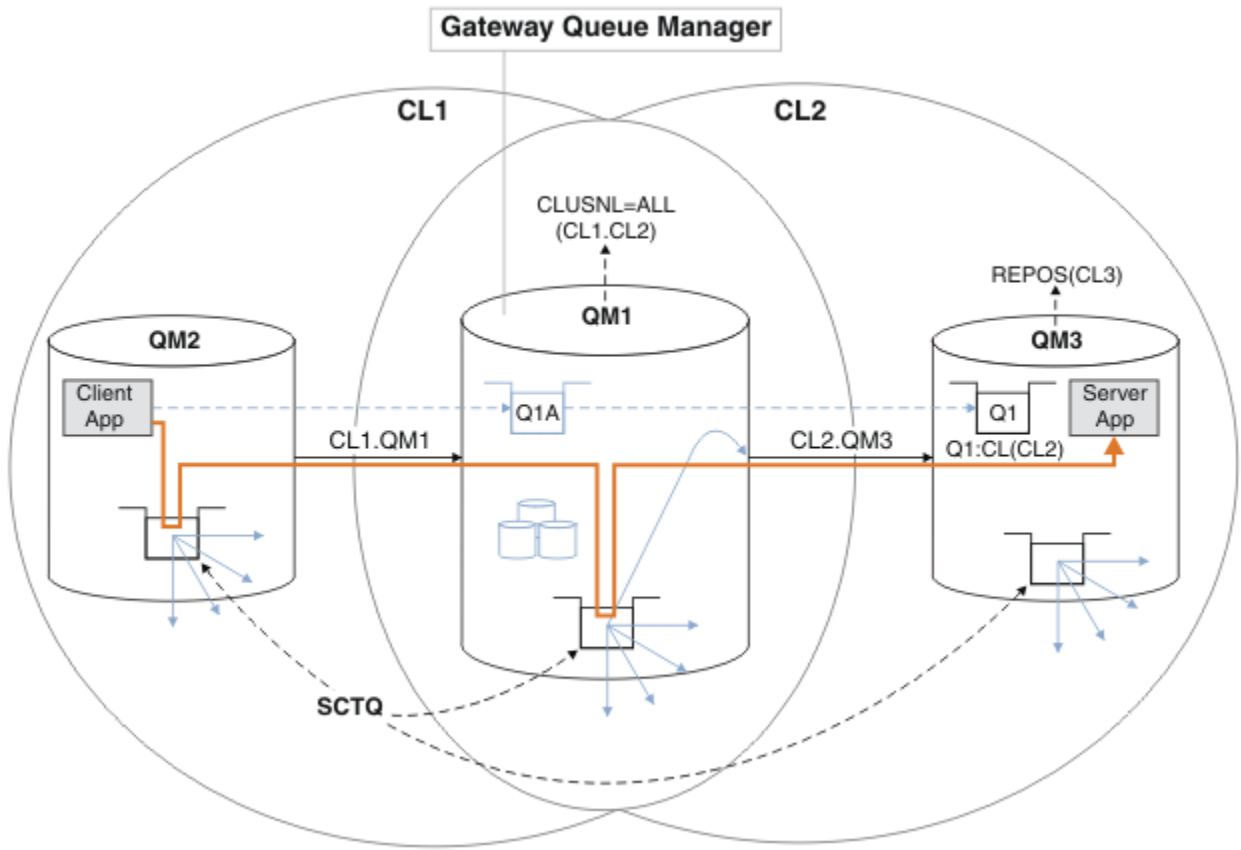
Şimdi tüm tanımları tamamladığınızı, henüz yapmadıysanız, kanal başlatıcıyı IBM WebSphere MQ for z/OS' ta başlatmadıysanız. On all platforms, start a listener program on queue manager TORONTO. Dinleyici programı, gelen ağ isteklerini bekler ve gerektiğinde küme alıcı kanalını başlatır.

Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

Bu görev hakkında

Küme ileti trafiğinin yalıtılmasını göstermek için kullanılan örnek küme yapılandırması Şekil 37 sayfa 214'te gösterilir. Örnek, "Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı" sayfa 278'te açıklanmıştır.



Şekil 37. IBM WebSphere MQ kümelerini kullanarak merkeze konuşlandırılan ve mimariye konuşlandırılmış istemci-sunucu uygulaması

Örneği mümkün olduğunca az sayıda oluşturmak için adımların sayısını yapmak için yapılandırma gerçekçi değil, basit tutulur. Bu örnek, iki ayrı kuruluş tarafından oluşturulan iki kümenin bütünleştirilmesini temsil edebilir. Daha gerçekçi bir senaryo için bkz. “Kümelleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281.

Kümeleri oluşturmak için adımları izleyin. Kümeler, istemci uygulamasından sunucu uygulamasına ileti trafiğinin yalıtılmasına ilişkin aşağıdaki örneklerde kullanılır.

Bu yönergelerde, her kümenin iki havuzu olması için fazladan birkaç kuyruk yöneticisi eklenir. Ağ geçidi kuyruk yöneticisi, performans nedenleriyle havuz olarak kullanılmaz.

Yordam

1. Kuyruk yöneticilerini QM1, QM2, QM3, QM4, QM5 oluşturun ve başlatın.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QMn
strmqm QmgrName
```

Not: QM4 ve QM5 , kümeler için tam havuzları yedekler.

2. Her kuyruk yöneticisi için dinleyici tanımlayın ve başlatın.

```
*... On QMn
DEFINE LISTENER(TCP141n) TRPTYPE(TCP) IPADDR(hostname) PORT(141n) CONTROL(QMGR) REPLACE
START LISTENER(TCP141n)
```

3. Tüm kümeler için bir küme adı listesi oluşturun.

```
*... On QM1
DEFINE NAMELIST(ALL) NAMES(CL1, CL2) REPLACE
```

4. CL2 için QM2 ve QM4 tam havuzlarını CL1, QM3 ve QM5 tam havuzlarını yapın.

a) CL1 için:

```
*... On QM2 and QM4
ALTER QMGR REPOS(CL1) DEFCLXQ(SCTQ)
```

b) CL2 için:

```
*... On QM3 and QM5
ALTER QMGR REPOS(CL2) DEFCLXQ(SCTQ)
```

5. Her kuyruk yöneticisi ve küme için küme gönderen ve küme alıcı kanallarını ekleyin.

QM2, QM3, QM4 ve QM5 sistemlerinde aşağıdaki komutları çalıştırın; burada *c*, *nve m*, her kuyruk yöneticisi için Çizelge 26 sayfa 215 içinde gösterilen değerleri alır:

Çizelge 26. Küme 1 ve 2 oluşturmak için parametre değerleri			
Kuyruk yöneticisi	Küme <i>c</i>	Diğer havuz <i>n</i>	Bu havuz <i>m</i>
QM2	1	4	2
QM4	1	2	4
QM3	2	5	3
QM5	2	3	5

```
*... On QMm
DEFINE CHANNEL(CLc.QMn) CHLTYPE(CLUSSDR) CONNAME('localhost(141n)') CLUSTER(CLc) REPLACE
DEFINE CHANNEL(CLc.QMm) CHLTYPE(CLUSRCVR) CONNAME('localhost(141m)') CLUSTER(CLc) REPLACE
```

6. Ağ geçidi kuyruk yöneticisini (QM1) kümelerin her birine ekleyin.

```
*... On QM1
DEFINE CHANNEL(CL1.QM2) CHLTYPE(CLUSSDR) CONNAME('localhost(1412)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL1.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL2.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL2) REPLACE
DEFINE CHANNEL(CL2.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL2) REPLACE
```

7. Q1 yerel kuyruğunu CL2 kümesindeki QM3 kuyruk yöneticisine ekleyin.

```
*... On QM3
DEFINE QLOCAL(Q1) CLUSTER(CL2) REPLACE
```

8. Q1A kümelenmiş kuyruk yöneticisi diğer adını ağ geçidi kuyruk yöneticisine ekleyin.

```
*... On QM1
DEFINE QALIAS(Q1A) CLUSNL(ALL) TARGET(Q1) TARGTYPE(Queue) DEFBIND(NOTFIXED) REPLACE
```

Not: QM1 dışında herhangi bir kuyruk yöneticisinde kuyruk yöneticisi diğer adını kullanan uygulamalar, diğer ad kuyruğunu açtığı anda DEFBIND(NOTFIXED) belirtmelidir. **DEFBIND**, kuyruk uygulama tarafından açıldığında ileti üstbilgisindeki yöneltme bilgilerinin düzeltilip düzeltilmeyeceğini belirtir. Varsayılan değere (OPEN) ayarlanırsa, iletiler Q1@QM1'e yöneltilir. Q1@QM1 yoktur, bu nedenle diğer kuyruk yöneticilerinden gelen iletiler gitmeyen iletiler kuyruğuna gönderilir. Kuyruk özneteliğini DEFBIND(NOTFIXED) olarak ayarlayarak, varsayılan olarak **DEFBIND** kuyruk ayarına ayarlanan **amqspu**t gibi uygulamalar doğru şekilde davranır.

9. Tüm kümelenmiş kuyruk yöneticilerine ilişkin küme kuyruğu yöneticisi diğer ad tanımlamalarını ağ geçidi kuyruk yöneticisine (QM1) ekleyin.

```
*... On QM1
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSNL(ALL) REPLACE
```

İpucu: Başka bir kümedeki bir kuyruk yöneticisine gönderme yapan ağ geçidi kuyruk yöneticisi aktarma iletilerindeki kuyruk yöneticisi diğer adı tanımlamaları; bkz. Kümelenmiş kuyruk yöneticisi diğer adları.

Sonraki adım

1. Kuyruk diğer adı tanımlamasını sınamak için, QM3 kuyruk diğer adı tanımlamasını Q1A kullanarak QM2 'dan Q1 ' a bir ileti gönderebilirsiniz.

a. Bir ileti koymak için QM2 üzerinde **amqspuT** örnek programını çalıştırın.

```
C:\IBM\MQ>amqspuT Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

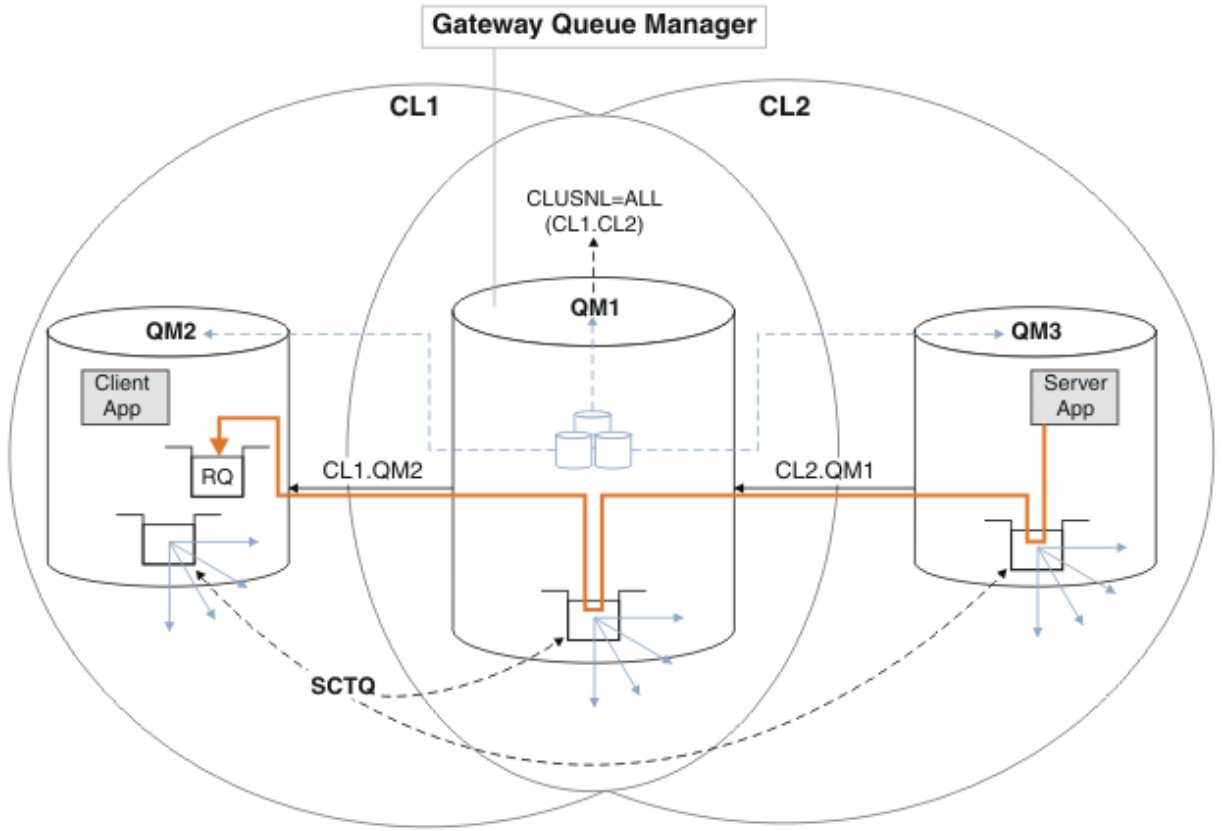
```
Sample AMQSPUT0 end
```

b. İletiyi şu adresten almak için **amqsgeT** örnek programını çalıştırın: Q1 on QM3

```
C:\IBM\MQ>amqsgeT Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Bir istek iletileri göndererek ve geçici dinamik yanıt kuyruğundan bir yanıt iletileri alarak kuyruk yöneticisi diğer ad tanımlarını sınavın.

Çizge, yanıt iletileri tarafından alınan yolu, RQolarak adlandırılan geçici bir dinamik kuyruğa geri gösterir. QM3' e bağlı sunucu uygulaması, QM2kuyruk yöneticisi adını kullanarak yanıt kuyruğunu açar. Kuyruk yöneticisi adı QM2 , QM1üzerinde kümelenmiş kuyruk yöneticisi diğer adı olarak tanımlanır. QM3 , yanıt iletilerini QM1' e yönlendirir. QM1 , iletiyi QM2' e yönlendirir.



Şekil 38. Yanıt iletisini farklı bir kümeye döndürmek için kuyruk yöneticisi diğer adının kullanılması

Yönlendirmenin çalışma şekli şu şekildedir. Her kümedeki her kuyruk yöneticisinin QM1 üzerinde bir kuyruk yöneticisi diğer adı tanımlaması vardır. Diğer adlar tüm kümelerde kümelenir. Her bir diğer addan bir kuyruk yöneticisine gelen gri çizgili oklar, her kuyruk yöneticisi diğer adının kümelerin en az birinde gerçek bir kuyruk yöneticisine çözümlendiğini gösterir. Bu durumda, QM2 diğer adı hem kümede CL1 hem de CL2 kümelenir ve CL1 içindeki QM2 gerçek kuyruk yöneticisine çözümlenir. Sunucu uygulaması, yanıt iletisini RQ kuyruk adını kullanarak ve QM2 kuyruk yöneticisine yanıt vererek yaratır. QM2 kuyruk yöneticisi diğer ad tanımlaması QM1 kümesinde CL2 ve QM2 kuyruk yöneticisinde CL2 tanımlı olmadığı için ileti QM1 olarak yöneltilir. İleti hedef kuyruk yöneticisine gönderilemediğinden, diğer ad tanımlamasına sahip kuyruk yöneticisine gönderilir.

QM1 places the message on the cluster transmission queue on QM1 for transferal to QM2. QM1, QM2 için QM1 üzerindeki kuyruk yöneticisi diğer adı tanımlaması QM2 'i gerçek hedef kuyruk yöneticisi olarak tanımladığından, iletiyi QM2 ' e yöneltir. Diğer ad tanımlamaları yalnızca gerçek tanımlamalara gönderme yapabildiği için tanımlama çevrimsel değildir; diğer ad kendisini gösteremez. Hem QM1 hem de QM2 aynı kümede (CL1) olduğundan, gerçek tanımlama QM1 tarafından çözümlenir. QM1, CL1 havuzundan QM2 ile ilgili bağlantı bilgilerini bulur ve iletiyi QM2 ' e yönlendirir. İletinin QM1 tarafından yeniden yönlendirilmesi için, sunucu uygulaması yanıt kuyruğunu DEFBIND seçeneği MQBND_BIND_NOT_FIXED olarak ayarlanmış şekilde açmış olmalıdır. Sunucu uygulaması yanıt kuyruğunu MQBND_BIND_ON_OPEN seçeneğiyle açtıysa, ileti yeniden yönlendirilmez ve gitmeyen iletiler kuyruğuna gönderilir.

- QM3 üzerinde bir tetikleyici ile kümelenmiş bir istek kuyruğu yaratın.

```
*... On QM3
DEFINE QLOCAL(QR) CLUSTER(CL2) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE)
PROCESS(ECHO) REPLACE
```

b. Ağ geçidi kuyruk yöneticisinde (QM1) QR için kümelenmiş bir kuyruk diğer adı tanımlaması yaratın.

```
*... On QM1
DEFINE QALIAS(QRA) CLUSNL(ALL) TARGET(QR) TARGTYPE(Queue) DEFBIND(NOTFIXED) REPLACE
```

c. QM3 üzerinde **amqsech** örnek yankı programını başlatmak için bir süreç tanımlaması yaratın.

```
*... On QM3
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

d. Geçici dinamik yanıt kuyruğunu yaratmak için QM2 örnek program **amqsreq** için bir model kuyruğu yaratın.

```
*... On QM2
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

e. Kuyruk diğer adı tanımlamasını sınamak için, QRA kuyruk diğer adı tanımlamasını kullanarak QM2 'den QR ' e QM3 üzerinde bir istek gönderin.

i) Tetikleyici izleme programını QM3 üzerinde çalıştırın.

```
runmqtrm -m QM3
```

Çıkış:

```
C:\IBM\MQ>runmqtrm -m QM3
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
01/02/2012 16:17:15: WebSphere MQ trigger monitor started.
```

```
-----
01/02/2012 16:17:15: Waiting for a trigger message
```

ii) İstek koymak ve yanıt beklemek için örnek programı **amqsreq** on QM2 çalıştırın.

```
C:\IBM\MQ>amqsreq QRA QM2
Sample AMQSREQ0 start
server queue is QRA
replies to 4F2961C802290020
A request message from QM2 to QR on QM3
```

```
response <A request message from QM2 to QR on QM3>
no more replies
Sample AMQSREQ0 end
```

İlgili kavramlar

[“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157](#)

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, SYSTEM.CLUSTER.TRANSMIT.QUEUE' a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanımlara karşı denetlenir.

[“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278](#)

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

İlgili görevler

[“Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194](#)

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

[“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213](#)

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelenmiş bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanız için bir yol sağlar.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'deki altyapılar için geçerlidir.

İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanız için bir yol sağlar.

Başlamadan önce

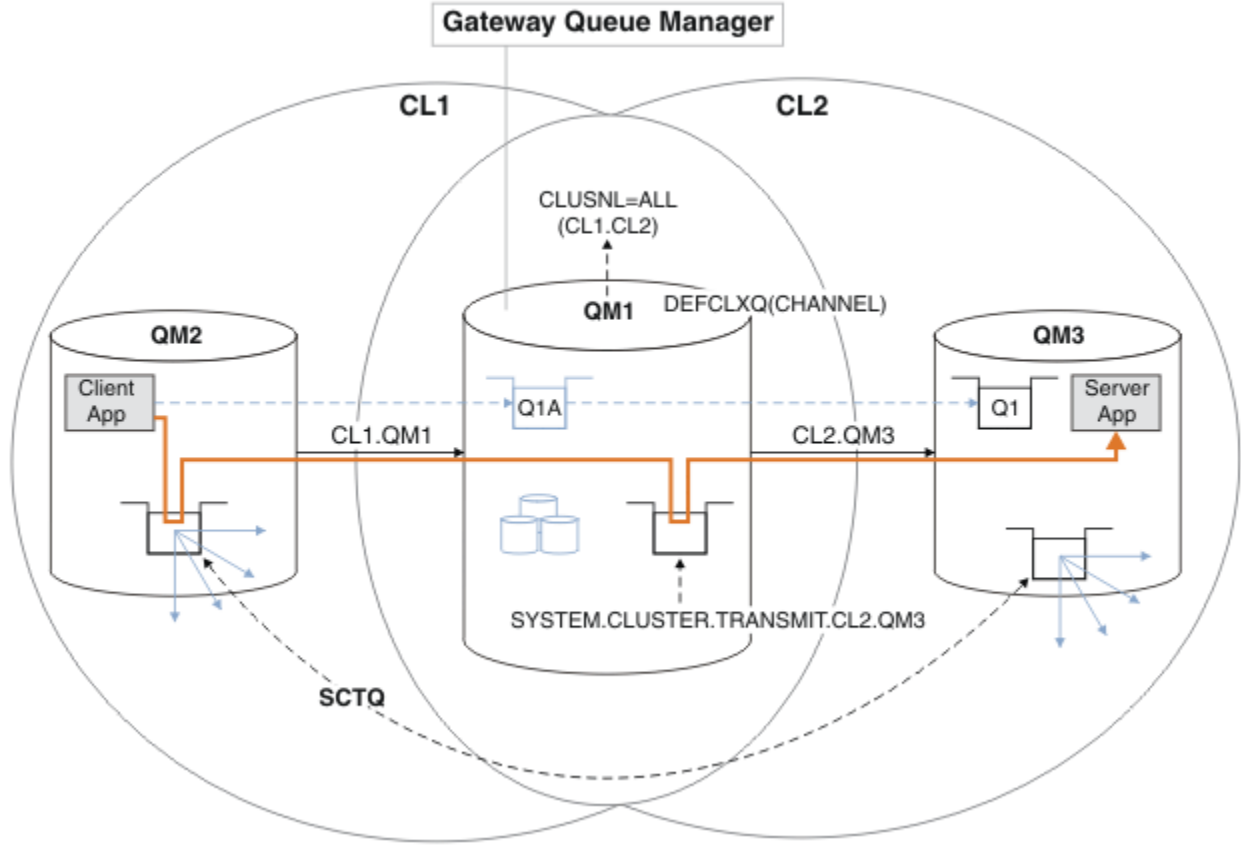
1. The gateway queue manager must be on Version 7.5, or later, and on a platform other than z/OS.
2. Construct the overlapping clusters shown in [Şekil 37 sayfa 214](#) in [“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213](#) by following the steps in that task.

Bu görev hakkında

Mimariyi birden çok küme kuyruğu ile uygulamak için, ağ geçidi kuyruk yöneticinizin Version 7.5 ya da daha sonraki bir sürümü olması gerekir. Birden çok küme iletim kuyruğu kullanmak için tek yapmanız gereken, ağ geçidi kuyruk yöneticisindeki varsayılan küme iletim kuyruğu tipini değiştirmenizdir. QM1 üzerindeki **DEFCLXQ** kuyruk yöneticisi özneliğinin değerini SCTQ 'dan KANAL' a değiştirin; bkz. [Şekil 39 sayfa 220](#). Çizgede bir ileti akışı gösterilir. Diğer kuyruk yöneticilerine ya da diğer kümelere yönelik akışlar için, kuyruk yöneticisi ek kalıcı dinamik küme iletim kuyrukları yaratır. Her bir küme gönderici kanalı, iletileri farklı bir küme iletim kuyruğundan aktarır.

Ağ geçidi kuyruk yöneticisini ilk kez kümelere bağlamadığınız sürece, değişiklik hemen yürürlüğe girmez. Bu görev, var olan bir yapılandırmada değişiklik yönetiminin tipik vakasına ilişkin adımları içerir. Bir kuyruk

yöneticisini, bir kümeye ilk katılırken ayrı küme iletim kuyrukları kullanacak şekilde ayarlamak için; bkz. "Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları" sayfa 194.



Şekil 39. İstemci sunucusu uygulaması göbekte konuşlandırıldı ve ağ geçidi kuyruk yöneticisinde ayrı küme iletim kuyruklarıyla mimari konuştu.

Yordam

1. Ağ geçidi kuyruk yöneticisini, ayrı küme iletim kuyrukları kullanacak şekilde değiştirin.

```
*... On QM1
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. Ayrı küme iletim kuyruklarına geçin.

Çalışmakta olan herhangi bir kümeli gönderici kanalı, bir sonraki başlatıldığında ayrı küme iletim kuyrukları kullanmaya geçiş yapar.

Çalışmakta olan kanalları değiştirmek için, kuyruk yöneticisini yeniden başlatın ya da aşağıdaki adımları izleyin:

- a) SYSTEM.CLUSTER.TRANSMIT.QUEUE ile çalışan küme gönderen kanallarını listele.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

Yanıt, kanal durumu raporlarının bir listesidir:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)           CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1412)) CURRENT
RQMNAME(QM2)              STATUS(RUNNING)
SUBSTATE(MQGET)           XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
```

```

CHANNEL (CL2.QM3)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3)             STATUS (RUNNING)
SUBSTATE (MQGET)          XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM5)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1415)) CURRENT
RQMNAME (QM5)             STATUS (RUNNING)
SUBSTATE (MQGET)          XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM4)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1414)) CURRENT
RQMNAME (QM4)             STATUS (RUNNING)
SUBSTATE (MQGET)          XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)

```

b) Çalışmakta olan kanalları durdurun

Listedeki her bir kanal için şu komutu çalıştırın:

```
*... On QM1
STOP CHANNEL(ChannelName)
```

Burada *ChannelName* , her CL1.QM2, CL1.QM4, CL1.QM3, CL1.QM5' dir.

Yanıt, komutun kabul edildiğinden emin olun:

```
AMQ8019: Stop WebSphere MQ channel accepted.
```

c) Durdurulan kanalların izlenmesi

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

Yanıt, halen çalışmakta olan kanalların ve durdurulduğu kanalların bir listesidir:

```

AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM2)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1412)) CURRENT
RQMNAME (QM2)             STATUS (STOPPED)
SUBSTATE ( )              XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM3)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3)             STATUS (STOPPED)
SUBSTATE ( )              XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL2.QM5)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1415)) CURRENT
RQMNAME (QM5)             STATUS (STOPPED)
SUBSTATE ( )              XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL (CL1.QM4)          CHLTYPE (CLUSSDR)
CONNNAME (127.0.0.1(1414)) CURRENT
RQMNAME (QM4)             STATUS (STOPPED)
SUBSTATE ( )              XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)

```

d) Her bir durdurulmuş kanalı başlatın.

Çalışmakta olan tüm kanallar için bu adımı gerçekleştirin. Bir kanal durdurulamazsa, **STOP CHANNEL** komutunu **FORCE** seçeneğiyle yeniden çalıştırabilirsiniz. Kanal durdurulamazsa ve kanalı uyumlulaştırmak için diğer kuyruk yöneticisini yeniden başlatamazsınız; **FORCE** seçeneğini ayarlamak için bir örnek de olur.

```
*... On QM1
START CHANNEL (CL2.QM5)
```

Yanıt, komutun kabul edildiğinden emin olun:

AMQ8018: Start WebSphere MQ channel accepted.

e) Anahtarlamalı iletim kuyruklarını izleyin.

Monitor the gateway queue manager error log for the message "AMQ7341 CL2.QM3 kanalına ilişkin iletim kuyruğu SYSTEM.CLUSTER.TRANSMIT.QUEUE / CL2.QM3 olur."

f) SYSTEM.CLUSTER.TRANSMIT.QUEUE ' un artık kullanılmadığını denetleyin

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
DISPLAY QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE) CURDEPTH
```

Yanıt, kanal durumu raporlarının ve SYSTEM.CLUSTER.TRANSMIT.QUEUE' in derinliğinin bir listesidir:

AMQ8420: Channel Status not found.

AMQ8409: Display Queue details.

```
QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE) TYPE(QLOCAL)
CURDEPTH(0)
```

g) Hangi kanalların başlatıldığı izleme

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

Yanıt, bu durumda yeni varsayılan küme iletim kuyruklarıyla çalışmakta olan bu durumda, kanalların bir listesidir:

AMQ8417: Display Channel Status details.

```
CHANNEL (CL1.QM2) CHLTYPE (CLUSSDR)
CONNAME (127.0.0.1(1412)) CURRENT
RQMNAME (QM2) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL1.QM2)
```

AMQ8417: Display Channel Status details.

```
CHANNEL (CL2.QM3) CHLTYPE (CLUSSDR)
CONNAME (127.0.0.1(1413)) CURRENT
RQMNAME (QM3) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL2.QM3)
```

AMQ8417: Display Channel Status details.

```
CHANNEL (CL2.QM5) CHLTYPE (CLUSSDR)
CONNAME (127.0.0.1(1415)) CURRENT
RQMNAME (QM5) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL2.QM5)
```

AMQ8417: Display Channel Status details.

```
CHANNEL (CL1.QM4) CHLTYPE (CLUSSDR)
CONNAME (127.0.0.1(1414)) CURRENT
RQMNAME (QM4) STATUS (RUNNING)
SUBSTATE (MQGET)
XMITQ (SYSTEM.CLUSTER.TRANSMIT.CL1.QM4)
```

Sonraki adım

1. Test the automatically defined cluster transmission queue by sending a message from QM2 to Q1 on QM3, resolving queue name with the queue alias definition Q1A

a. Bir ileti koymak için QM2 üzerinde **amqspout** örnek programını çalıştırın.

```
C:\IBM\MQ>amqspout Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

b. İletiyi şu adresten almak için **amqsget** örnek programını çalıştırın: Q1 on QM3

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Küme kuyruklarına ilişkin iletilerin kaynağı olduğu kuyruk yöneticilerindeki küme kuyrukları için güvenliği yapılandırarak, güvenliği yeniden yapılandırmayı düşünün.

İlgili kavramlar

“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, SYSTEM . CLUSTER . TRANSMIT . QUEUE' a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanımlara karşı denetlenir.

“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

İlgili görevler

“Bir kümeye kuyruk yöneticisi eklenmesi: ayrı iletim kuyrukları” sayfa 194

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Kümeleme kuyruklarına ve konulara ilişkin iletiler, birden çok küme iletim kuyruğu kullanılarak aktarılır.

“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümedenmiş bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümedenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanız için bir yol sağlar.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra, ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtım için ek bir küme kullanır.

[“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281](#)

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS dışındaki altyapılar için geçerlidir.

Küme Kuyruğunun Kuyruk Yöneticisinden Kaldırılması

Toronto 'da INVENTQ kuyruğunu devre dışı bırakın. Send all the inventory messages to New York, and delete the INVENTQ queue at Toronto when it is empty.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Kuyruk barındıran bir kuyruk yöneticisi eklenmesi” sayfa 210 içinde açıklandığı şekilde ayarlanmıştır. Dört kuyruk yöneticisi içerir. LONDON ve NEWYORK her ikisi de tam havuzları içerir. PARIS ve TORONTO , kısmi havuzları barındırır. The inventory application runs on the systems in New York and Toronto and is driven by the arrival of messages on the INVENTQ queue.
- Azaltılmış iş yükü nedeniyle, artık Toronto 'da stok uygulamasını çalıştırmak istemeniz gerekmez. You want to disable the INVENTQ queue hosted by the queue manager TORONTO, and have TORONTO feed messages to the INVENTQ queue in NEWYORK.
- Tüm dört sistem arasında ağ bağlantısı vardır.
- Ağ protokolü TCP 'dir.

Bu görev hakkında

Bir küme kuyruğunu kaldırmak için aşağıdaki adımları izleyin.

Yordam

1. Kuyruğun artık kullanılmadığını belirtir.

Bir kuyruktan bir kuyruğu kaldırmak için, küme adını yerel kuyruk tanımlamasından kaldırın. Alter the INVENTQ on TORONTO so that it is not accessible from the rest of the cluster:

```
ALTER QLOCAL(INVENTQ) CLUSTER(' ')
```

2. Kuyruğun artık kullanılmadığından emin olun.

On a full repository queue manager, either LONDON or NEWYORK, check that the queue is no longer hosted by queue manager TORONTO by issuing the following command:

```
DIS QCLUSTER (INVENTQ)
```

ALTER komutu başarıyla tamamlandıysa, sonuçlarda TORONTO listelenmez.

3. Kuyruğu geçersiz kılın.

Disable the INVENTQ queue at TORONTO so that no further messages can be written to it:

```
ALTER QLOCAL(INVENTQ) PUT(DISABLED)
```


Şimdi MQ00_BIND_ON_OPEN komutunu kullanarak bu kuyruğa aktarılan iletiler, ölü-harfli kuyruğa gider. Tüm uygulamaları, iletileri bu kuyruk yöneticisinden kuyruğa belirtik olarak yerleştirmekten alıkoymanız gerekir.

4. Boş oluncaya kadar kuyruğu izleyin.

Monitor the queue using the DISPLAY QUEUE command, specifying the attributes IPPROCS, İŞLEMLER, and DENE, or use the WRKMQMSTS command on IBM i. Giriş ve çıkış işlemleri sayısı ve kuyruğun yürürlükteki derinliği sıfır olduğunda, kuyruk boş olur.

5. Kuşku içinde olmayan iletiler olup olmadığını görmek için kanalı izleyin.

INVENTORY .TORONTO kanalındaki belirsiz iletiler olmadığından emin olmak için, diğer kuyruk yöneticilerinin her birinde INVENTORY .TORONTO adlı küme gönderici kanalını izleyin. Her kuyruk yöneticisinden INDOUBT deęiştirgesini belirterek DISPLAY CHSTATUS komutunu verin:

```
DISPLAY CHSTATUS(INVENTORY.TORONTO) INDOUBT
```

Belirsiz iletiler varsa, devam etmeden önce bunları çözmeniz gerekir. Örneğin, RESOLVE CHANNEL komutunu vermeyi ya da kanalı durdurup yeniden başlatmayı deneyebilirsiniz.

6. Yerel kuyruğu silin.

TORONTO' ta stok uygulamasına teslim edilecek başka ileti kalmadığından emin olun, kuyruğu silebilirsiniz:

```
DELETE QLOCAL(INVENTQ)
```

7. Şimdi, döküm uygulamasını Toronto 'daki sistemden kaldırabilirsiniz.

Uygulamanın kaldırılması, yinelemeyi önler ve sistemde yer tasarrufu sağlar.

Sonuçlar

Bu görev tarafından ayarlanan küme, önceki görevin ayarlarıyla aynı şekilde ayarlanır. The difference is the INVENTQ queue is no longer available at queue manager TORONTO.

1. adımda kuyruğu hizmet dışına çıkardığınızda, TORONTO kuyruk yöneticisi, iki tam havuz kuyruğu yöneticisine bir ileti gönderdi. Bu durum, durum deęişimlerinden haberdar oldu. The full repository queue managers pass on this information to other queue managers in the cluster that have requested updates to information concerning the INVENTQ.

Kuyruk yöneticisi INVENTQ kuyruğuna bir ileti yerleştirdiğinde, güncellenen kısmi havuz, INVENTQ kuyruğunun yalnızca NEWYORK kuyruk yöneticisinde kullanılabilir olduğunu belirtir. İleti, NEWYORK kuyruk yöneticisine gönderilir.

Sonraki adım

Bu görevde, kaldırmak için yalnızca bir kuyruk ve bunu kaldırmak için yalnızca bir küme vardı.

Birçok küme adı içeren bir ad listesine başvuruda bulunan birçok kuyruk olduğunu varsayın. Örneğin, TORONTO kuyruk yöneticisi yalnızca INVENTQ deęil, aynı zamanda PAYROLLQ, SALESQ ve PURCHASESQ anasistemlerini barınayabilir. TORONTO , bu kuyrukların tüm uygun kümelerde, INVENTORY , PAYROLL, SALES ve PURCHASES ' de kullanılabilir olmasını sağlar. TORONTO kuyruk yöneticisinde küme adlarının bir ad listesini tanımlayın:

```
DEFINE NAMELIST(TOROLIST)
  DESCR('List of clusters TORONTO is in')
  NAMES(INVENTORY, PAYROLL, SALES, PURCHASES)
```

Ad listesini her kuyruk tanımlamasına ekleyin:

```
DEFINE QLOCAL(INVENTQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PAYROLLQ) CLUSNL(TOROLIST)
```

```
DEFINE QLOCAL(SALESQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PURCHASESQ) CLUSNL(TOROLIST)
```

SALES işlemi PURCHASES işlemi tarafından ele alınmadığından, SALES kümesinden tüm bu kuyrukları kaldırmak istediğinizi varsayalım. Yapmanız gereken tek şey, TOROLIST ' un adını, içinden SALES kümesinin adını kaldırmak için değiştirmeniz.

Ad listesindeki kümelerin birinden tek bir kuyruğu kaldırmak istiyorsanız, kalan küme adları listesini içeren bir ad listesi yaratın. Daha sonra, yeni ad listesini kullanmak için kuyruk tanımlamasını değiştirin. PAYROLLQ ögesini INVENTORY kümesinden kaldırmak için:

1. Ad listesi yarat:

```
DEFINE NAMELIST(TOROSHORTLIST)
DESCR('List of clusters TORONTO is in other than INVENTORY')
NAMES(PAYROLL, SALES, PURCHASES)
```

2. PAYROLLQ kuyruk tanımlamasını değiştirin:

```
ALTER QLOCAL(PAYROLLQ) CLUSNL(TOROSHORTLIST)
```

Tüm havuzu başka bir kuyruk yöneticisine taşıma

Bir tam havuzu bir kuyruk yöneticisinden diğerine taşıyın ve ikinci havuzda tutulan bilgilerden yeni havuzu oluşturmak için.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Küme için kuyruk yöneticisi eklenmesi” sayfa 191’inde açıklandığı şekilde ayarlanmıştır.
- İş nedenlerinden dolayı, şimdi tam havuzu kuyruk yöneticisinden (LONDON) kaldırmak ve bunu PARIS kuyruk yöneticisinde tam bir havuzla değiştirmek istiyorsunuz. NEWYORK kuyruk yöneticisi dolu bir havuzu tutmaya devam etmek için.

Bu görev hakkında

Tüm havuzu başka bir kuyruk yöneticisine taşımak için aşağıdaki adımları izleyin.

Yordam

1. Tam havuz kuyruk yöneticisi olmasını sağlamak için PARIS seçeneğini değiştirin.

PARIS' ta şu komutu verin:

```
ALTER QMGR REPOS(INVENTORY)
```

2. PARIS üzerine bir CLUSTSDR kanalı ekle

PARIS , şu anda LONDON' e işaret eden bir küme gönderici kanalı var. LONDON artık küme için tam bir havuz tutmaktan başka bir şey değil. PARIS , şu anda diğer tam havuzun tutulduğu NEWYORK' ye işaret eden yeni bir kümeli gönderici kanalına sahip olmalıdır.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at NEWYORK')
```

3. Define a CLUSSDR channel on NEWYORK that points to PARIS

Şu anda NEWYORK , LONDON' e işaret eden bir küme gönderen kanalına sahiptir. Diğer tam havuz PARIS'a taşındıysa, NEWYORK ' ta PARIS' e işaret eden yeni bir kümeli gönderici kanalı eklemeniz gerekir.

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from NEWYORK to repository at PARIS')
```

When you add the cluster-sender channel to PARIS, PARIS learns about the cluster from NEWYORK. NEWYORK' tan gelen bilgileri kullanarak kendi tam havuzunu oluşturur.

4. Check that queue manager PARIS now has a full repository

Check that queue manager PARIS has built its own full repository from the full repository on queue manager NEWYORK. Aşağıdaki komutları verin:

```
DIS QCLUSTER(*) CLUSTER (INVENTORY)
DIS CLUSQMR(*) CLUSTER (INVENTORY)
```

Bu komutların, NEWYORK' ta olduğu gibi, bu kümedeki aynı kaynaklara ilişkin ayrıntıları gösterdiğine bakın.

Not: If queue manager NEWYORK is not available, this building of information cannot complete. Görev tamamlanincaya kadar sonraki adıma geçmeyin.

5. LONDON' üzerinde kuyruk yöneticisi tanımlamasını değiştirme

Son olarak LONDON konumunda kuyruk yöneticisini değiştirerek, küme için artık tam bir havuz tutmamasını sağlar. LONDON' ta şu komutu verin:

```
ALTER QMGR REPOS(' ')
```

Kuyruk yöneticisi artık herhangi bir küme bilgisi almıyor. 30 günden sonra, tam havuzunda saklanan bilgilerin süresi dolur. Kuyruk yöneticisi LONDON artık kendi kısmi havuzunu oluşturur.

6. Üstün tanımlamaları kaldırın ya da değiştirin.

Küğünüzün yeni düzenlemesinin beklendiği gibi çalıştığını doğruladığınızda, artık doğru olmayan, el ile tanımlanan CLUSSDR tanımlarını kaldırın ya da değiştirin.

- PARIS kuyruk yöneticisinde, küme gönderen kanalını LONDON' e durdurmanız ve silmeniz ve daha sonra, kümenin otomatik kanalları yeniden kullanabilmesi için kanal başlatma komutunu yayınlamanız gerekir:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
START CHANNEL(INVENTORY.LONDON)
```

- NEWYORK kuyruk yöneticisinde, küme gönderen kanalını LONDON' e durdurmanız ve silmeniz ve daha sonra, kümenin otomatik kanalları yeniden kullanabilmesi için kanal başlatma komutunu yayınlamanız gerekir:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
START CHANNEL(INVENTORY.LONDON)
```

- Replace all other cluster-sender channels in the cluster that point to LONDON with channels that point to either NEWYORK or PARIS. Bu küçük örnekte, başka kimse yok. Unuttuğunuz başka kişilerin olup olmadığını denetlemek için, her kuyruk yöneticisinden DISPLAY CHANNEL komutunu verin ve TYPE(CLUSSDR) komutunu belirtin. Örneğin:

```
DISPLAY CHANNEL(*) TYPE(CLUSSDR)
```

Tam havuzu LONDON 'tan PARIS' a taşıdıktan sonra en kısa zamanda bu görevi gerçekleştirmeniz önemlidir. In the time before you perform this task, queue managers that have manually defined

CLUSDR channels named INVENTORY . LONDON might send requests for information using this channel.

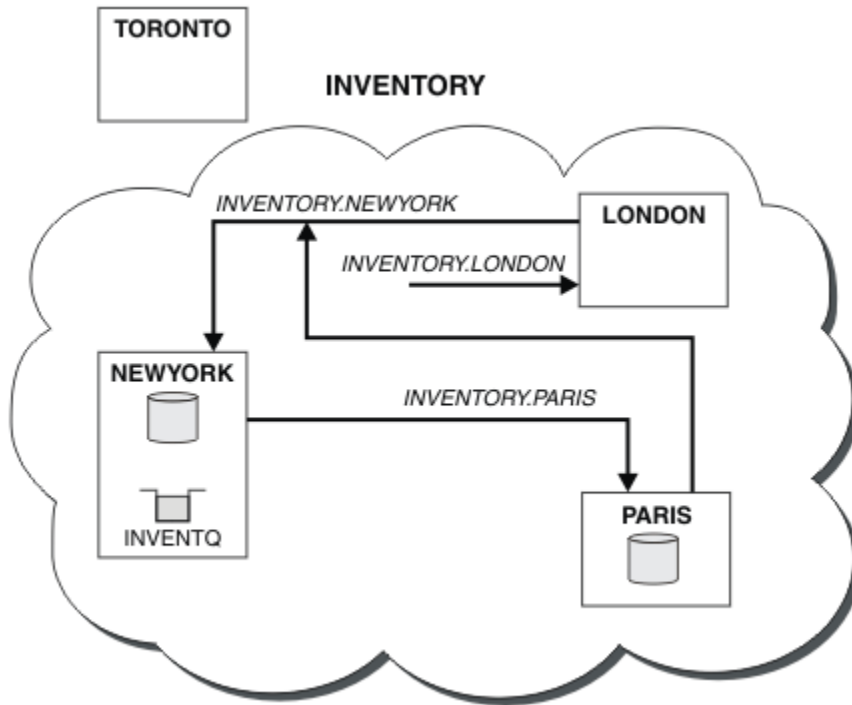
LONDON , tam havuz olmayı bıraktıktan sonra, bu tür istekler alırsa, hata iletilerini kuyruk yöneticisi hata günlüğüne yazacaktır. Aşağıdaki örneklerde, LONDON' ta hangi hata iletilerinin görülebileceği gösterilmektedir:

- AMQ9428: Unexpected publication of a cluster queue object received
- AMQ9432: Query received by a non-repository queue manager

Kuyruk yöneticisi LONDON , artık tam havuz olmadığı için bilgi isteklerine yanıt vermiyor. The queue managers requesting information from LONDON must rely on NEWYORK for cluster information until their manually defined CLUSSDR definitions are corrected to point to PARIS. Bu durumun uzun vadede geçerli bir yapılandırma olarak kabul edilmemesi gerekir.

Sonuçlar

Şekil 40 sayfa 228 , bu görev tarafından ayarlanan kümeyi gösterir.



Şekil 40. Tam havuzun bulunduğu INVENTORY kümesi PARIS' e taşındı

Var olan bir ağı bir kümeye dönüştürme

Var olan dağıtılmış kuyruklama ağını bir kümeye dönüştürün ve kapasiteyi artırmak için ek bir kuyruk yöneticisi ekleyin.

Başlamadan önce

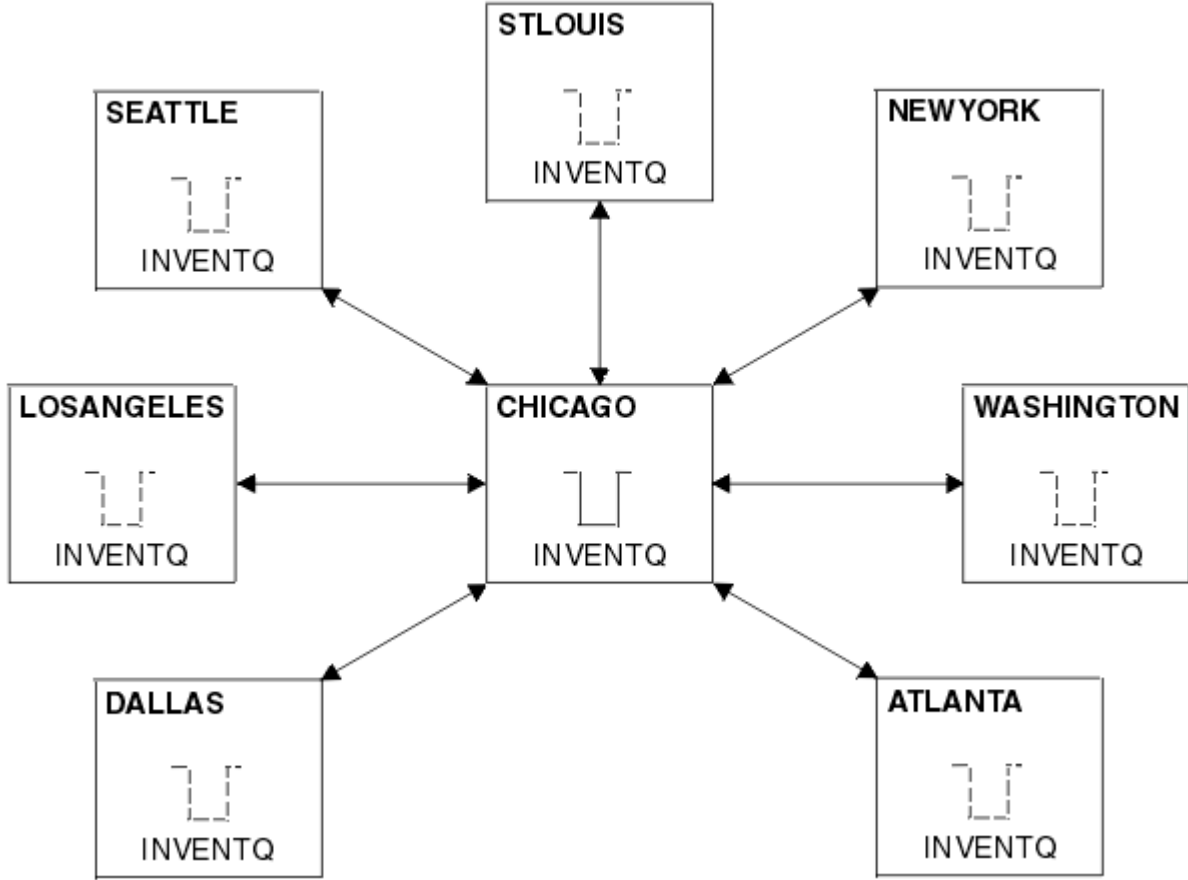
“Yeni bir küme ayarlanıyor” sayfa 181 ile “Tüm havuzu başka bir kuyruk yöneticisine taşıma” sayfa 226 içinde yeni bir küme yaratmış ve genişletmiş olduğunuz. Sonraki iki görev farklı bir yaklaşımı keşfeder: var olan bir kuyruk yöneticisi ağını bir kümeye dönüştürmesi.

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- An IBM WebSphere MQ network is already in place, connecting the nationwide branches of a chain store. Bir merkezi ve konuşan yapısı vardır: tüm kuyruk yöneticileri tek bir merkezi kuyruk yöneticisine bağlıdır. Merkezi kuyruk yöneticisi, döküm uygulamasının çalıştığı sistemde yer alıyor. The application is driven by the arrival of messages on the INVENTQ queue, for which each queue manager has a remote-queue definition.

Bu ağ Şekil 41 sayfa 229' ta gösterilmektedir.



Şekil 41. Bir göbek ve konuşan ağ

- Yönetimi kolaylaştırmak için, bu ağı bir kümeye dönüştürecek ve iş yükünü paylaşmak üzere merkezi bir yerde başka bir kuyruk yöneticisi oluşturacağız.

Küme adı CHNSTORE.

Not: The cluster name CHNSTORE was selected to allow cluster-receiver channel names to be created using names in the format *cluster-name.queue-manager* that do not exceed the maximum length of 20 characters, for example CHNSTORE.WASHINGTON.

- Hem merkezi kuyruk yöneticileri, tam havuzları barınabilmekte ve döküm uygulaması için erişilebilir durumda olmalıdır.
- Döküm uygulaması, merkezi kuyruk yöneticilerinden herhangi biri tarafından barındırılan INVENTQ kuyruğunda iletilerin gelmesiyle yönlendirilir.
- Döküm uygulaması, birden çok kuyruk yöneticisi tarafından paralel ve erişilebilir durumda çalışan tek uygulama olacaktır. Diğer tüm uygulamalar daha önce olduğu gibi çalışmaya devam eder.
- Tüm dalların, iki merkezi kuyruk yöneticisine ağ bağlantısı vardır.
- Ağ protokolü TCP 'dir.

Bu görev hakkında

Var olan bir ağı bir kümeye dönüştürmek için aşağıdaki adımları izleyin.

Yordam

1. İleti zenginlikleri için envanter uygulamasını gözden geçirin.

Devam etmeden önce, uygulamanın ileti zenginlikleri ile başa çıkabilmesi için emin olun. İleti zenginlikleri, iki uygulama arasında değiş tokuş edilen ve iletilerin belirli bir kuyruk yöneticisi tarafından ya da belirli bir sırada işlenmesi gereken etkileşimli iletiler arasındaki ilişkidir. İleti zenginlikleri hakkında daha fazla bilgi için bkz: [“İleti zenginlikleri işleniyor” sayfa 271](#)

2. İki merkezi kuyruk yöneticisini, tam havuz kuyruğu yöneticilerine yapmak için değiştirin.

The two queue managers CHICAGO and CHICAG02 are at the hub of this network. Zincir mağaza kümesiyle ilişkili tüm etkinliği, bu iki kuyruk yöneticisine yoğunlaştırabilirsiniz. Döküm uygulamasının yanı sıra, INVENTQ kuyruğuna ilişkin tanımlamalar da, bu kuyruk yöneticilerinin küme için iki tam havuza ev sahipliği etmesini istersiniz. Her iki kuyruk yöneticisinin her birinde aşağıdaki komutu verin:

```
ALTER QMGR REPOS(CHNSTORE)
```

3. Her kuyruk yöneticisi için bir CLUSTRVR kanalı tanımlayın.

Kümedeki her kuyruk yöneticisinde, bir küme alıcı kanalı ve bir küme gönderici kanalı tanımlayın. İlk olarak hangi kanalda tanımladığınız önemli değil.

Her bir kuyruk yöneticisini, ağ adresini ve diğer bilgileri kümeye duyurmak için bir CLUSTRVR tanımlaması yapın. For example, on queue manager ATLANTA:

```
DEFINE CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(ATLANTA.CHSTORE.COM) CLUSTER(CHNSTORE)
DESCR('Cluster-receiver channel')
```

4. Her kuyruk yöneticisinde bir CLUSSDR kanalı tanımlayın

Kuyruk yöneticisini tüm havuz kuyruğu yöneticilerinden birine ya da birine bağlamak için her kuyruk yöneticisinde bir CLUSSDR tanımlaması yapın. Örneğin, ATLANTA ile CHICAG02 arasında bağlantı verebilirsiniz:

```
DEFINE CHANNEL(CHNSTORE.CHICAG02) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAG02.CHSTORE.COM) CLUSTER(CHNSTORE)
DESCR('Cluster-sender channel to repository queue manager')
```

5. Install the inventory application on CHICAG02.

Kuyruk yöneticisine ilişkin döküm uygulamasına zaten sahip olduğunuz CHICAGO. Now you need to make a copy of this application on queue manager CHICAG02.

6. Merkezi kuyruk yöneticilerindeki INVENTQ kuyruğunu tanımlayın.

On CHICAGO, modify the local queue definition for the queue INVENTQ to make the queue available to the cluster. Komutu verin:

```
ALTER QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

CHICAG02' ta, aynı kuyruk için bir tanımlama yapın:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

On z/OS, you can use the MAKEDEF option of the KOMUT function of **CSQUTIL** to make an exact copy on CHICAG02 of the INVENTQ on CHICAGO.

Bu tanımları yaptığınız zaman, CHICAGO ve CHICAG02 ' taki tüm havuzlara bir ileti gönderilir ve bunlar içindeki bilgiler güncellenir. Kuyruk yöneticisi, INVENTQ' e bir ileti yerleştirdiğinde, tüm havuzlarda, iletiler için bir hedef seçimi olduğunu öğrenir.

7. Küme değişikliklerinin yayıldığını doğrulayın.

Önceki adımda yarattığınız tanımların küme olarak yayıldığını doğrulayın. Tam havuz kuyruk yöneticisinde aşağıdaki komutu verin:

Birbirine bağı yeni bir küme eklenmesi

Bazı kuyruk yöneticilerini var olan bir kümeyle paylaşan yeni bir küme ekleyin.

Başlamadan önce

Not:

1. Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.
2. Bu görevi başlatmadan önce, kuyruk adı çakışmalarını denetleyin ve sonuçları anlayın. Devam etmeden önce bir kuyruğu yeniden adlandırmanız ya da kuyruk diğer adlarını ayarlamanız gerekebilir.

Senaryo:

- Bir WebSphere MQ kümesi, “[Var olan bir ağı bir kümeye dönüştürme](#)” sayfa 228 içinde açıklandığı şekilde ayarlandı.
- MAILORDER adlı yeni bir küme gerçekleştirilmek üzere. Bu küme, CHNSTORE kümesi, CHICAGO, CHICAGO2 , SEATTLE, ve ATLANTA ve iki ek kuyruk yöneticisi; HARTFORD ve OMAHA olmak üzere, dört kuyruk yöneticisinde oluşur. The MAILORDER application runs on the system at Omaha, connected to queue manager OMAHA. It is driven by the other queue managers in the cluster putting messages on the MORDERQ queue.
- The full repositories for the MAILORDER cluster are maintained on the two queue managers CHICAGO and CHICAGO2.
- Ağ protokolü TCP 'dir.

Bu görev hakkında

Birbiriyle bağlantılı yeni bir küme eklemek için bu adımları izleyin.

Yordam

1. Küme adları için bir ad listesi oluşturun.

The full repository queue managers at CHICAGO and CHICAGO2 are now going to hold the full repositories for both of the clusters CHNSTORE and MAILORDER . İlk olarak, kümelerin adlarını içeren bir ad listesi oluşturun. Define the namelist on CHICAGO and CHICAGO2 , as follows:

```
DEFINE NAMELIST(CHAINMAIL)
  DESCR('List of cluster names')
  NAMES(CHNSTORE, MAILORDER)
```

2. İki kuyruk yöneticisi tanımlamasını değiştirin.

Şimdi, CHICAGO ve CHICAGO2 konumunda iki kuyruk yöneticisi tanımını değiştirin. Currently these definitions show that the queue managers hold full repositories for the cluster CHNSTORE. Bu tanımlamayı, kuyruk yöneticilerinin CHAINMAIL ad listesinde listelenen tüm kümeler için tam havuz tutturacağını göstermek için değiştirin. CHICAGO ve CHICAGO2 kuyruk yöneticisi tanımlarını değiştirin:

```
ALTER QMGR REPOS(' ') REPOSNL(CHAINMAIL)
```

3. CHICAGO ve CHICAGO2 üzerindeki CLUSTRVR kanallarını değiştirin.

CHICAGO ve CHICAGO2 'daki CLUSTRVR kanal tanımlamaları, kanalların CHNSTORE kümesinde olduğunu gösterir. Kanalların, CHAINMAIL ad listesinde listelenen tüm kümelerde kullanılabilir

olduğunu göstermek için kümeli alıcı tanımlamasını değiştirmeniz gerekir. Change the cluster-receiver definition at CHICAGO:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

CHICAGO2' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

4. CHICAGO ve CHICAGO2 üzerindeki CLUSTSDR kanallarını değiştirin.

İki CLUSSDR kanal tanımlamalarını, ad listesini eklemek üzere değiştirin. CHICAGO' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

CHICAGO2' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

5. SEATTLE ve ATLANTA üzerinde bir ad listesi oluşturun.

SEATTLE ve ATLANTA , birden fazla kümenin üyesi olacağı için, kümelerin adlarını içeren bir ad listesi oluşturmalısınız. Define the namelist on SEATTLE and ATLANTA , as follows:

```
DEFINE NAMELIST(CHAINMAIL)
DESCR('List of cluster names')
NAMES(CHNSTORE, MAILORDER)
```

6. SEATTLE ve ATLANTA üzerindeki CLUSTRVR kanallarını değiştirin.

SEATTLE ve ATLANTA 'daki CLUSTRVR kanal tanımlamaları, kanalların CHNSTORE kümesinde olduğunu gösterir. Küme alma kanalı tanımlamalarını, kanalların CHAINMAIL ad listesinde listelenen tüm kümelerde kullanılabilir olduğunu gösterecek şekilde değiştirin. SEATTLE' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.SEATTLE) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

ATLANTA' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

7. SEATTLE ve ATLANTA üzerindeki CLUSTSDR kanallarını değiştirin.

İki CLUSSDR kanal tanımlamalarını, ad listesini eklemek üzere değiştirin. SEATTLE' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

ATLANTA' da şu komutu girin:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

8. HARTFORD ve OMAHA üzerinde CLUSRCVR ve CLUSSDR kanallarını tanımlayın.

İki yeni kuyruk yöneticisi (HARTFORD ve OMAHA), küme alıcılığı ve küme göndericisi kanalları tanımlayın. Tanımları hangi sırada gerçekleştirdiğiniz önemli değildir. HARTFORD' da şunu girin:

```
DEFINE CHANNEL(MAILORDER.HARTFORD) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(HARTFORD.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for HARTFORD')
```



```
DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from HARTFORD to repository at CHICAGO')
```

OMAHA' da şunu girin:

```
DEFINE CHANNEL(MAILORDER.OMAHA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(OMAHA.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for OMAHA')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from OMAHA to repository at CHICAGO')
```

9. OMAHA kuyruğunda MORDERQ kuyruğunu tanımlayın.

The final step to complete this task is to define the queue MORDERQ on the queue manager OMAHA . OMAHA' da şunu girin:

```
DEFINE QLOCAL(MORDERQ) CLUSTER(MAILORDER)
```

10. Küme değişikliklerinin yayıldığını doğrulayın.

Önceki adımlarla yarattığınız tanımların küme olmasına karşın yayıldığını doğrulayın. Tam havuz kuyruk yöneticisinde aşağıdaki komutları verin:

```
DIS QCLUSTER (MORDERQ)
DIS CLUSQMGR
```

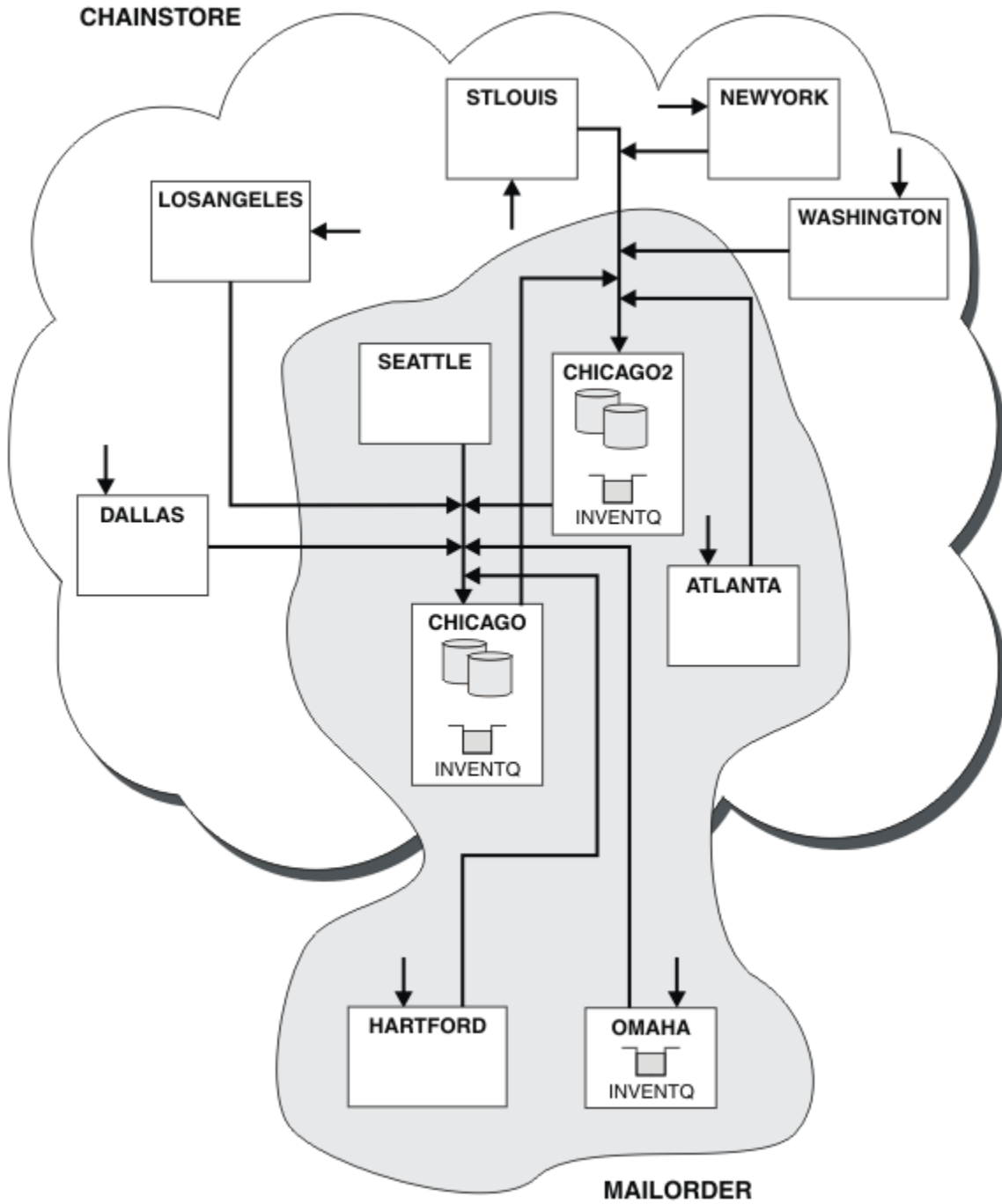
11.

Sonuçlar

Bu görev tarafından ayarlanan küme [Şekil 42 sayfa 234](#) içinde gösterilir.

Şimdi iki tane çakışan kümesimiz var. Her iki küme için de tam havuz CHICAGO ve CHICAGO2 konumunda tutulur. OMAHA üzerinde çalışan posta sırası uygulaması, CHICAGO' da çalışan döküm uygulamasından bağımsızdır. Ancak, CHNSTORE kümesindeki bazı kuyruk yöneticileri de MAILORDER kümesinde yer alıyor ve bu nedenle her iki uygulamaya da ileti gönderebilecekler. Bu görevi gerçekleştirmeden önce iki küme çakışması için kuyruk adı çakışmalarının olasılığını göz önünde bulundurun.

Suppose that on NEWYORK in cluster CHNSTORE and on OMAHA in cluster MAILORDER, there is a queue called ACCOUNTQ . Kümelerle çakışır ve ardından SEATTLE üzerindeki bir uygulama, ACCOUNTQ kuyruğuna bir ileti yerleştirirse, ileti ACCOUNTQ yönetim ortamına gidebilir.



Şekil 42. Birbirine bağlı kümeler

Sonraki adım

Suppose you decide to merge the MAILORDER cluster with the CHNSTORE cluster to form one large cluster called CHNSTORE.

To merge the MAILORDER cluster with the CHNSTORE cluster, such that CHICAGO and CHICAGO2 hold the full repositories:

- Alter the queue manager definitions for CHICAGO and CHICAGO2, removing the REPOSNL attribute, which specifies the namelist (CHAINMAIL), and replacing it with a REPOS attribute specifying the cluster name (CHNSTORE). For example:

```
ALTER QMGR(CHICAGO) REPOSNL(' ') REPOS(CHNSTORE)
```

- MAILORDER kümesindeki her kuyruk yöneticisinde, tüm kanal tanımlamalarını ve kuyruk tanımlamalarını değiştirerek, CLUSTER özneliğinin değerini MAILORDER değerinden CHNSTORE değerine değiştirin. Örneğin, HARTFORD konumunda şunu girin:

```
ALTER CHANNEL(MAILORDER.HARTFORD) CLUSTER(CHNSTORE)
```

OMAHA girişte:

```
ALTER QLOCAL(MORDERQ) CLUSTER(CHNSTORE)
```

- Alter all definitions that specify the cluster namelist CHAINMAIL, that is, the CLUSRCVR and CLUSSDR channel definitions at CHICAGO, CHICAGO2, SEATTLE, and ATLANTA , to specify instead the cluster CHNSTORE.

Bu örnekten, ad listelerini kullanmanın avantajını görebilirsiniz. CHICAGO ve CHICAGO2 için kuyruk yöneticisi tanımlamalarını değiştirmek yerine, ad listesi CHAINMAIL değerini değiştirebilirsiniz. Benzer şekilde, CHICAGO, CHICAGO2 , SEATTLE ve ATLANTA konumunda CLUSTVR ve CLUSSDR kanal tanımlamalarını değiştirmek yerine, ad listesini değiştirerek gereken sonucu elde edebilirsiniz.

Küme Ağının Kaldırılması

Bir ağdan bir kümeyi kaldırın ve dağıtılmış kuyruğa alma yapılandırmasını geri yükleyin.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- Bir IBM WebSphere MQ kümesi, [“Var olan bir ağı bir kümeye dönüştürme” sayfa 228](#) içinde açıklandığı şekilde ayarlanmıştır.
- Bu küme artık sistemden kaldırılacaktır. Kuyruk yöneticilerinin ağı, küme uygulanmadan önce olduğu gibi çalışmaya devam etmektedir.

Bu görev hakkında

Bir küme ağını kaldırmak için bu adımları izleyin.

Yordam

1. Küme kuyruklarını CHNSTORE kümesinden kaldırın.

On both CHICAGO and CHICAGO2, modify the local queue definition for the queue INVENTQ to remove the queue from the cluster. Komutu verin:

```
ALTER QLOCAL(INVENTQ) CLUSTER(' ')
```

Kuyruğu değiştirdiğinizde, tüm havuzlardaki bilgiler güncellenir ve küme boyunca yayılır. Active applications using MQ00_BIND_NOT_FIXED, and applications using MQ00_BIND_AS_Q_DEF where the queue has been defined with DEFBIND(NOTFIXED), fail on the next attempted MQPUT or MQPUT1 call. MQRC_UNKNOWN_OBJECT_NAME neden kodu döndürüldü.

Önce 1. adımı gerçekleştirmenize gerek yoktur, ancak 4. Adımdan sonra bunu gerçekleştirmezseniz, bunu gerçekleştirin.

2. Küme kuyruğuna erişimi olan tüm uygulamaları durdurun.

Küme kuyruklarına erişimi olan tüm uygulamaları durdurun. Bunu yapmazsanız, 5. Adımda kümeyi yenilediğinizde, bazı küme bilgileri yerel kuyruk yöneticisinde kalabilir. Tüm uygulamalar durdurulduğunda ve küme kanallarının bağlantısı kesildiğinde bu bilgiler kaldırılır.

3. Havuz özniteliğini tam havuz kuyruğu yöneticilerinden kaldırın.

Hem CHICAGO hem de CHICAGO2 üzerinde, havuz özniteliğini kaldırmak için kuyruk yöneticisi tanımlamalarını değiştirin. Bu işlemi yapmak için şu komutu verin:

```
ALTER QMGR REPOS(' ')
```

Kuyruk yöneticileri, kümedeki diğer kuyruk yöneticilerine, tam havuzları artık tutmadıkları konusunda bilgilendirir. Diğer kuyruk yöneticileri bu bilgileri aldıklarında, tam havuzun sona erdiğini belirten bir ileti görürsünüz. Ayrıca, CHNSTORE kümesi için artık kullanılabilir herhangi bir havuz olmadığını belirten bir ya da daha fazla ileti de görürsünüz.

4. Küme kanallarını kaldırın.

CHICAGO üzerinde küme kanallarını kaldırın:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) CLUSTER(' ')\nALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

Not: Önce CLUSSDR komutu, sonra CLUSTVR komutu vermek önemlidir. Önce CLUSRCVR komutunu girmeyin, sonra CLUSSDR komutunu girin. Bunu yapmak için, DURDURULDU durumu olan belirsiz kanallar yaratılır. Daha sonra, durdurulan kanalları kurtarmak için bir START CHANNEL komutu yayınlamaya gereksinim duyarsınız; örneğin, START CHANNEL(CHNSTORE.CHICAGO).

You see messages indicating that there are no repositories for the cluster CHNSTORE.

Adım 1 'de açıklandığı gibi küme kuyruklarını kaldırmadıysanız, şimdi yapın.

5. Küme kanallarını durdurun.

CHICAGO ' ta küme kanallarını aşağıdaki komutlarla durdurun:

```
STOP CHANNEL(CHNSTORE.CHICAGO2)\nSTOP CHANNEL(CHNSTORE.CHICAGO)
```

6. Kümedeki her kuyruk yöneticisi için adım 4 ve 5 'i yineleyin.

7. Küme kanallarını durdurun ve her bir kuyruk yöneticisinden küme kanallarına ve küme kuyruklarına ilişkin tüm tanımlamaları kaldırın.

8. İsteğe bağlı: Kuyruk yöneticisi tarafından tutulan, önbelleğe alınmış küme bilgilerini temizleyin.

Kuyruk yöneticileri artık kümenin üyesi olmasalar da, her biri kümeye ilişkin bilgilerin önbelleğe alınmış bir kopyasını saklar. If you want to remove this data, see task [“Kuyruk yöneticisini küme öncesi durumuna geri yükleme” sayfa 240.](#)

9. INVENTQ için uzak kuyruk tanımlamalarını değiştirin

Bu nedenle, ağ çalışmaya devam edebilir ve her kuyruk yöneticisinde INVENTQ için uzak kuyruk tanımlamasını değiştirin.

10. Kümeyi toparlayın.

Artık gerekli olmayan kuyruk ya da kanal tanımlamalarını silin.

Kuyruk Yöneticisinin Kümeden Kaldırılması

Kuyruk yöneticisinin kümede en az bir tam havuzla olağan şekilde iletişim kurabileceği senaryolarda, bir kuyruk yöneticisini bir kümeden kaldırın.

Başlamadan önce

Bu yöntem, en az bir tam havuzun kullanılabilir olduğu senaryolar için en iyi uygulamadır ve kaldırılmakta olan kuyruk yöneticisi ile iletişim kurulabilir. Bu yöntem, en az el ile yapılan müdahaleyi içerir ve kuyruk yöneticisinin kümeden denetimli bir geri çekilme kararlaştırabilmesini sağlar. Kaldırılmakta olan kuyruk yöneticisi tam bir havuzla iletişim kuramıyorsa, bkz. [“Bir kümeden kuyruk yöneticisi kaldırılıyor: Alternatif yöntem” sayfa 238](#).

Kuyruk yöneticisini kümeden kaldırmadan önce, kuyruk yöneticisinin kümenin gereksinim duyduğu kaynakları artık barındırmadığından emin olmanız gerekir:

- If the queue manager hosts a full repository, complete steps 1-4 from [“Tüm havuzu başka bir kuyruk yöneticisine taşıma” sayfa 226](#).
- Kuyruk yöneticisi küme kuyruklarını barındırorsa, [“Küme Kuyruğunun Kuyruk Yöneticisinden Kaldırılması” sayfa 224](#)’ tan 1-7 arasındaki adımları tamamlayın.
- Kuyruk yöneticisi küme konularını barındırorsa, konuları silin (örneğin, [DELETE TOPIC](#) komutunu kullanarak) ya da bunları diğer anasistemlere taşıyın.

Not: Bir kuyruk yöneticisini bir kümeden kaldırırsanız ve kuyruk yöneticisi hala bir küme konusunu barındırmaya devam ederse, kuyruk yöneticisi, konu silinceye kadar kümede kalan kuyruk yöneticilerine yayınları teslim etmeyi denemeye devam edebilir.

Bu görev hakkında

This example task removes the queue manager LONDON from the INVENTORY cluster. INVENTORY kümesi, [“Küme için kuyruk yöneticisi eklenmesi” sayfa 191](#)’de açıklandığı şekilde ayarlanır ve [“Küme Kuyruğunun Kuyruk Yöneticisinden Kaldırılması” sayfa 224](#)’de açıklandığı şekilde değiştirilir.

Kuyruk yöneticisini kümeden kaldırma işlemi, kuyruk yöneticisi ekleme işleminden daha karmaşıktır.

Bir kuyruk yöneticisi bir kümeye katıldığında, kümenin var olan üyelerinin yeni kuyruk yöneticisi hakkında bilgisi yoktur ve bu nedenle onunla hiçbir etkileşimleri yoktur. Yeni gönderici ve alıcı kanalları, tam bir havuza bağlanabilmesi için birleştirme kuyruğunda yöneticiye yaratılmalıdır.

Bir kuyruk yöneticisi bir kümeden kaldırıldığında, kuyruk yöneticisine bağlı uygulamaların, kümenin başka bir yerinde barındırılan kuyruklar gibi nesnelere kullanması olasıdır. Ayrıca, kümedeki diğer kuyruk yöneticilerine bağlı olan uygulamalar, hedef kuyruk yöneticisinde barındırılan nesnelere kullanıyor olabilir. Bu uygulamaların bir sonucu olarak, yürürlükteki kuyruk yöneticisi, kümeye katılmak için kullanılan tam havuz dışındaki küme üyeleriyle iletişim kurmak için ek gönderen kanalları yaratabilir. Kümedeki her kuyruk yöneticisinin, diğer küme üyelerini tanımlayan önbelleğe alınmış bir kopyasını vardır. Bu, kaldırılmakta olan birini içerebilir.

Yordam

1. Alter the manually defined cluster receiver channels to remove them from the cluster, on queue manager LONDON:

```
ALTER CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

2. Alter the manually defined cluster sender channels to remove them from the cluster, on queue manager LONDON:

```
ALTER CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) CLUSTER(' ')
```

Kümedeki diğer kuyruk yöneticileri, bu kuyruk yöneticisinin ve küme kaynaklarının artık kümenin bir parçası olmadığını öğrenir.

3. Monitor the cluster transmit queue, on queue manager LONDON, until there are no messages that are waiting to flow to any full repository in the cluster.

```
DISPLAY CHSTATUS(INVENTORY.LONDON) XQMSGSA
```

İletiler iletim kuyruğunda kalırsa, devam etmeden önce neden PARIS ve NEWYORK tam havuzlarına gönderilmediklerini belirleyin.

Sonuçlar

Kuyruk yöneticisi LONDON artık kümenin bir parçası değil. Ancak, bağımsız bir kuyruk yöneticisi olarak işlev görmeye devam edebilir.

Sonraki adım

Bu değişikliklerin sonucu, kümenin geri kalan üyesinde aşağıdaki komut verilerek doğrulanabilir:

```
DISPLAY CLUSQMgr(LONDON)
```

Kuyruk yöneticisi, otomatik olarak tanımlı küme gönderen kanalları durdurulana kadar görüntülenmeye devam eder. Bunun olmasını bekleyebilir ya da aşağıdaki komutu vererek etkin yönetim ortamları için izleme işlemine devam edebilirsiniz:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Bu kuyruk yöneticisine başka ileti gönderilmeyecek olduğundan eminseniz, kümenin geri kalan üyesinde aşağıdaki komutu vererek, küme gönderen kanallarını LONDON ' e durdurabilirsiniz:

```
STOP CHANNEL(INVENTORY.LONDON) STATUS(INACTIVE)
```

Değişiklikler küme boyunca yayıldıktan sonra, bu kuyruk yöneticisine başka ileti teslim edilmezse, LONDON üzerindeki CLUSRCVR kanalını durdurun ve silin:

```
STOP CHANNEL(INVENTORY.LONDON)  
DELETE CHANNEL(INVENTORY.LONDON)
```

Kaldırılan kuyruk yöneticisi, “Küme için kuyruk yöneticisi eklenmesi” sayfa 191’inde açıklandığı gibi daha sonraki bir noktada kümeye yeniden eklenebilir. Kaldırılan kuyruk yöneticisi, en fazla 90 gün boyunca kümenin kalan üyelerinin bilgilerini önbelleğe almak için devam eder. Bu önbellek sona erinceye kadar beklememeyi tercih ederseniz, “Kuyruk yöneticisini küme öncesi durumuna geri yükleme” sayfa 240’inde açıklandığı şekilde zorla çıkarılabilir.

Bir kümeden kuyruk yöneticisi kaldırılıyor: Alternatif yöntem

Önemli bir sistem ya da yapılandırma sorunu nedeniyle, kuyruk yöneticisi kümedeki tüm havuzlarla iletişim kuramadığından, bir kuyruk yöneticisini bir kümeden kaldırın.

Başlamadan önce

Bir kuyruk yöneticisini kümeden kaldırmanın bu alternatif yöntemi el ile durdurulur ve kaldırılan kuyruk yöneticisini kümeye bağlayan tüm küme kanallarını siler ve kuyruk yöneticisini kümeden zorla kaldırır. Bu yöntem, kaldırılmakta olan kuyruk yöneticisinin tüm havuzların hiçbirisiyle iletişim kuramayacağı senaryolarda kullanılır. Kuyruk yöneticisi çalışmayı durdurduğu için ya da kuyruk yöneticisiyle küme arasında uzun süreli bir iletişim hatası olduğu için bu olabilir (örneğin). Ters durumda, en sık kullanılan yöntemi kullanın: “Kuyruk Yöneticisinin Kümeden Kaldırılması” sayfa 236.

Kuyruk yöneticisini kümeden kaldırmadan önce, kuyruk yöneticisinin kümenin gereksinim duyduğu kaynakları artık barındırmadığından emin olmanız gerekir:

- If the queue manager hosts a full repository, complete steps 1-4 from “Tüm havuzu başka bir kuyruk yöneticisine taşıma” sayfa 226.
- Kuyruk yöneticisi küme kuyruklarını barındırorsa, “Küme Kuyruğunun Kuyruk Yöneticisinden Kaldırılması” sayfa 224’ tan 1-7 arasındaki adımları tamamlayın.

- Kuyruk yöneticisi küme konularını barındırsa, konuları silin (örneğin, `DELETE TOPIC` komutunu kullanarak) ya da bunları diğer anasistemlere taşıyın.

Not: Bir kuyruk yöneticisini bir kümeden kaldırırsanız ve kuyruk yöneticisi hala bir küme konusunu barındırmaya devam ederse, kuyruk yöneticisi, konu silininceye kadar kümede kalan kuyruk yöneticilerine yayınları teslim etmeyi denemeye devam edebilir.

Bu görev hakkında

This example task removes the queue manager LONDON from the INVENTORY cluster. INVENTORY kümesi, “Küme için kuyruk yöneticisi eklenmesi” sayfa 191’inde açıklandığı şekilde ayarlanır ve “Küme Kuyruğunun Kuyruk Yöneticisinden Kaldırılması” sayfa 224’inde açıklandığı şekilde değiştirilir.

Kuyruk yöneticisini kümeden kaldırma işlemi, kuyruk yöneticisi ekleme işleminden daha karmaşıktır.

Bir kuyruk yöneticisi bir kümeye katıldığında, kümenin var olan üyelerinin yeni kuyruk yöneticisi hakkında bilgisi yoktur ve bu nedenle onunla hiçbir etkileşimleri yoktur. Yeni gönderici ve alıcı kanalları, tam bir havuza bağlanabilmesi için birleştirme kuyruğunda yöneticiye yaratılmalıdır.

Bir kuyruk yöneticisi bir kümeden kaldırıldığında, kuyruk yöneticisine bağlı uygulamaların, kümenin başka bir yerinde barındırılan kuyruklar gibi nesnelere kullanması olasıdır. Ayrıca, kümedeki diğer kuyruk yöneticilerine bağlı olan uygulamalar, hedef kuyruk yöneticisinde barındırılan nesnelere kullanıyor olabilir. Bu uygulamaların bir sonucu olarak, yürürlükteki kuyruk yöneticisi, kümeye katılmak için kullanılan tam havuz dışındaki küme üyeleriyle iletişim kurmak için ek gönderen kanalları yaratabilir. Kümedeki her kuyruk yöneticisinin, diğer küme üyelerini tanımlayan önbelleğe alınmış bir kopyasını vardır. Bu, kaldırılmakta olan birini içerebilir.

Bu yordam, acil durumlarda, kuyruk yöneticisinin kümeyi incelikle bırakmasını beklemek olanaklı değilse, uygun bir durumda uygun olabilir.

Yordam

1. Kümedeki diğer kuyruk yöneticileriyle iletişim kurmak için kullanılan tüm kanalları durdurun. Use `MODE (FORCE)` to stop the `CLUSRCVR` channel, on queue manager LONDON. Tersi durumda, gönderenin kuyruk yöneticisinin kanalı durdurmasını beklemeniz gerekebilir:

```
STOP CHANNEL (INVENTORY.LONDON) MODE(FORCE)
STOP CHANNEL (INVENTORY.TORONTO)
STOP CHANNEL (INVENTORY.PARIS)
STOP CHANNEL (INVENTORY.NEWYORK)
```

2. Monitor the channel states, on queue manager LONDON, until the channels stop:

```
DISPLAY CHSTATUS (INVENTORY.LONDON)
DISPLAY CHSTATUS (INVENTORY.TORONTO)
DISPLAY CHSTATUS (INVENTORY.PARIS)
DISPLAY CHSTATUS (INVENTORY.NEWYORK)
```

Kanal durduktan sonra kümedeki diğer kuyruk yöneticilerine ya da diğer kuyruk yöneticilerine başka uygulama iletişi gönderilmez.

3. Delete the manually defined cluster channels, on queue manager LONDON:

```
DELETE CHANNEL (INVENTORY.NEWYORK)
DELETE CHANNEL (INVENTORY.TORONTO)
```

4. Kümedeki geri kalan kuyruk yöneticileri, kaldırılan kuyruk yöneticisine ilişkin bilgileri alkoymaya devam eder ve iletiye ileti göndermeye devam edebilir. Bilgi birikimi kalan kuyruk yöneticilerinden temizlemek için, kaldırılan kuyruk yöneticisini tüm havuzlardan birinde kümeden sıfırlayın:

```
RESET CLUSTER (INVENTORY) ACTION (FORCEREMOVE) QMNAME (LONDON) QUEUES (YES)
```

Kümede kaldırılan kuyruk yöneticisiyle aynı adı taşıyan başka bir kuyruk yöneticisi varsa, kaldırılan kuyruk yöneticisinin **QMID** değerini belirtin.

Sonuçlar

Kuyruk yöneticisi LONDON artık kümenin bir parçası değil. Ancak, bağımsız bir kuyruk yöneticisi olarak işlev görmeye devam edebilir.

Sonraki adım

Bu değişikliklerin sonucu, kümenin geri kalan üyesinde aşağıdaki komut verilerek doğrulanabilir:

```
DISPLAY CLUSQMGR(LONDON)
```

Kuyruk yöneticisi, otomatik olarak tanımlı küme gönderen kanalları durdurulana kadar görüntülenmeye devam eder. Bunun olmasını bekleyebilir ya da aşağıdaki komutu vererek etkin yönetim ortamları için izleme işlemine devam edebilirsiniz:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Değişiklikler küme boyunca yayılır ve bu kuyruk yöneticisine başka ileti teslim edilmezse, LONDON' taki CLUSRCVR kanalını silin:

```
DELETE CHANNEL(INVENTORY.LONDON)
```

Kaldırılan kuyruk yöneticisi, “[Küme için kuyruk yöneticisi eklenmesi](#)” sayfa 191’inde açıklandığı gibi daha sonraki bir noktada kümeye yeniden eklenebilir. Kaldırılan kuyruk yöneticisi, en fazla 90 gün boyunca kümenin kalan üyelerinin bilgilerini önbelleğe almak için devam eder. Bu önbellek sona erinceye kadar beklememeyi tercih ederseniz, “[Kuyruk yöneticisini küme öncesi durumuna geri yükleme](#)” sayfa 240’inde açıklandığı şekilde zorla çıkarılabilir.

Kuyruk yöneticisini küme öncesi durumuna geri yükleme

Bir kuyruk yöneticisi bir kümeden kaldırıldığında, kalan küme üyeleri hakkında bilgi saklayacaktır. Bu bilgi sonunda sona erer ve otomatik olarak silinir. Ancak, hemen silmeyi tercih ederseniz, bu konudaki adımları kullanabilirsiniz.

Başlamadan önce

Kuyruk yöneticisinin kümeden kaldırıldığı ve artık kümede herhangi bir çalışma gerçekleştirmediği varsayılır. Örneğin, kuyrukları artık kümeden ileti almıyor ve hiçbir uygulama bu kuyruklara ulaşmasını bekleyen hiçbir uygulama beklemiyor.

Önemli: Bir kuyruk yöneticisini bir kümeden kaldırırsanız ve REPOS (YES) kullanarak yenilediyeniz, CLUSTRVR ' ın CLUSTER özniteliğini değiştirdiğinizde, bunu yeniden ekleyemeyeceksiniz. CLUSRCVR ' nin CLUSTER özniteliğini boş olmayan (yani, kümesiad) değiştirdikten sonra, REPOS (NO) ile yenileme kümesini de yayınlamaya gereksinim duyarsınız. Bu noktada, CLUSRCVR üzerindeki iç sıra numaraları güncel olarak getirilecektir. Daha sonra kuyruk yöneticisi, kendisini Tam Havuzlar ve küme üyelerinin geri kalanına yeniden tanıtırken başarılı olur. (CLUSTRVR kanalından doğru küme adı verildikten sonra, komutun REPOS (NO) sürümünün çalıştırılması gerektiğini unutmayın.)

Bu kısıtlama yalnızca IBM WebSphere MQ Version 7.5 için geçerlidir.

Bu görev hakkında

Bir kuyruk yöneticisi kümeden kaldırıldığında, 90 güne kadar kalan küme üyeleri hakkında bilgi sahibi olmaya devam eder. Özellikle kuyruk yöneticisi kümeyi hızlı bir şekilde yeniden birleştirirse, bunun sistem yararları olabilir. Bu bilgi sonunda süresi dolduğunda, otomatik olarak silinir. Ancak, bu bilgileri el ile silmeyi tercih etme nedenleri de vardır. Örneğin:

- Daha önce küme kaynaklarını kullanan bu kuyruk yöneticilerindeki her uygulamayı durdurduğunu doğrulamak isteyebilirsiniz. Kalan küme üyelerinin bilgileri sona erinceye kadar, bu tür uygulamalar bir iletim kuyruğuna yazmaya devam eder. Küme bilgisi silindikten sonra, bu tür bir uygulama küme kaynaklarını kullanmayı denediğinde sistem bir hata iletişi oluşturur.
- Kuyruk yöneticisine ilişkin durum bilgilerinizi görüntülediğinizde, kalan küme üyeleri hakkında süresi dolan bilgileri görmemeyi tercih edebilirsiniz.

Bu görev, örnek olarak INVENTORY kümesini kullanır. LONDON kuyruk yöneticisi, “[Kuyruk Yöneticisinin Kümeden Kaldırılması](#)” sayfa 236’de açıklandığı şekilde INVENTORY kümesinden kaldırılmıştır. Kümenin geri kalan üyelerinin bilgilerini silmek için, LONDON kuyruk yöneticisine aşağıdaki komutları verin.

Yordam

1. Bu kuyruk yöneticisinden kümedeki diğer kuyruk yöneticilerine ilişkin tüm belleği kaldırın:

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

2. Tüm küme kaynakları kayboluncaya kadar kuyruk yöneticisini izleyin:

```
DISPLAY CLUSQMR(*) CLUSTER(INVENTORY)
DISPLAY QCLUSTER(*) CLUSTER(INVENTORY)
DISPLAY TOPIC(*) CLUSTER(INVENTORY)
```

İlgili kavramlar

Kümeler

“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

“Bir kümenin bileşenleri” sayfa 161

Kümeler, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

“IBM WebSphere MQ kümelerini yönetme” sayfa 181

IBM WebSphere MQ kümelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

Kuyruk yöneticisinin bakımı

Bakım gerçekleştirmek için bir kuyruk yöneticisinden bir kuyruk yöneticisini askıya alın ve sürdürün.

Bu görev hakkında

Zaman zaman, kümenin bir parçası olan bir kuyruk yöneticisinden bakım gerçekleştirmeniz gerekebilir. Örneğin, kuyruklarındaki verilerin yedeklerini almak ya da düzeltmeleri yazılıma uygulamak için gerekebilir. Kuyruk yöneticisi kuyrukları barındırıyorsa, etkinliklerinin askıya alınması gerekir. Bakım işlemi tamamlandığında, etkinliklerine devam edilebilir.

Yordam

1. Bir kuyruk yöneticisini askıya almak için SUSPEND QMGR **runmqsc** komutunu kullanın:

```
SUSPEND QMGR CLUSTER(SALES)
```

SUSPEND **runmqsc** komutu, bu kuyruk yöneticisinin askıya alındığını SALES kümesindeki kuyruk yöneticilerine bildirir.

SUSPEND QMGR komutunun amacı, yalnızca diğer kuyruk yöneticilerine, mümkünse bu kuyruk yöneticisine ileti göndermekten kaçınmalarını önermek için kullanılabilir. Bu, kuyruk yöneticisinin devre dışı bırakıldığı anlamına gelmez. Bu kuyruk yöneticisi tarafından işlenmesi gereken bazı iletiler, örneğin, bu kuyruk yöneticisi kümelenecek bir kuyruğun tek anasistemiye, bu iletilerin gönderilmesine neden olur.

Kuyruk yöneticisi, iş yükü yönetimi yordamlarını askıya almışken, bu yordama ileti göndermekten kaçınılabiliyor. Bu kuyruk yöneticisi tarafından işlenmesi gereken iletiler, yerel kuyruk yöneticisi tarafından gönderilen iletileri içerir.

WebSphere MQ , mümkün olduğunda yerel kuyruk yöneticisini seçmek yerine hangi hedeflere uygun olduğunu belirlemek için bir iş yükü dengeleme algoritması kullanır.

- a) Enforce the suspension of a queue manager by using the FORCE option on the SUSPEND QMGR command:

```
SUSPEND QMGR CLUSTER(SALES) MODE(FORCE)
```

MODE (FORCE) , kümedeki diğer kuyruk yöneticilerinden gelen tüm gelen kanalları zorla durdurur. MODE (FORCE) belirtmezseniz, varsayılan MODE (QUIESCE) uygulanır.

2. Hangi bakım görevlerinin gerekli olduğunu yapın.

3. Resume the queue manager by issuing the RESUME QMGR **runmqsc** command:

```
RESUME QMGR CLUSTER(SALES)
```

Sonuçlar

RESUME **runmqsc** komutu, kuyruk yöneticisinin yeniden kullanılabilir olduğunu tam havuzlara bildirir. Tüm havuz kuyruğu yöneticileri bu bilgileri, bu kuyruk yöneticisine ilişkin bilgileri isteyen diğer kuyruk yöneticilerine dağıtabiliyor.

Küme iletim kuyruğunun korunması

Küme iletim kuyruklarını kullanılabilir tutmak için her türlü çabayı gösteriniz. Bunlar, kümelerin performansı için gereklidir.

Başlamadan önce

- Küme iletim kuyruğunun dolu olmamasını denetleyin.
- Bir ALTER **runmqsc** komutunu, devre dışı bırakılacak ya da yanlışlıkla devre dışı bırakılacak şekilde ayarlamaya özen göstermeyin.
- cluster' ta küme iletim kuyruğunun saklandığından emin olun .

Küme Kuyruk Yöneticisinin Yenilenmesi

REFRESH CLUSTER komutunu kullanarak, otomatik olarak tanımlanmış kanalları ve otomatik olarak tanımlanmış küme nesnelere yerel havuzdan kaldırabilirsiniz. Hiçbir ileti kaybolmaz.

Başlamadan önce

You might be asked to use the command by your IBM Support Center. Dikkatli bir şekilde dikkate alınmadan komutu kullanmayın. Örneğin, **REFRESH CLUSTER** komutunun büyük kümeler için kullanımı, devam etmekte olduğu sürece kümeden kesintiye neden olabilir ve daha sonra, küme nesnelere otomatik olarak ilgili tüm kuyruk yöneticilerine durum güncellemeleri gönderdiğinde, 27 gün aralıklarla küme için kesintiye neden olabilir. Bkz. [“Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma” sayfa 299.](#)

Bu görev hakkında

Kuyruk yöneticisi bir kümede yeni bir başlangıç yapabilir. Olağan koşullarda, REFRESH CLUSTER komutunu kullanmanız gerekmez.

Yordam

Otomatik olarak tanımlanmış küme kuyruğu yöneticisini ve kuyruk nesnelere yerel havuzdan kaldırmak için bir kuyruk yöneticisinden REFRESH CLUSTER **MQSC** komutunu verin.

Komut yalnızca diğer kuyruk yöneticilerine gönderme yapan nesnelere kaldırır, yerel kuyruk yöneticisiyle ilgili nesnelere kaldırmaz. Komut, otomatik olarak tanımlanmış kanalları da kaldırır. Küme iletim kuyruğunda iletilmeyen kanalları kaldırır ve tam havuz kuyruğu yöneticisine bağlanmaz.

Sonuçlar

Effectively, the REFRESH CLUSTER command allows a queue manager to be cold-started with respect to its full repository content. IBM WebSphere MQ , kuyruklarınızdan veri kaybolmamasını sağlar.

İlgili kavramlar

“Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma” sayfa 299

Bir kümeye ilişkin olarak yerel olarak tutulan tüm bilgileri atmak ve bu bilgileri kümedeki tüm havuzlardan yeniden oluşturmak için **REFRESH CLUSTER** komutunu kullanıyorsunuz. Bu komutu, istisnai koşullar dışında, kullanmanıza gerek yoktur. Bunu kullanmanız gerekiyorsa, bunu nasıl kullanmanız gerektiği konusunda dikkat edilmesi gereken noktalar vardır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuздur.

Kuyruk yöneticisinin kurtarılıyor

Bring the cluster information about a queue manager up to date using the REFRESH CLUSTER **runmqsc** command. Bir kuyruk yöneticisini anlık yedeklemeden kurtardıktan sonra bu yordamı izleyin.

Başlamadan önce

Bir küme kuyruk yöneticisini anlık yedeklemeden geri yüklediniz.

Bu görev hakkında

To recover a queue manager in a cluster, restore the queue manager, and then bring the cluster information up to date using the REFRESH CLUSTER **runmqsc** command.

Not: Büyük kümeler için, **REFRESH CLUSTER** komutunun kullanımı devam ederken kümeyi kesintiye uğratabilir ve bundan sonra 27 gün aralıklarla küme nesnelere, ilgili tüm kuyruk yöneticilerine otomatik olarak durum güncellemeleri gönderdiğinde, bu işlem yine 27 gün aralıklarla kesintiye uğrayabilir. Bkz. [Büyük bir kümede yenilenme, kümenin performansını ve kullanılabilirliğini etkileyebilir.](#)

Yordam

Geri yüklenen kuyruk yöneticisinde, kuyruk yöneticisinin katıldığı tüm kümeler için REFRESH CLUSTER komutunu verin.

Sonraki adım

Başka bir kuyruk yöneticisinde REFRESH CLUSTER komutunu yayınlamaya gerek yoktur.

İlgili kavramlar

“Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma” sayfa 299

Bir kümeye ilişkin olarak yerel olarak tutulan tüm bilgileri atmak ve bu bilgileri kümedeki tüm havuzlardan yeniden oluşturmak için **REFRESH CLUSTER** komutunu kullanıyorsunuz. Bu komutu, istisnai koşullar dışında, kullanmanıza gerek yoktur. Bunu kullanmanız gerekiyorsa, bunu nasıl kullanmanız gerektiği konusunda dikkat edilmesi gereken noktalar vardır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuздur.

Kullanılabilirlik için küme kanallarının yapılandırılması

Aralıklı ağ süreölçerleri varsa, küme kanallarının sorunsuz bir şekilde çalışmasını sürdürürebilmek için iyi yapılandırma uygulamalarını izleyin.

Başlamadan önce

Kümelere, kanal tanımlama gereksinimi gidermenizi sağlar, ancak yine de bunları korumanız gerekir. Aynı kanal teknolojisi, dağıtılmış kuyruğa alma sırasında kullanılan bir kümedeki kuyruk yöneticileri arasındaki iletişim için kullanılır. Küme kanallarını anlamak için aşağıdakilerle ilgili bilgi sahibi olmanız gerekir:

- Kanalların çalışması
- Durumlarının nasıl bulunması
- Kanal çıkışlarının kullanılması

Bu görev hakkında

Aşağıdaki noktalara özel bir önem vermek isteyebilirsiniz:

Yordam

Küme kanallarını yapılandırırken aşağıdaki noktaları göz önünde bulundurun

- Ağa yük olmayan ya da canlı tutma akışları içeren, küme gönderici kanallarında ve küme alıcı kanallarında HBNT ya da KAJNT için değer seçin. Ağınız bazen yavaşlarsa ve bu uzunluğun gecikmelerini tanıyor, 10 saniyeden daha az bir aralık yanlış hata verir.
- Set the BATCHHB value to reduce the window for causing a marooned message because it is indoubt on a failed channel. Kümenin doldurulması daha uzun süre verilirse, başarısız olan bir kanalda belirsiz toplu iş daha çok oluşabilir. Kanal boyunca gelen ileti trafiği, uzun süre boyunca, ileti patlamaları arasında başarısız olan bir toplu işin daha büyük bir olasılıkla ortaya çıkıyorsa, bu durum daha yüksek olur.
- Bir kanalın küme gönderen bitişi başarısız olursa ve sağlıklı işletim bildirim ya da canlı tutma başarısız olduğunda yeniden başlatmayı denediğinde sorun ortaya çıkar. Kanalın küme alıcı ucu etkin kalıyorsa, kanal-gönderici yeniden başlatma işlemi reddedilir. Başarısızlığı önlemek için, kümeli alıcı kanalının bir küme gönderici kanalı yeniden başlatmayı denediğinde sonlandırılacak ve yeniden başlatılacak şekilde ayarlayın.

z/OS'deki platformlarda

Control the problem of the cluster-receiver end of the channel remaining active using the AdoptNewMCA, AdoptNewMCATimeout, and AdoptNewMCACheck attributes in the qm.ini file or the Windows NT Registry.

İletileri kümeden ve kümeden yönlendirme

Kümelere dış kuyruk yöneticilerine ve diğer kümelere bağlamak için kuyruk diğer adlarını, kuyruk yöneticisi diğer adlarını ve uzak kuyruk tanımlamalarını kullanın.

İletilerin kümelere yönelmesine ilişkin ayrıntılar için aşağıdaki alt başlıklara bakın:

İlgili kavramlar

Kümelere

Kümelere nasıl çalışır

“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

“Bir kümenin bileşenleri” sayfa 161

Kümelere, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

“IBM WebSphere MQ kümelerini yönetme” sayfa 181

IBM WebSphere MQ kümelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

“Kuyruk yöneticisi diğer adları ve kümelere” sayfa 255

Bir kümeye ileti gönderirken ya da bir kümeye gönderilen iş yükü dengesi iletilerine ileti gönderirken kuyruk yöneticilerinin adını gizlemek için kuyruk yöneticisi diğer adlarını kullanın.

“Kuyruk diğer adları ve kümeleri” sayfa 257

Bir küme kuyruğunun adını gizlemek, bir kuyruğu kümelemek, farklı öznitelikleri benimsemek ya da farklı erişim denetimlerini benimsemek için kuyruk diğer adlarını kullanın.

“Yanıt-kuyruk diğer adları ve kümeleri için” sayfa 257

Yanıt kuyruğu diğer adı tanımlaması, yanıt bilgileri için diğer adları belirtmek için kullanılır. Yanıtın gönderileceği kuyruk diğer adı tanımlamaları, dağıtılmış bir kuyruklama ortamında olduğu gibi kümelerle kullanılabilir.

İlgili görevler

“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

“Yeni bir küme ayarlanıyor” sayfa 181

Örnek kümeyi ayarlamak için bu yönergeleri izleyin. Ayrı yönergeler, TCP/IP, LU 6.2 üzerinde ve tek bir iletim kuyruğu ya da birden çok iletim kuyruğundaki kümeyi ayarlayıp tanımlamalarını açıklar. Bir kuyruk yöneticisinden diğerine ileti göndererek, kümenin çalıştığını test edin.

İstekte bulunanın/yanıtının bir kümeye ilişkin yapılandırılması

Bir küme yöneticisi dışında bir kuyruk yöneticisinden bir istek/yanıt iletisi yolu yapılandırın. Bir ağ geçidi kuyruk yöneticisini kümeden ve kümeden iletişim yolu olarak kullanarak, kümenin iç ayrıntılarını gizleyebilirsiniz.

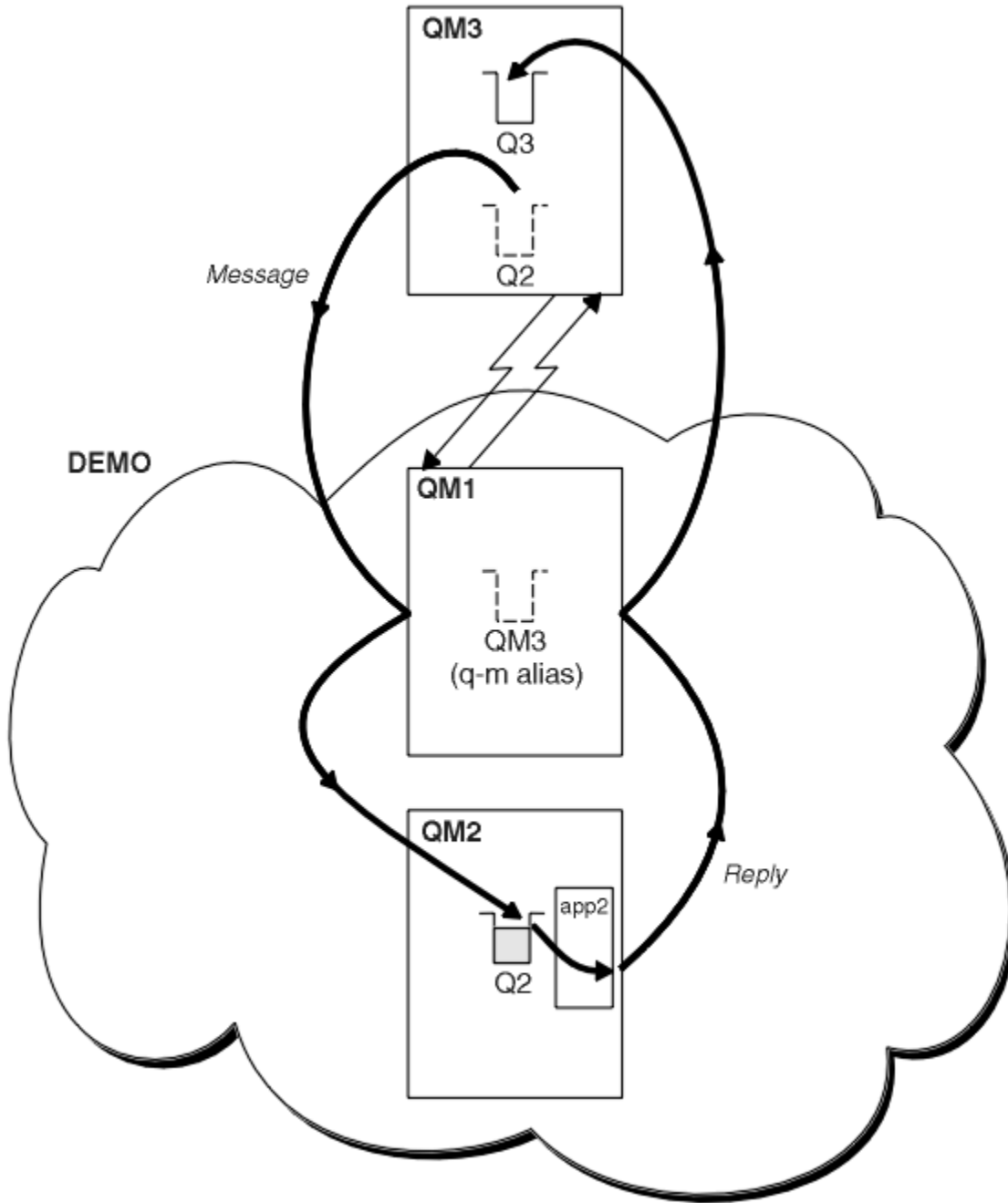
Başlamadan önce

Şekil 43 sayfa 246 shows a queue manager called QM3 that is outside the cluster called DEMO. QM3 , kümeleri desteklemediği bir WebSphere MQ ürününde bir kuyruk yöneticisi olabilir. QM3 , aşağıdaki gibi tanımlanan Q3adlı bir kuyruğu barındırır:

```
DEFINE QLOCAL(Q3)
```

Kümenin içinde, QM1 ve QM2adlı iki kuyruk yöneticisi vardır. QM2 hosts a cluster queue called Q2, which is defined as follows:

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO)
```



Şekil 43. Kuyruk yöneticisinden kümenin dışında konması

Bu görev hakkında

İstek ve yanıt iletileri yolunu ayarlamak için yordamda bulunan öneriyi izleyin.

Yordam

1. İstek iletilerini kümeye gönderin.

Consider how the queue manager that is outside the cluster puts a message to the queue Q2 at QM2, that is inside the cluster. Kümenin dışında bir kuyruk yöneticisinin, ileti koyduğu kümedeki her kuyruk için bir QREMOTE tanımlaması olmalıdır.

- a) QM3 üzerinde Q2 için bir uzak kuyruk tanımlayın.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(QM2) XMITQ(QM1)
```

QM3 , bir kümenin parçası olmadığından, dağıtılmış kuyruklama tekniklerini kullanarak iletişim kurmalıdır. Bu nedenle, aynı zamanda bir gönderici kanalı ve QM1' e bir iletim kuyruğu da olmalıdır. QM1 , karşılık gelen bir alıcı kanalına gerek duyar. Kanallar ve iletim kuyrukları Şekil 43 sayfa 246'inde açık bir şekilde gösterilmiyor.

In the example, an application at QM3 issues an MQPUT call to put a message to Q2. The QREMOTE definition causes the message to be routed to Q2 at QM2 using the sender-channel that is getting messages from the QM1 transmission queue.

2. Kümeden yanıt iletisini alın.

Bir kuyruk yöneticisi diğer adı kullanarak, küme dışındaki bir kuyruk yöneticisine yanıt için bir dönüş yolu yaratın. The gateway, QM1, advertises a queue-manager alias for the queue manager that is outside the cluster, QM3. Küme özneliğini, QM3 için bir kuyruk yöneticisi diğer adı tanımlamasına ekleyerek, kümenin içindeki kuyruk yöneticilerine QM3 reklamı yapılıyor. Kuyruk yöneticisi diğer adı tanımlaması uzak bir kuyruk tanımlaması gibi, ancak boş bir RNAME ile.

a) QM1 üzerinde QM3 için bir kuyruk yöneticisi diğer adı tanımlayın.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

We must consider the choice of name for the transmission queue used to forward replies back from QM1 to QM3. QREMOTE tanımlamasındaki örtük, XMITQ özneliğinin omission ile, iletim kuyruğunun adı QM3 olur. Ancak, kuyruk yöneticisi diğer adını kullanarak kümenin geri kalanına reklam vermek için beklediğimiz adla aynı ad QM3 . WebSphere MQ , hem iletim kuyruğunu, hem de kuyruk yöneticisi diğer adını aynı adı vermenize izin vermez. Bir çözüm, iletileri kuyruk yöneticisi diğer adına farklı bir adla QM3 ' e iletmek için bir iletim kuyruğu oluşturmaktan.

b) Provide the transmission queue name in the QREMOTE definition.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO) XMITQ(QM3.XMIT)
```

Yeni kuyruk yöneticisi diğer adı, newadlı yeni iletim kuyruğunun QM3 kuyruk yöneticisi diğer adıyla QM3 . XMIT adlı çiftlerini içerir. Bu basit ve doğru bir çözümdür, ancak tamamen tatmin edici değildir. Hedef kuyruk yöneticisiyle aynı adı taşıyan, iletim kuyrukları için adlandırma kuralını bozmuştur. İletim kuyruğunda adlandırma kuralını koruyan başka çözümler var mı?

The problem arises because the requester defaulted to passing QM3 as the reply-to queue manager name in the request message that is sent from QM3. The server on QM2 uses the QM3 reply-to queue manager name to address QM3 in its replies. The solution required QM1 to advertise QM3 as the queue manager alias to return reply messages to and prevented QM1 from using QM3 as the name of the transmission queue.

Yanıt kuyruğu yöneticisi adı olarak QM3 değerini sağlamak yerine, QM3 üzerindeki uygulamaların yanıt iletileri için bir yanıt kuyruk yöneticisi diğer adı QM1 olarak iletilmesi gerekir. Ağ geçidi kuyruk yöneticisi QM1 , yanıt için kuyruk yöneticisi diğer adını QM3 yerine QM3 yerine, iletim kuyruğu adıyla çakışmayı önleyerek advertises' a bildirir.

c) QM1 üzerinde QM3 için bir kuyruk yöneticisi diğer adı tanımlayın.

```
DEFINE QREMOTE(QM3.ALIAS) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

Yapılandırma komutlarında iki değişiklik yapılması gerekir.

- i) The QREMOTE at QM1 now advertises our queue manager alias QM3 . ALIAS to the rest of the cluster, coupling it to the name of the real queue manager QM3. QM3 , yanıt kuyruklarını QM3' a geri göndermek için yine iletim kuyruğunun adıdır.
- ii) İstemci uygulaması, istek iletisini oluştururken, yanıt kuyruğu yöneticisinin adı olarak QM3 . ALIAS sağlamalıdır. QM3 . ALIAS ' u istemci uygulamasına iki yoldan birini sağlayabilirsiniz.
 - Code QM3 . ALIAS in the reply-to queue manager name field constructed by MQPUT in the MQMD. Yanıtlar için dinamik bir kuyruk kullanıyorsanız, bunu bu şekilde yapmanız gerekir.

- Use a reply-to queue alias, Q3 . ALIAS, rather than a reply-to queue when providing the reply-to queue name.

```
DEFINE QREMOTE(Q3 . ALIAS) RNAME(Q3) RQMNAME(QM3 . ALIAS)
```

Sonraki adım

Not: AMQSREQ0 ile yanıtlama kuyruğu diğer adlarının kullanımını gösteremezsiniz. Yanıt kuyruğunda, parametre 3 'te sağlanan kuyruk adı ya da varsayılan SYSTEM . SAMPLE . REPLY model kuyruğu kullanılarak kuyruğa alma işlemi açılır. Örneğin, yanıt kuyruğu diğer adını içeren başka bir değiştirge sağlayan örneği, MQPUT için yanıt kuyruğu yöneticisi diğer adını belirterek değiştirmeniz gerekir.

İlgili görevler

[“Küme hedef kuyruk yöneticisinin adının gizlenmesi” sayfa 248](#)

Bir iletiyi, kuyruk yöneticisini adlandırmadan, kümedeki herhangi bir kuyruk yöneticisinde tanımlı olan bir küme kuyruğuna yöneltir.

Küme hedef kuyruk yöneticisinin adının gizlenmesi

Bir iletiyi, kuyruk yöneticisini adlandırmadan, kümedeki herhangi bir kuyruk yöneticisinde tanımlı olan bir küme kuyruğuna yöneltir.

Başlamadan önce

- Kümenin içinde olan kuyruk yöneticilerinin adlarını kümenin dışındaki kuyruk yöneticilerine açıklamaktan kaçının.
 - Küme içindeki bir kuyruğu barındıran kuyruk yöneticisine yönelik başvuruların çözülmesi, iş yükü dengelemesi yapma esnekliğini ortadan kaldırır.
 - Ayrıca, kümede kuyruğun bulunduğu bir kuyruk yöneticisini değiştirmeniz de zorlaşmanızı sağlar.
 - Diğer seçenek, küme denetimcisi tarafından sağlanan bir kuyruk yöneticisi diğer adı ile RQMNAME ' in yerine geçmeniz.
 - [“Küme hedef kuyruk yöneticisinin adının gizlenmesi” sayfa 248](#) , kuyruk yöneticisi dışındaki bir kuyruk yöneticisini bir kümenin içindeki kuyruk yöneticilerinin yönetiminden bir kümenin dışında kesmek için kuyruk yöneticisi diğer adını kullanarak tanımlar.
- Ancak, iletim kuyruklarının adlarına ilişkin önerilen yol, onlara hedef kuyruk yöneticisinin adını vermesidir. İletim kuyruğunun adı, kümedeki bir kuyruk yöneticisinin adını gösterir. Hangi kuralı takip etmek gerektiğini seçmelisin. İletim kuyruğunun adını kuyruk yöneticisi adını ya da küme adını kullanarak adlamayı seçebilirsiniz:

Ağ geçidi kuyruk yöneticisi adını kullanarak iletim kuyruğunu adı

Ağ geçidi kuyruk yöneticisi adının bir kümenin dışındaki kuyruk yöneticilerine açıklanması, küme kuyruk yöneticisi adlarının gizlenmesi kuralı için makul bir kural dışı durum sağlar.

İletim kuyruğunun adını, kümenin adını kullanarak yazın.

Hedef kuyruk yöneticisi adı ile iletim kuyrukları adlandırma kuralını izlemiyorsanız, küme adını kullanın.

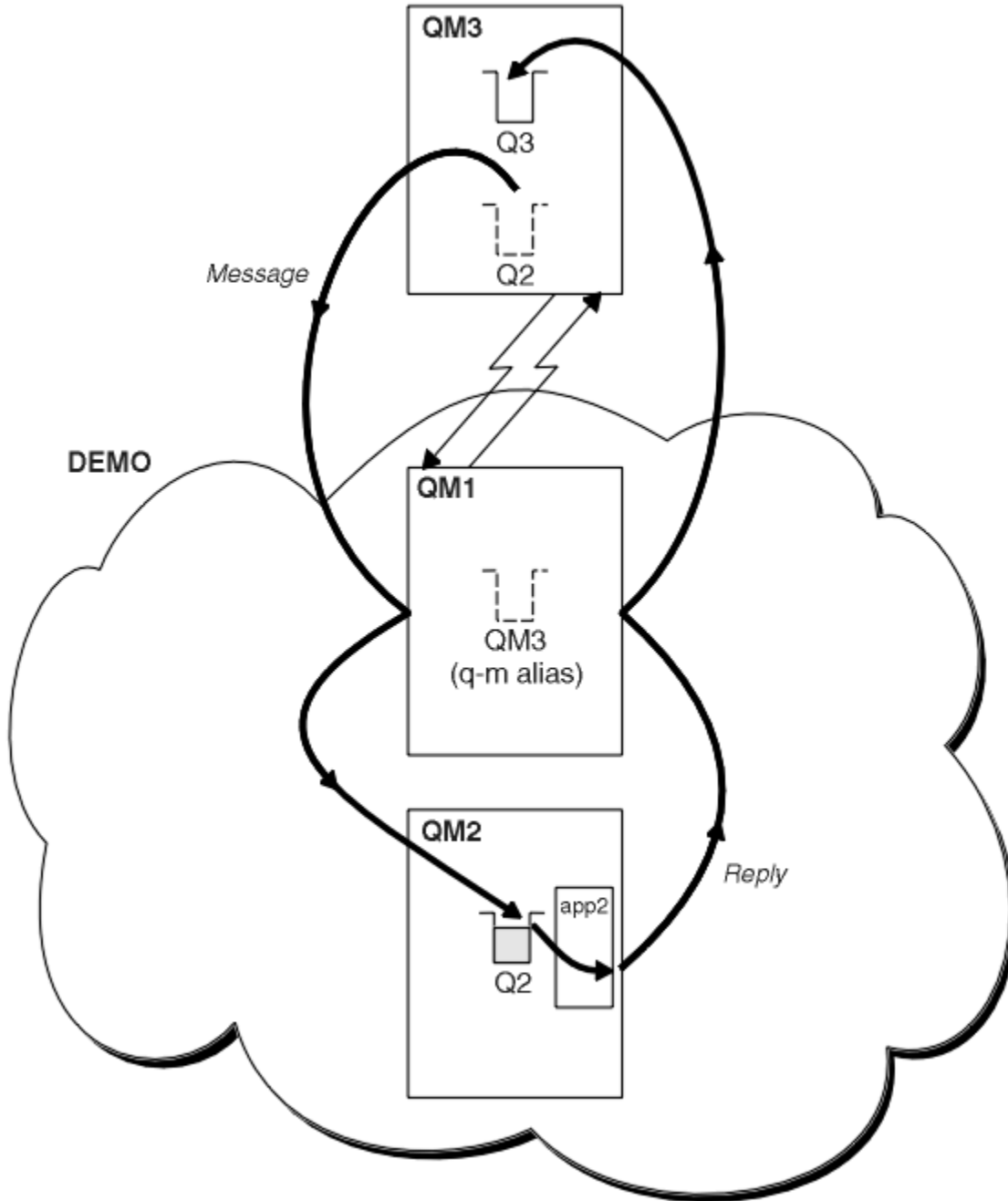
Bu görev hakkında

Modify the task [“İstekte bulunanın/yanıtının bir kümeye ilişkin yapılandırılması” sayfa 245](#), to hide the name of the target queue manager inside the cluster.

Yordam

In the example, see [Şekil 44 sayfa 249](#) , define a queue manager alias on the gateway queue manager QM1 called DEMO:


```
DEFINE QREMOTE(DEMO) RNAME(' ') RQMNAME(' ')
```



Şekil 44. Kuyruk yöneticisinden kümenin dışında konması

The QREMOTE definition on QM1 makes the queue manager alias DEMO known to the gateway queue manager. QM3, the queue manager outside the cluster, can use the queue manager alias DEMO to send messages to cluster queues on DEMO, rather than having to use an actual queue manager name.

Bir kümeye bağlanan iletim kuyruğunu adlamak için küme adını kullanma kuralını kabul etseniz, Q2 için uzak kuyruk tanımı şu hale gelir:

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(DEMO)
```

Sonuçlar

DEMO üzerinde Q2 için yazılmış iletiler, DEMO iletim kuyruğuna yerleştirilir. Gönderici kanalın aktardıkları iletim kuyruğundan, gatewayağ geçidi kuyruk yöneticisine (QM1) aktarılır. Ağ geçidi kuyruk yöneticisi, iletileri, Q2küme kuyruğunu barındıran kümedeki herhangi bir kuyruk yöneticisine yönlendirir.

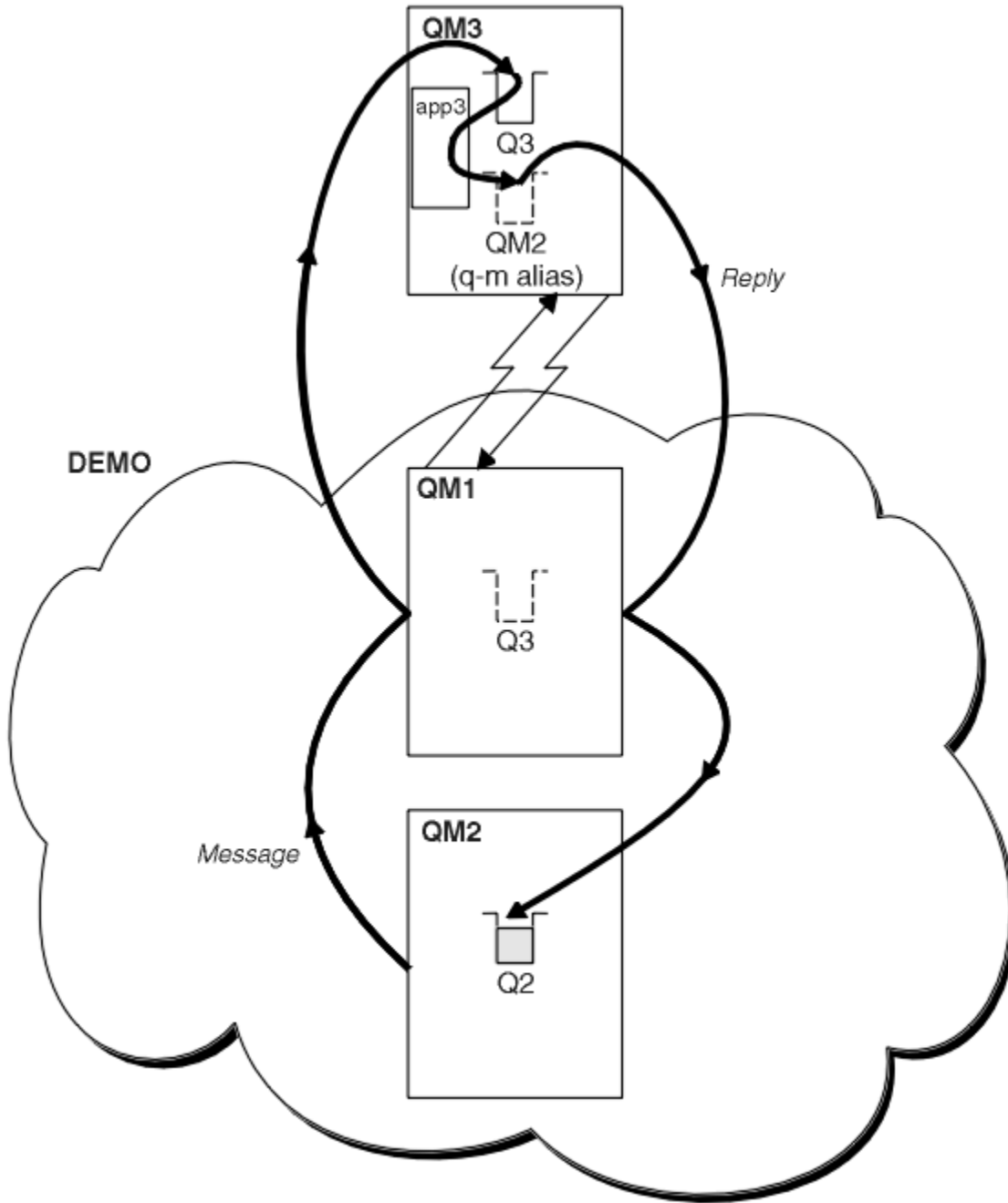
Bir kümeden request/reply yapılandırılıyor

Kümeden bir kuyruk yöneticisine bir istek/yanıt iletisi yolunu kümeden yapılandırın. Bir ağ geçidi kuyruk yöneticisi kullanarak, küme içindeki bir kuyruk yöneticisinin kümenin dışında nasıl iletişim kurduyla ilgili ayrıntıları gizleyebilirsiniz.

Başlamadan önce

Şekil 45 sayfa 251 shows a queue manager, QM2, inside the cluster DEMO. Küme dışında kuyruk yöneticisine barındırılan Q3kuyruğuna bir istek gönderir. Yanıtlar, kümenin içinde QM2 konumundaki Q2 ' e döndürülür.

Küme dışında kuyruk yöneticisiyle iletişim kurmak için, kümenin içindeki bir ya da daha çok kuyruk yöneticisi ağ geçidi olarak işlev görmektedir. Ağ geçidi kuyruk yöneticisinin, küme dışındaki kuyruk yöneticilerine ilişkin bir iletişim yolu vardır. Örnekte QM1 , ağ geçididir.



Şekil 45. Bir kuyruk yöneticisine kümenin dışında konması

Bu görev hakkında

İstek ve yanıt iletileri için yolu ayarlamak üzere yönergeleri izleyin

Yordam

1. İstek iletilerini kümeden gönderin.

Küme içindeki kuyruk yöneticisinin (QM2), kümenin dışında olan QM3 konumunda Q3 kuyruğuna bir ileti yerleştirdiğini göz önünde bulundurun.

- a) Create a QREMOTE definition on QM1 that advertises the remote queue Q3 to the cluster

```
DEFINE QREMOTE(Q3) RNAME(Q3) RQMNAME(QM3) CLUSTER(DEMO)
```

Ayrıca, kümenin dışında olan kuyruk yöneticisine bir gönderen kanalı ve bir iletim kuyruğu da içerir. QM3 , karşılık gelen bir alıcı kanalına sahiptir. Kanallar, [Şekil 45 sayfa 251](#) içinde gösterilmez.

QM2 ' ta bir uygulama, hedef kuyruğu ve yanıtların gönderileceği kuyruğu belirten bir MQPUT çağırısını yayınlar. Hedef kuyruk Q3 ve yanıtlama kuyruğu Q2.

The message is sent to QM1, which uses its remote-queue definition to resolve the queue name to Q3 at QM3.

2. Yanıt iletisini, küme yöneticisinin dışındaki kuyruk yöneticisinden alın.

Kümenin dışında bir kuyruk yöneticisinin, bir ileti göndereceği kümedeki her kuyruk yöneticisi için bir kuyruk yöneticisi diğer adı olmalıdır. Kuyruk yöneticisi diğer adı, ağ geçidi kuyruk yöneticisine iletim kuyruğunun adını da belirtmelidir. Bu örnekte QM3 , QM2 için bir kuyruk yöneticisi diğer adı tanımlamasına gereksinim duyar:

- a) QM3 üzerinde bir kuyruk yöneticisi diğer adı QM2 yarat

```
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) XMITQ(QM1)
```

QM3 also needs a sender-channel and transmission queue to QM1 and QM1 needs a corresponding receiver-channel.

The application, **app3**, on QM3 can then send replies to QM2, by issuing an MQPUT call and specifying the queue name, Q2 and the queue manager name, QM2.

Sonraki adım

Bir kümeden birden çok rota tanımlayabilirsiniz.

Bir kümeden iş yükü dengelemesini yapılandırma

Kuyruk yöneticisinden, küme kuyruğundan herhangi bir kopyaya bir ileti yolu yapılandırın. Sonuç, kümenin dışından bir küme kuyruğunun her bir eşgörünümüne yönelik iş yükü dengeleme isteklerine neden olur.

Başlamadan önce

Configure the example, as shown in [Şekil 43 sayfa 246](#) in “[İstekte bulunanın/yanıtının bir kümeye ilişkin yapılandırılması](#)” sayfa 245.

Bu görev hakkında

In this scenario, the queue manager outside the cluster, QM3 in [Şekil 46 sayfa 253](#), sends requests to the queue Q2. Q2 is hosted on two queue managers within cluster DEMO to use workload balancing. A queue named Q2 is defined on the queue managers QM2 and QM4 but not on the gateway queue manager QM1. The requests from QM3, the queue manager outside the cluster, are sent to either instance of Q2.

QM3 , bir kümenin parçası değildir ve dağıtılmış kuyruklama tekniklerini kullanarak iletişim kurar. It must have a sender-channel and a transmission queue to QM1. QM1 , karşılık gelen bir alıcı kanalına gerek duyar. Kanallar ve iletim kuyrukları [Şekil 46 sayfa 253](#) içinde açık bir şekilde gösterilmiyor.

The procedure extends the example in [Şekil 43 sayfa 246](#) in “[İstekte bulunanın/yanıtının bir kümeye ilişkin yapılandırılması](#)” sayfa 245.

Yordam

1. Define a local queue called Q2 on each of QM2 and QM4.

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO) DEFBIND(NOTFIXED)
```

2. QM3 üzerinde Q2 için bir QREMOTE tanımlaması yaratın.

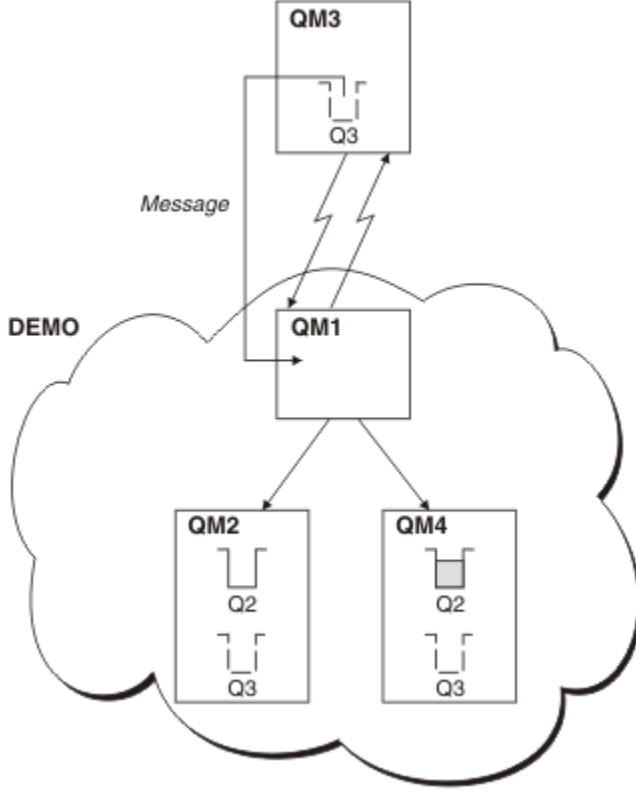
```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(Q3) XMITQ(QM1)
```

- QM3 ' un iletileri yerleştirdiği kümedeki her kuyruk için bir QREMOTE tanımlaması yaratın.
3. QM3 üzerinde bir kuyruk yöneticisi diğer adı Q3 yaratın.

```
DEFINE QREMOTE(Q3) RNAME(' ') RQMNAME(' ') CLUSTER(DEMO) DEFBIND(NOTFIXED)
```

- Q3 , gerçek bir kuyruk yöneticisi adı değil. It is the name of a queue manager alias definition in the cluster that equates the queue manager alias name Q3 with blank, ' '
4. QM1, the gateway queue manager, has no special definitions.

Sonuçlar



Şekil 46. Kuyruk yöneticisinden kümenin dışında konması

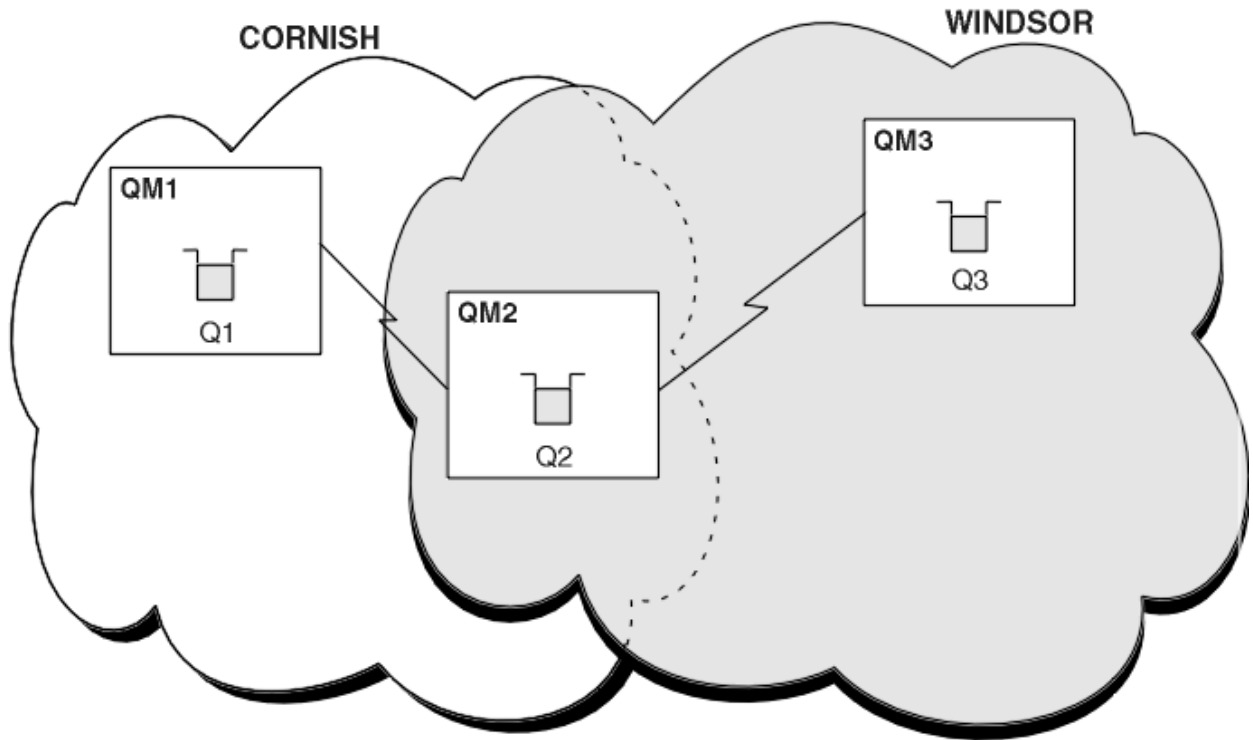
When an application at QM3 issues an MQPUT call to put a message to Q2, the QREMOTE definition causes the message to be routed through the gateway queue manager QM1. QM1 uses workload balancing to distribute messages targeted to Q2 between the queues called Q2 on the two queue managers, QM2 and QM4, which have cluster queue manager aliases for Q3.

Kümeler arasındaki ileti yollarının yapılandırılması

Bir ağ geçidi kuyruk yöneticisini kullanarak kümeleri birbirine bağlayın. Ağ geçidi kuyruk yöneticisinde küme kuyruğu ya da küme kuyruğu yöneticisi diğer adlarını tanımlayarak kuyrukları ya da kuyruk yöneticilerini tüm kümeler tarafından görülebilir yapın.

Bu görev hakkında

Tüm kuyruk yöneticilerinizi tek bir büyük kümede gruplamak yerine, daha küçük kümeleriniz olabilir. Her kümede, köprü görevi gören bir ya da daha çok kuyruk yöneticisi vardır. Bunun avantajı, kümeler genelinde kuyruk ve kuyruk yöneticisi adlarının görünürlüğüne kısıtlayabilmendir. Bkz. “Çakışan kümeler” sayfa 176. Ad çakışmalarını önlemek ya da yerel adlandırma kurallarına uymak üzere kuyruk ve kuyruk yöneticilerinin adlarını değiştirmek için diğer adları kullanın.



Şekil 47. Kümeler arasında köprü

Şekil 47 sayfa 254 içinde aralarında bir köprü bulunan iki küme gösterilmektedir. Birden fazla köprü olabilir.

Aşağıdaki yordamı kullanarak kümeleri yapılandırın:

Yordam

1. Define a cluster queue, Q1 on QM1.

```
DEFINE QLOCAL(Q1) CLUSTER(CORNISH)
```

2. Define a cluster queue, Q3 on QM3.

```
DEFINE QLOCAL(Q3) CLUSTER(WINDSOR)
```

3. Her iki kümenin adlarını içeren CORNISHWINDSOR on QM2adlı bir ad listesi oluşturun.

```
DEFINE NAMELIST(CORNISHWINDSOR) DESCR('CornishWindsor namelist')
  NAMES(CORNISH, WINDSOR)
```

4. QM2 üzerinde Q2 küme kuyruğunun tanımlanması

```
DEFINE QLOCAL(Q2) CLUSNL(CORNISHWINDSOR)
```

Sonraki adım

QM2 , her iki kümenin bir üyesidir ve aralarındaki köprüdür. Köprüde görünür kılmak istediğiniz her kuyruk için köprüde bir QALIAS tanımlaması gerekir. Örneğin, Şekil 47 sayfa 254 içinde, QM2 üzerinde aşağıdakilere gereksinim duyarsınız:

```
DEFINE QALIAS(MYQ3) TARGET(Q3) CLUSTER(CORNISH) DEFBIND(NOTFIXED)
```

Kuyruk diğer adını kullanarak, CORNISH içindeki bir kuyruk yöneticisine bağlı bir uygulama (örneğin, QM1), Q3' e bir ileti gönderebilir. Q3 ögesine MYQ3 olarak başvuruda bulunur. İleti, QM3 adresindeki Q3 adresine yönlendirilir.

Bir kuyruğu açtığınızda, DEFBIND değerini NOTFIXED ya da QDEF olarak ayarlamanız gerekir. DEFBIND varsayılan olarak bırakılırsa OPEN, kuyruk yöneticisi diğer ad tanımlamasını onu barındıran köprü kuyruk yöneticisine çözer. Köprü mesajı iletmiyor.

Görünür kılmak istediğiniz her kuyruk yöneticisi için bir kuyruk yöneticisi diğer adı tanımlaması gerekir. Örneğin, QM2 üzerinde aşağıdakilere gereksinim duyarsınız:

```
DEFINE QREMOTE(QM1) RNAME(' ') RQMNAME(QM1) CLUSTER(WINDSOR)
```

WINDSOR içindeki herhangi bir kuyruk yöneticisine bağlı bir uygulama (örneğin, QM3), MQOPEN çağrısında QM1 ögesini belirttik olarak adlandırarak QM1 üzerindeki herhangi bir kuyruğa bir ileti yerleştirebilir.

Kuyruk yöneticisi diğer adları ve kümeleri

Bir kümeye ileti gönderirken ya da bir kümeye gönderilen iş yükü dengesi iletilerine ileti gönderirken kuyruk yöneticilerinin adını gizlemek için kuyruk yöneticisi diğer adlarını kullanın.

Boş bir RNAME ile uzak kuyruk tanımlaması kullanılarak oluşturulan kuyruk yöneticisi diğer adları beş kullanım alanı içerir:

İleti gönderirken kuyruk yöneticisi adı yeniden eşleniyor

A queue-manager alias can be used to remap the queue-manager name specified in an MQOPEN call to another queue manager. Bu bir küme kuyruğu yöneticisi olabilir. Örneğin, kuyruk yöneticisi kuyruk yöneticisi diğer adı tanımlamasına sahip olabilir:

```
DEFINE QREMOTE(YORK) RNAME(' ') RQMNAME(CLUSQM)
```

YORK, CLUSQM adlı kuyruk yöneticisi için bir diğer ad olarak kullanılabilir. Bu tanımlamayı yapan kuyruk yöneticisine ilişkin bir uygulama, YORK kuyruk yöneticisine bir ileti yerleştirdiğinde, yerel kuyruk yöneticisi adı CLUSQM olarak çözer. If the local queue manager is not called CLUSQM, it puts the message on the cluster transmission queue to be moved to CLUSQM. Ayrıca, iletim üstbilgisini YORK yerine CLUSQM olarak ifade eder.

Not: Tanım yalnızca, bunu yapan kuyruk yöneticisinde geçerlidir. Diğer adı tüm kümeye duyurmak için, uzak kuyruk tanımlamasına CLUSTER özniteliğini eklemeniz gerekir. Daha sonra, YORK için yazılmış diğer kuyruk yöneticilerinden gelen iletiler CLUSQM ' e gönderilir.

İleti gönderirken iletim kuyruğunu değiştirme ya da belirleme

Yöneltilme, küme olmayan bir sisteme kümeye katılmak için kullanılabilir. For example, queue managers in the cluster ITALY could communicate with the queue manager called PALERMO, which is outside the cluster. İletişim kurmak için, kümedeki kuyruk yöneticilerinden biri ağ geçidi işlevi görmelidir. Ağ geçidi kuyruk yöneticisinden şu komutu verin:

```
DEFINE QREMOTE(ROME) RNAME(' ') RQMNAME(PALERMO) XMITQ(X) CLUSTER(ITALY)
```

Komut, bir kuyruk yöneticisi diğer adı tanımlamasıdır. It defines and advertises ROME as a queue manager over which messages from any queue manager in the cluster ITALY can multi-hop to reach their destination at PALERMO. Kuyruk yöneticisi adı ROME ile açılmış bir kuyruğa yollanmış iletiler, ağ geçidi kuyruk yöneticisine kuyruk yöneticisi diğer adı tanımlamasıyla gönderilir. Buradan bir kez, iletiler iletim kuyruğuna (X) yerleştirilir ve küme dışı kanallarla kuyruk yöneticisine taşınır PALERMO .

Bu örnekteki ROME adının seçimi anlamlı değildir. QREMOTE ve RQMNAME değerleri her ikisi de aynı olabilir.

İleti alınırken hedef belirleniyor

Kuyruk yöneticisi bir ileti aldığı anda, iletim üstbilgisinden hedef kuyruğunun ve kuyruk yöneticisinin adını alır. Bu, iletim üstbilgisindeki kuyruk yöneticisiyle aynı adı taşıyan bir kuyruk yöneticisi diğer adı tanımını arar. bir değer bulursa, iletim üstbilgisindeki kuyruk yöneticisi adına ilişkin kuyruk yöneticisi diğer adı tanımlamasından RQMNAME ' yi değiştirir.

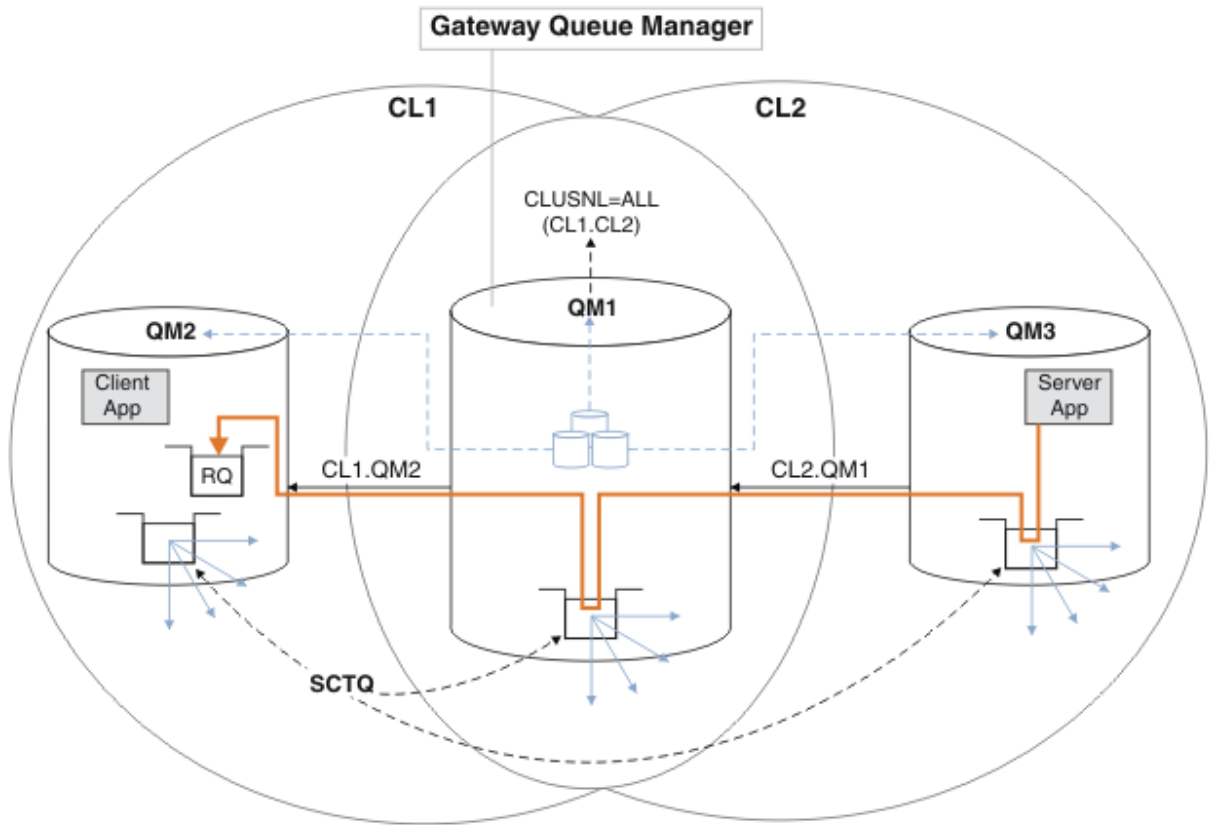
Bu şekilde kuyruk yöneticisi diğer adı kullanmanın iki nedeni vardır:

- İletileri başka bir kuyruk yöneticilerine yönlendirmek için
- Kuyruk yöneticisi adını yerel kuyruk yöneticisiyle aynı olacak şekilde değiştirmek için

Farklı kümelerdeki kuyruk yöneticileri arasında iletileri yönlendirmek için, bir ağ geçidi kuyruk yöneticisinde kuyruk yöneticisi diğer adlarını kullanma.

Bir uygulama, kuyruk yöneticisi diğer adı kullanılarak farklı bir kümede bulunan bir kuyruğa ileti gönderebilir. Kuyruk, bir küme kuyruğu olmak zorunda değil. Kuyruk bir kümede tanımlıdır. Uygulama, farklı bir kümede bulunan bir kuyruk yöneticisine bağlı. Bir ağ geçidi kuyruk yöneticisi iki kümeyi bağlar. Kuyruk kümelenmiş olarak tanımlanmadıysa, doğru yöneltmenin gerçekleşmesini sağlamak için, uygulamanın kuyruk adı ve kümelenmiş kuyruk yöneticisi diğer adı kullanılarak kuyruğu açması gerekir. For an example of a configuration, see “Ağ geçidi kuyruk yöneticisiyle çıkan iki küme oluşturma” sayfa 213, from which the reply message flow illustrated in figure 1, is taken.

Çizge, yanıt iletisi tarafından alınan yolu, RQolarak adlandırılan geçici bir dinamik kuyruğa geri gösterir. QM3' e bağlı sunucu uygulaması, QM2kuyruk yöneticisi adını kullanarak yanıt kuyruğunu açar. Kuyruk yöneticisi adı QM2 , QM1üzerinde kümelenmiş kuyruk yöneticisi diğer adı olarak tanımlanır. QM3 , yanıt iletisini QM1' e yönlendirir. QM1 , iletiyi QM2' e yönlendirir.



Şekil 48. Yanıt iletisini farklı bir kümeye döndürmek için kuyruk yöneticisi diğer adının kullanılması

Yönlendirmenin çalışma şekli şu şekildedir. Her kümedeki her kuyruk yöneticisinin QM1üzerinde bir kuyruk yöneticisi diğer adı tanımlaması vardır. Diğer adlar tüm kümelerde kümelenir. Her bir diğer addan bir kuyruk yöneticisine gelen gri çizgili oklar, her kuyruk yöneticisi diğer adının kümelerin en az birinde gerçek bir kuyruk yöneticisine çözümlendiğini gösterir. Bu durumda, QM2 diğer adı hem kümede CL1 hem de CL2kümelenir ve CL1içindeki QM2 gerçek kuyruk yöneticisine çözümlenir. Sunucu uygulaması, yanıt iletisini RQkuyruk adını kullanarak ve QM2kuyruk yöneticisine yanıt vererek yaratır. QM2 kuyruk yöneticisi diğer ad tanımlaması QM1 kümesinde CL2 ve QM2 kuyruk yöneticisinde CL2tanımlı olmadığı için ileti QM1 olarak yönetilir. İleti hedef kuyruk yöneticisine gönderilemediğinden, diğer ad tanımlamasına sahip kuyruk yöneticisine gönderilir.

QM1 places the message on the cluster transmission queue on QM1 for transferal to QM2. QM1 , QM2 için QM1 üzerindeki kuyruk yöneticisi diğer adı tanımlaması QM2 'i gerçek hedef

kuyruk yöneticisi olarak tanımladığından, iletiyi QM2 ' e yöneltir. Diğer ad tanımlamaları yalnızca gerçek tanımlamalara gönderme yapabildiği için tanımlama çevrimsel değildir; diğer ad kendisini gösteremez. Hem QM1 hem de QM2 aynı kümede (CL1) olduğundan, gerçek tanımlama QM1 tarafından çözümlenir. QM1 , CL1havuzundan QM2 ile ilgili bağlantı bilgilerini bulur ve iletiyi QM2' e yönlendirir. İletin QM1 tarafından yeniden yönlendirilmesi için, sunucu uygulaması yanıt kuyruğunu DEFBIND seçeneği MQBND_BIND_NOT_FIXEDolarak ayarlanmış şekilde açmış olmalıdır. Sunucu uygulaması yanıt kuyruğunu MQBND_BIND_ON_OPENseçeneğiyle açtıysa, ileti yeniden yönlendirilmez ve gitmeyen iletiler kuyruğuna gönderilir.

Bir kuyruk yöneticisini kümeden gelen iş yükü dengelemesi iletilerine iş yükü dengelemesi için bir ağ geçidi olarak kullanma.

Kümede birden çok kuyruk yöneticisinde EDINBURGH adlı bir kuyruk tanımladınız. Kümeleme mekanizmasının, kümenin dışından gelen iletilere ilişkin iş yükünü dengelemesini istiyorsunuz.

Kümeden bir kuyruk yöneticisinin, küme içindeki bir kuyruk yöneticisine bir iletim kuyruğu ve gönderici kanalı olması gerekir. Bu kuyruğun adı bir ağ geçidi kuyruk yöneticisi olarak adlandırılır. Varsayılan iş yükü dengeleme mekanizmasından yararlanmak için aşağıdaki kurallardan birinin geçerli olması gerekir:

- Ağ geçidi kuyruk yöneticisi, EDINBURGH kuyruğunun bir eşgörünümünü içermemelidir.
- Ağ geçidi kuyruk yöneticisi, ALTER QMGRüzerinde CLWLUSEQ (ANY) değerini belirtir.

Bir küme dışından iş yükü dengelemesi örneği için bkz. [“Bir kümeden iş yükü dengelemesini yapılandırma” sayfa 252](#)

Yanıt-kuyruk diğer adları ve kümeleri için

Yanıt kuyruğu diğer adı tanımlaması, yanıt bilgileri için diğer adları belirtmek için kullanılır. Yanıtın gönderileceği kuyruk diğer adı tanımlamaları, dağıtılmış bir kuyruklama ortamında olduğu gibi kümelerle kullanılabilir.

Örneğin:

- An application at queue manager VENICE sends a message to queue manager PISA using the MQPUT call. Uygulama, ileti tanımlayıcısında aşağıdaki yanıtlama kuyruğu bilgilerini sağlar:

```
ReplyToQ='QUEUE'  
ReplyToQMgr=' '
```

- In order that replies sent to QUEUE can be received on OTHERQ at PISA, create a remote-queue definition on VENICE that is used as a reply-to queue alias. Diğer ad, yalnızca yaratıldığı sistemde yürürlükte olan diğer ad.

```
DEFINE QREMOTE(QUEUE) RNAME(OTHERQ) RQMNAME(PISA)
```

RQMNAME bir küme kuyruk yöneticisi olsa bile,RQMNAME ve QREMOTE aynı adları belirtebilir.

Kuyruk diğer adları ve kümeleri

Bir küme kuyruğunun adını gizlemek, bir kuyruğu kümelemek, farklı öznitelikleri benimsemek ya da farklı erişim denetimlerini benimsemek için kuyruk diğer adlarını kullanın.

QALIAS tanımlaması, kuyruğun tanınacağı bir diğer ad yaratmak için kullanılır. Çeşitli nedenlerden ötürü bir diğer ad yaratabilirsiniz:

- Farklı bir kuyruk kullanmaya başlamak istiyorsunuz, ancak uygulamalarınızı değiştirmek istemiyorsunuz.
- Uygulamaların, iletileri koydukları kuyruğun gerçek adını bilmesini istemezsiniz.
- Kuyruğun tanımlandığı yerden farklı bir adlandırma kuralınız olabilir.
- Uygulamalarınızın kuyruğa gerçek adıyla değil, yalnızca diğer adıyla erişme yetkisi olabilir.

DEFINE QALIAS komutunu kullanarak bir kuyruk yöneticisinde QALIAS tanımlaması yaratın. Örneğin, şu komutu çalıştırın:

```
DEFINE QALIAS(PUBLIC) TARGET(LOCAL) CLUSTER(C)
```

Komut, Ckümesindeki kuyruk yöneticilerine PUBLIC adlı bir kuyruk tanıtır. PUBLIC , LOCALadlı kuyruğa çözülen bir diğer addır. PUBLIC ' e gönderilen iletiler, LOCALadlı kuyruğa yönlendirilir.

Bir kuyruk adını küme kuyruğuna çözümlenmek için kuyruk diğer adı tanımlamasını da kullanabilirsiniz. Örneğin, şu komutu çalıştırın:

```
DEFINE QALIAS(PRIVATE) TARGET(PUBLIC)
```

Bu komut, bir kuyruk yöneticisinin kümenin başka bir yerinde PUBLICadıyla yayınlanan bir kuyruğa erişmek için PRIVATE adını kullanmasını sağlar. Bu tanımlama CLUSTER özneliğini içermediğinden, yalnızca onu yapan kuyruk yöneticisi için geçerlidir.

İş yükü yönetimi için kümeleri kullanma

Bir kümenin farklı kuyruk yöneticilerindeki bir kuyruğun birden çok örneğini tanımlayarak, kuyruğa bakım yapma işini birden çok sunucu üzerinden dağıtabilirsiniz. Hata durumunda, iletilerin farklı bir kuyruk yöneticisinde yeniden kuyruğa girilmesini önleyen birkaç faktör vardır.

Sistem yönetimini azaltmak için kümeleri ayarlamamanın yanı sıra, birden çok kuyruk yöneticisinin aynı kuyruğun bir örneğini barındırdığı kümeleri yaratabilirsiniz.

Kümelerinizi, kuyruk yöneticilerinin birbirinin klonlarında olduğu gibi düzenleyebilirsiniz. Her kuyruk yöneticisi aynı uygulamaları çalıştırabiliyor ve aynı kuyruklara ilişkin yerel tanımlara sahip. Bir uygulamanın birden çok eşgörünümünü kullanarak, iş yükünü kuyruk yöneticilerinizle birlikte yayınlatabilirsiniz. Uygulamanın her bir eşgörünümü ileti alır ve birbirinden bağımsız olarak çalışır.

Kümeleri bu şekilde kullanmanın avantajları şunlardır:

- Kuyruklarınızın ve uygulamalarınızın kullanılabilirliğinin artırılmış olması
- İletilerin daha hızlı veri çıkışı
- Ağınızda iş yükünün daha fazla dağıtılması

Belirli bir kuyruğa ilişkin bir yönetim ortamını bulandıran kuyruk yöneticilerinden herhangi biri, bu kuyruğa ilişkin iletileri işleyebilirler. İletiler gönderilirken, uygulamalar kuyruk yöneticisi adını göstermiyor. Bir iş yükü yönetimi algoritması, iletiyi hangi kuyruk yöneticisinin işleyeceğini belirler.

İş yükü yönetimine ilişkin küme yapılandırılmalarıyla ilgili daha fazla bilgi için aşağıdaki alt konulara bakın:

İlgili kavramlar

Kümeler

Kümeler nasıl çalışır

[“Kümeleme ve dağıtılmış kuyruklama karşılaştırması” sayfa 158](#)

Dağıtılmış kuyruğa alma ve kümeleme kullanarak kuyruk yöneticilerini bağlamak için tanımlanması gereken bileşenleri karşılaştırın.

[“Bir kümenin bileşenleri” sayfa 161](#)

Kümeler, kuyruk yöneticilerinden, küme havuzlarından, küme kanallarından ve küme kuyruklarından oluşur.

[“IBM WebSphere MQ kümelerini yönetme” sayfa 181](#)

IBM WebSphere MQ kümelerini oluşturabilir, genişletebilir ve koruyabilirsiniz.

[“İletileri kümeden ve kümeden yönlendirme” sayfa 244](#)

Kümeleri dış kuyruk yöneticilerine ve diğer kümelere bağlamak için kuyruk diğer adlarını, kuyruk yöneticisi diğer adlarını ve uzak kuyruk tanımlamalarını kullanın.

İlgili görevler

[“Kuyruk yöneticisi kümesinin yapılandırılması” sayfa 156](#)

Kümelerin nasıl çalıştığını, küme yapılandırmasının nasıl tasarlayacağını ve basit bir kümenin nasıl ayarlanacağına ilişkin bir örnek almak için bu konudaki bağlantıları kullanın.

[“Yeni bir küme ayarlanıyor” sayfa 181](#)

Örnek kümeyi ayarlamak için bu yönergeleri izleyin. Ayrı yönergeler, TCP/IP, LU 6.2 üzerinde ve tek bir iletim kuyruğu ya da birden çok iletim kuyruğundaki kümeyi ayarlayıp tanımlamalarını açıklar. Bir kuyruk yöneticisinden diğerine ileti göndererek, kümenin çalıştığını test edin.

[Küme iş yükü çıkışlarının yazılması ve derlenmesi](#)

Kuyruğun birden çok eşgörünümü olan bir küme örneği

Kuyruğun birden çok örneğine sahip bir küme örneğinde, iletiler kuyruğun farklı örneklerine yönlendirilir. Bir iletiyi, kuyruğun belirli bir örneğine zorlayabilir ve kuyruk yöneticilerinden birine ileti dizisi göndermeyi seçebilirsiniz.

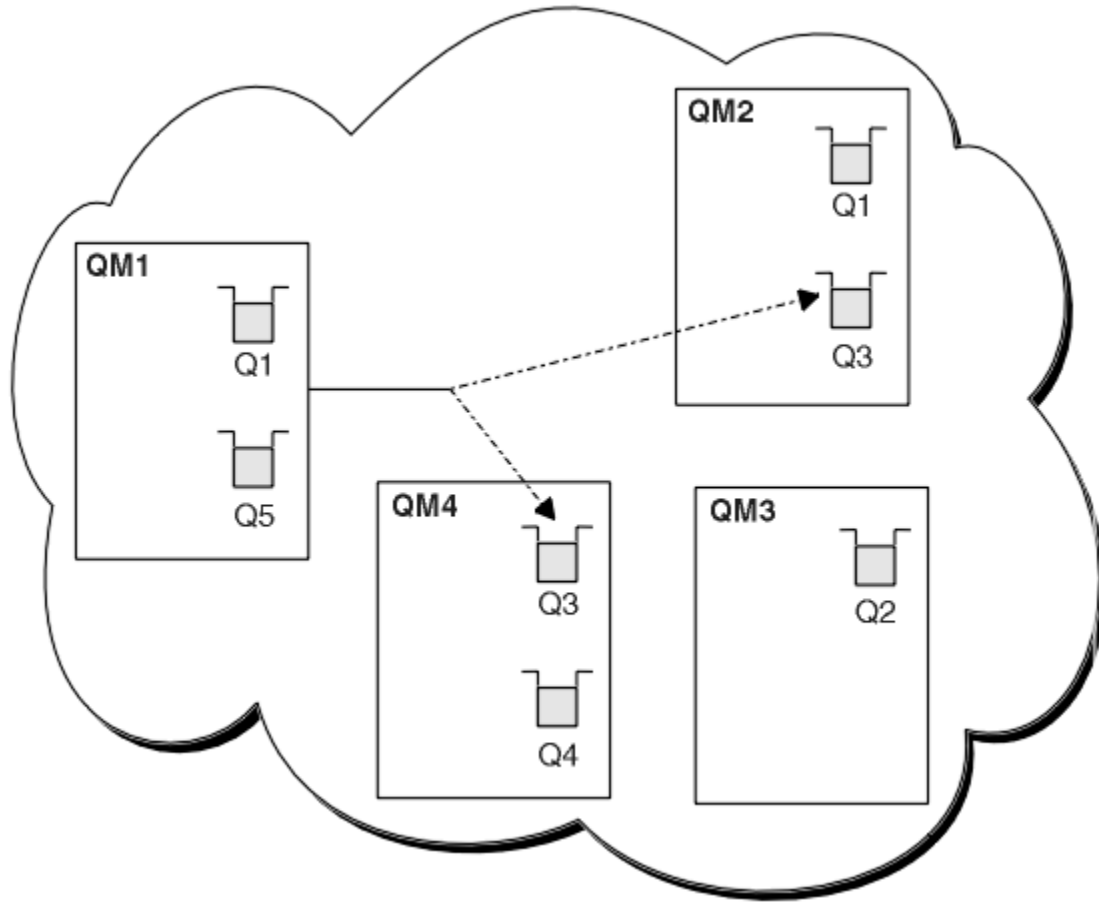
Şekil 49 sayfa 260 shows a cluster in which there is more than one definition for the queue Q3. If an application at QM1 puts a message to Q3, it does not necessarily know which instance of Q3 is going to process its message. If an application is running on QM2 or QM4, where there are local instances of Q3, the local instance of Q3 is opened by default. CLWLUSEQ kuyruk özneliğini ayarlayarak, kuyruğun yerel yönetim ortamı, kuyruğun uzak bir örneğiyle aynı şekilde değerlendirilebilir.

MQOPEN seçeneği DefBind, MQOPEN çağrısı yayınlandığında hedef kuyruk yöneticisinin seçilip seçilmediğini ya da ileti iletim kuyruğundan aktarılıp aktarılmadığını denetler.

DefBind seçeneğini MQBND_BIND_NOT_FIXED olarak ayarlarsanız, ileti iletildiğinde kullanılabilir olan bir kuyruk örneğine ileti gönderilebilir. Bu, aşağıdaki sorunları önler:

- Hedef kuyruk yöneticisinde gelen ileti geldiğinde hedef kuyruk kullanılmıyor.
- Kuyruğun durumu değişti.
- İleti, küme kuyruğu diğer adı kullanılarak alındı ve kuyruk yöneticisinin, küme kuyruğu diğer adının tanımlandığı kuyruk yöneticinde hiçbir hedef kuyruk örneği yok.

İletim sırasında bu sorunlar saptanırsa, hedef kuyruğun başka bir kullanılabilir eşgörünümü aranır ve ileti yeniden yönlendirilir. Kuyruğun hiçbir örneği kullanılabilir değilse, ileti, ölü-mektup kuyruğuna yerleştirilir.



Şekil 49. Aynı kuyruğun birden çok örneğine sahip bir küme

One factor that can prevent messages being rerouted is if messages have been assigned to a fixed queue manager or channel with MQBND_BIND_ON_AÇIK. MQOPEN ' a bağlı iletiler hiçbir zaman başka bir kanala yeniden atanmaz. Ayrıca, ileti yeniden konumlarının yalnızca bir küme kanalı gerçekte başarısız olduğunda gerçekleşeceğini unutmayın. Kanal önceden başarısız olursa, gerçek konum oluşmaz.

Hedef kuyruk yöneticisi hizmetten çıkıyorsa, sistem bir iletiyi yeniden yöneltmeye çalışır. Bu durumda, bunu kaybetme riskini çalıştırarak ya da bir yineleme yaratarak, iletinin bütünlüğünü etkilemez. Bir kuyruk yöneticisi başarısız olursa ve bir iletiyi kuşku içinde bırakırsa, bu ileti yeniden yönlendirilmez.

Kuyruğu yerel olarak barındıran bir kuyruk yöneticisi eklenmesi

Paris ve New York 'taki envanter uygulama sistemini çalıştırmak üzere ek kapasite sağlamak üzere INVENTQ yönetim ortamını eklemek için bu yönergeleri izleyin.

Başlamadan önce

Not: Küme genelinde bir kümede geçirecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, Küme için yeni bir kuyruk yöneticisi eklenmesibaşlıklı konu anlatıldığı gibi ayarlandı. Bu, üç kuyruk yöneticisi içerir; LONDON ve NEWYORK her ikisi de tam havuz barındırıyor, PARIS kısmi bir havuz barındırıyor. Döküm uygulaması, NEWYORK kuyruk yöneticisine bağlı olarak, sistemde New York 'ta çalışır. The application is driven by the arrival of messages on the INVENTQ queue.
- Paris ve New York 'taki envanter uygulama sistemini çalıştırmak için ek kapasite sağlamak üzere INVENTQ yönetim ortamı eklemek istiyoruz.

Bu görev hakkında

Kuyruğu yerel olarak barındıran bir kuyruk yöneticisi eklemek için aşağıdaki adımları izleyin.

Yordam

1. PARIS kuyruk yöneticisini değiştirin.

Paris 'te, Paris 'te ve New York 'ta bulunan INVENTQ uygulaması için, kuyruk yöneticisini bilgilendirmek zorundayız. PARIS üzerinde şu komutu verin:

```
ALTER QMGR CLWLUSEQ (ANY)
```

2. İleti zenginlikleri için envanter uygulamasını gözden geçirin.

Devam etmeden önce, döküm uygulamasının iletilerin işlenmesi sırasında herhangi bir bağımlılıkları olmadığından emin olun. Daha fazla bilgi için, bkz. [“İleti zenginlikleri işleniyor” sayfa 271](#).

3. Döküm uygulamasını sisteme Paris 'te kurun.

4. Define the cluster queue INVENTQ.

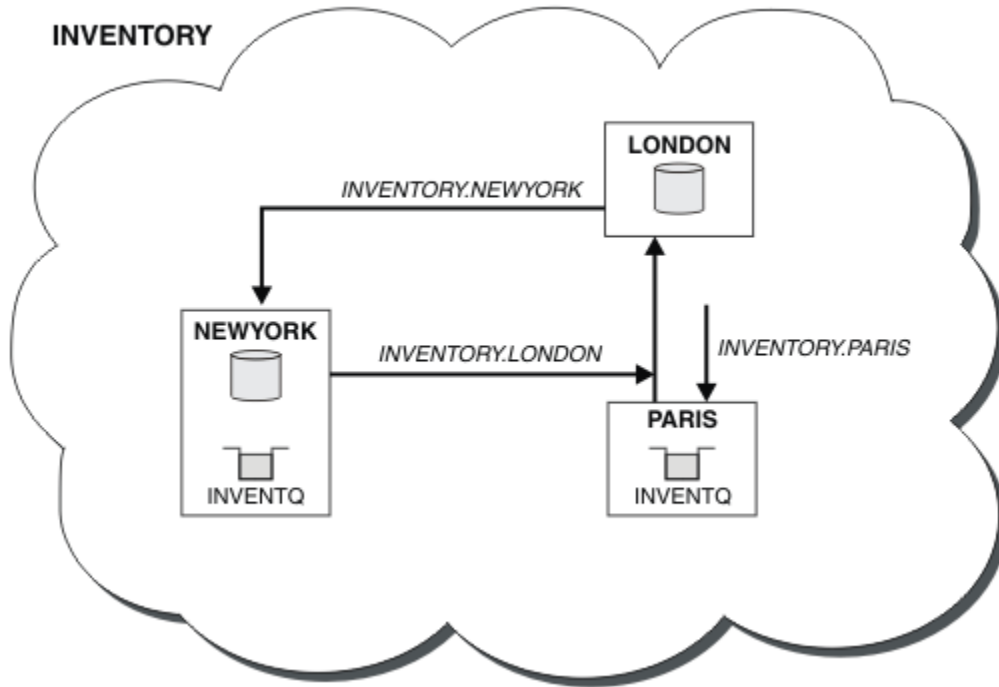
NEWYORK kuyruk yöneticisi tarafından önceden barındırılan INVENTQ kuyruğu da PARIS tarafından barındırılır. Bunu PARIS kuyruk yöneticisine aşağıdaki şekilde tanımlayın:

```
DEFINE QLOCAL (INVENTQ) CLUSTER (INVENTORY)
```

Şimdi tüm tanımları tamamladığınızı, henüz yapmadıysanız, kanal başlatıcısını WebSphere MQ for z/OS üzerinde başlatın. On all platforms, start a listener program on queue manager PARIS. Dinleyici, gelen ağ isteklerini dinler ve gerektiğinde kümeli alıcı kanalını başlatır.

Sonuçlar

[Şekil 50 sayfa 261](#) , bu görev tarafından ayarlanan kümeyi gösterir.



Şekil 50. Üç kuyruk yöneticisiyle birlikte INVENTORY kümesi

The modification to this cluster was accomplished without you altering the queue managers NEWYORK or LONDON. The full repositories in these queue managers are updated automatically with the information they need to be able to send messages to INVENTQ at PARIS.

Sonraki adım

INVENTQ kuyruğu ve döküm uygulaması, kümedeki iki kuyruk yöneticisinde artık barındırılır. Bu, onların kullanılabilirliğini artırır, ileti verimini hızlandırır ve iş yükünün iki kuyruk yöneticisi arasında dağıtılmasını sağlar. Messages put to INVENTQ by any of the queue managers LONDON, NEWYORK, PARIS are routed alternately to PARIS or NEWYORK, so that the workload is balanced.

Bir kümede iki ağın kullanılması

Follow these instructions to add a new store in TOKYO where there are two different networks. Her iki durumda da Tokyo 'daki kuyruk yöneticisiyle iletişim kurmak için kullanılabilir olması gerekir.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- "Küme için kuyruk yöneticisi eklenmesi" başlıklı konu anlatıldığı gibi, INVENTORY kümesi ayarlandı. Bu, üç kuyruk yöneticisi içerir; LONDON ve NEWYORK her ikisi de tam havuz barındırıyor, PARIS kısmi bir havuz barındırıyor. Döküm uygulaması, NEWYORK kuyruk yöneticisine bağlı olarak, sistemde New York 'ta çalışır. The application is driven by the arrival of messages on the INVENTQ queue.
- TOKYO ' ta iki farklı ağ olan yeni bir mağaza ekleniyor. Her iki durumda da Tokyo 'daki kuyruk yöneticisiyle iletişim kurmak için kullanılabilir olması gerekir.

Bu görev hakkında

Bir kümede iki ağı kullanmak için bu adımları izleyin.

Yordam

1. TOKYO hangi tam havuzunun ilk önce başvurmaya karar vereceğine karar verin.

Bir kümedeki her kuyruk yöneticisinin, kümeyle ilgili bilgi toplamak için, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Kendi kısmi havuzunu oluşturur. Hangi havuzu seçtiğiniz belli bir öneme sahip değildir. Bu örnekte, NEWYORK seçilidir. Yeni kuyruk yöneticisi, her iki havuzla iletişim kurduğu kümeye yeni bir kuyruk yöneticisi katıldıktan sonra.

2. CLUSTRCVR kanallarını tanımlayın.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcısını tanımlamak gerekir. Bu kuyruk yöneticisinin her bir ağ üzerinde iletişim kurabilmesi gerekir.

```
DEFINE CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME('TOKYO.NETB.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel using network B for TOKYO')
```

```
DEFINE CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME('TOKYO.NETA.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel using network A for TOKYO')
```

3. Define a CLUSSDR channel on queue manager TOKYO.

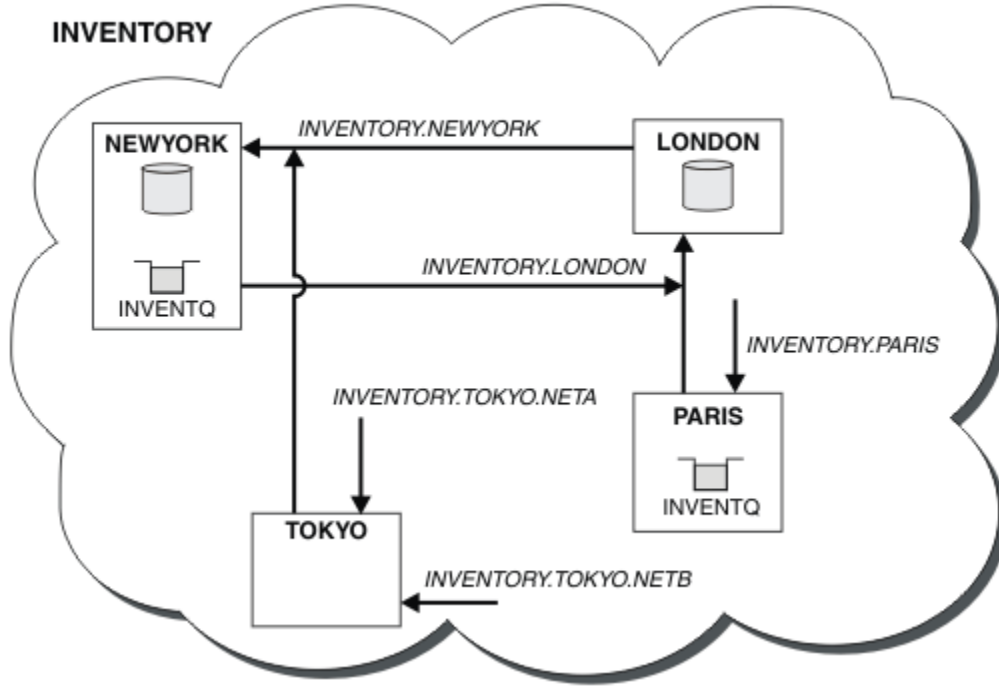
Bir kümedeki her kuyruk yöneticisinin, ilk tam havuzuna ileti gönderebileceği bir küme gönderici kanalı tanımlanması gerekir. Bu durumda NEWYORK'u seçmiş olduk, bu nedenle TOKYO ' in aşağıdaki tanıma gereksinimi vardır:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
channel from TOKYO to repository at NEWYORK')
```

Şimdi, tüm tanımları tamamladığınızı, henüz yapmadıysanız, kanal başlatıcısını WebSphere MQ for z/OS üzerinde başlatmış olmadıysanız. On all platforms, start a listener program on queue manager PARIS. Dinleyici programı gelen ağ isteklerini dinler ve gerektiğinde küme alıcı-alıcı kanalını başlatır.

Sonuçlar

Şekil 51 sayfa 263 , bu görev tarafından ayarlanan kümeyi gösterir.



Şekil 51. Dört kuyruk yöneticisiyle birlikte INVENTORY kümesi

By making only three definitions, we have added the queue manager TOKYO to the cluster with two different network routes available to it.

İlgili görevler

“Küme için kuyruk yöneticisi eklenmesi” sayfa 191

Yarattığınız kümeye kuyruk yöneticisi eklemek için bu yönergeleri izleyin. Küme kuyruklarına ve konulara ilişkin iletiler tek küme iletim kuyruğu SYSTEM . CLUSTER . TRANSMIT . QUEUE kullanılarak aktarılır.

Bir kümede birincil ve ikincil bir ağ kullanılması

Bir ağı birincil ağ ve başka bir ağ ağı yapmak için bu yönergeleri izleyin. Birincil ağıyla ilgili bir sorun varsa, yedek ağı kullanın.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Bir kümede iki ağın kullanılması” sayfa 262 içinde açıklandığı şekilde ayarlanmıştır. Bu, dört kuyruk yöneticisi içerir; LONDON ve NEWYORK her ikisi de tam havuz içerir; PARIS ve TOKYO kısmi havuzları içerir. The inventory application runs on the system in New York, connected to the queue manager NEWYORK. TOKYO kuyruk yöneticisinin iletişim kurabileceği iki farklı ağı vardır.

- Ağlardan birini birincil ağ, diğeri de yedek ağ ağlarından biri yapmak istiyorsunuz. Birincil ağıyla ilgili bir sorun varsa, yedek ağı kullanmayı planlayın.

Bu görev hakkında

Bir kümede birincil ve ikincil bir ağı yapılandırmak için NETPRTY özneliğini kullanın.

Yordam

TOKYO üzerindeki var olan CLUSRCVR kanallarını değiştirin.

A kanalının birincil kanal olduğunu ve ağ B kanalının ikincil kanal olduğunu belirtmek için aşağıdaki komutları kullanın:

- ALTER CHANNEL (INVENTORY.TOKYO.NETA) CHLTYPE (CLUSRCVR) NETPRTY (2) DESCR ('Main cluster-receiver channel for TOKYO')
- ALTER CHANNEL (INVENTORY.TOKYO.NETB) CHLTYPE (CLUSRCVR) NETPRTY (1) DESCR ('Backup cluster-receiver channel for TOKYO')

Sonraki adım

Kanalı farklı ağ önceliklerine sahip olarak yapılandırarak, artık birincil ağ ve ikincil bir ağ olan kümeyle tanımlandınız. Kümedeki kuyruk yöneticileri, kullanılabilir olduğunda, bu kanalları kullanan birincil ağı otomatik olarak kullanır. Birincil ağ kullanılabilir durumda değilse, ikincil ağ kullanmak için hata durumunda yedek kuyruk yöneticilerine geçiş sağlar.

Yedekleme olarak işlem yapmak üzere kuyruk ekleme

Şu anda New York 'ta çalışan envanter sistemi için Chicago 'da bir yedekleme sağlamak için bu yönergeleri izleyin. Chicago sistemi sadece New York sistemi ile ilgili bir sorun olduğunda kullanılır.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Küme için kuyruk yöneticisi eklenmesi” sayfa 191 içinde açıklandığı şekilde ayarlanmıştır. Bu, üç kuyruk yöneticisi içerir; LONDON ve NEWYORK her ikisi de tam havuz barındırıyor, PARIS kısmi bir havuz barındırıyor. Döküm uygulaması, NEWYORK kuyruk yöneticisine bağlı olarak, sistemde New York 'ta çalışır. The application is driven by the arrival of messages on the INVENTQ queue.
- Chicago 'da yeni bir mağaza, New York 'ta çalışan envanter sistemi için bir destek sağlamak üzere hazırlanıyor. Chicago sistemi sadece New York sistemi ile ilgili bir sorun olduğunda kullanılır.

Bu görev hakkında

Yedekleme olarak işlem yapmak üzere bir kuyruk eklemek için aşağıdaki adımları izleyin.

Yordam

1. CHICAGO hangi tam havuzunun ilk önce başvurmasına karar vereceğine karar verin.

Bir kümedeki her kuyruk yöneticisinin, kümeyle ilgili bilgi toplamak için, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Kendi kısmi havuzunu oluşturur. Belirli bir kuyruk yöneticisi için hangi havuzu seçmiş olduğunuz belli bir öneme sahip değildir. Bu örnekte, NEWYORK seçilidir. Yeni kuyruk yöneticisi, her iki havuzla iletişim kurduğu kümeyle yeni bir kuyruk yöneticisi katıldıktan sonra.

2. CLUSTVR kanalını tanımlayın.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcısını tanımlamak gerekir. CHICAGO üzerinde şunları tanımlayın:

```
DEFINE CHANNEL(INVENTORY.CHICAGO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(CHICAGO.CMSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel for CHICAGO')
```

3. Define a CLUSSDR channel on queue manager CHICAGO.

Bir kümedeki her kuyruk yöneticisinin, ilk tam havuzuna ileti gönderebileceği bir küme gönderici kanalı tanımlanması gerekir. Bu durumda NEWYORK'u seçmiş olduk, bu nedenle CHICAGO ' in aşağıdaki tanıma gereksinimi vardır:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
channel from CHICAGO to repository at NEWYORK')
```

4. Alter the existing cluster queue INVENTQ.

NEWYORK kuyruk yöneticisi tarafından önceden barındırılan INVENTQ , kuyruğun ana yönetim ortağıdır.

```
ALTER QLOCAL(INVENTQ) CLWLPRTY(2)
```

5. İleti zenginlikleri için envanter uygulamasını gözden geçirin.

Devam etmeden önce, döküm uygulamasının iletilerin işlenmesi sırasında herhangi bir bağımlılıkları olmadığından emin olun.

6. Install the inventory application on the system in CHICAGO.

7. INVENTQ yedek küme kuyruğunu tanımlayın

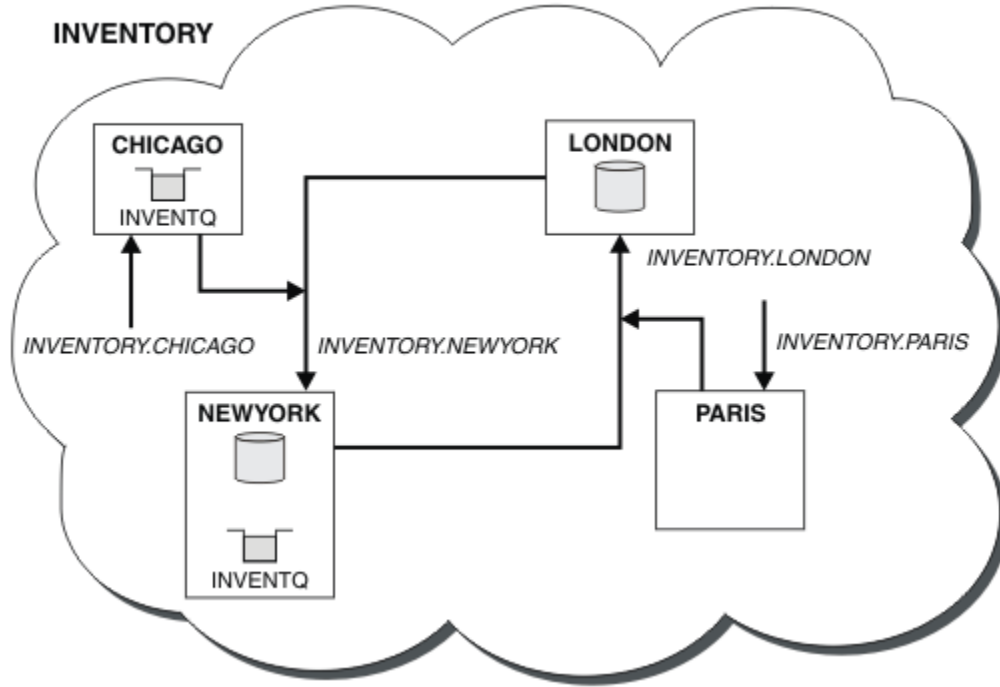
NEWYORK kuyruk yöneticisi tarafından önceden barındırılan INVENTQ , CHICAGO tarafından bir yedek olarak da barındırılır. Bunu CHICAGO kuyruk yöneticisine aşağıdaki şekilde tanımlayın:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) CLWLPRTY(1)
```

Şimdi, tüm tanımları tamamladığınızı, henüz yapmadıysanız, kanal başlatıcısını WebSphere MQ for z/OS üzerinde başlatmış olmadıysanız. On all platforms, start a listener program on queue manager CHICAGO. Dinleyici programı gelen ağ isteklerini dinler ve gerektiğinde küme alıcı-alıcı kanalını başlatır.

Sonuçlar

Şekil 52 sayfa 266 , bu görev tarafından ayarlanan kümeyi gösterir.



Şekil 52. Dört kuyruk yöneticisiyle birlikte INVENTORY Cluster

INVENTQ kuyruğu ve döküm uygulaması, kümedeki iki kuyruk yöneticisinde artık barındırılır. CHICAGO kuyruk yöneticisi bir yedeklemedir. Messages put to INVENTQ are routed to NEWYORK unless it is unavailable when they are sent instead to CHICAGO.

Not:

Uzak kuyruk yöneticisinin kullanılabilirliği, kanalın bu kuyruk yöneticisine ilişkin durumunu temel alır. Kanallar başlatıldığında, bazı durumların küme iş yükü yönetimi algoritmasına daha az tercihli olması nedeniyle, bunların durumu birkaç kez değişir. Uygulamada, daha yüksek öncelikli (birincil) varış noktaları kanalları başlatılırken daha düşük öncelik (yedekleme) hedeflerinin seçilebileceği anlamına gelir.

Herhangi bir iletinin bir yedekleme hedefine gitmediğinden emin olmanız gerekiyorsa, CLWLPRTY' yi kullanmayın. Consider using separate queues, or PENCERE ARALığı with a manual switch over from the primary to backup.

Kullanılan kanal sayısının sınırlandırılıyor

Çeşitli kuyruk yöneticilerine bir fiyat denetimi uygulaması kurulduğunda her bir sunucunun çalıştırıldığı etkin kanal sayısını sınırlamak için bu yönergeleri izleyin.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- Çeşitli kuyruk yöneticilerine bir fiyat denetimi uygulaması kurulabilecektir. Kullanılmakta olan kanalların sayısını düşük bir sayıya alıkoymak için, her bir sunucu çalıştırmasının etkin olduğu etkin kanal sayısı sınırlıdır. The application is driven by the arrival of messages on the PRICEQ queue.
- Dört sunucu kuyruğu yöneticisi fiyat denetimi uygulamasını barınabiliyor. İki sorgu kuyruğu yöneticisi, bir fiyatı sorgulamak için PRICEQ ' e ileti gönderir. İki kuyruk yöneticisi daha fazla havuz olarak yapılandırılıyor.

Bu görev hakkında

Kullanılan kanalların sayısını sınırlamak için aşağıdaki adımları izleyin.

Yordam

1. İki tam havuz seçin.

Fiyat denetimi kümeniz için tam havuz olmak üzere iki kuyruk yöneticisi seçin. Bunlar REPOS1 ve REPOS2 olarak adlandırılırlar.

Şu komutu verin:

```
ALTER QMGR REPOS(PRICECHECK)
```

2. Her kuyruk yöneticisi için bir CLUSTRVR kanalı tanımlayın.

Kümedeki her kuyruk yöneticisinde, bir küme alıcı kanalı ve bir küme gönderici kanalı tanımlayın. Önce ne tanımlı olduğu önemli değil.

```
DEFINE CHANNEL(PRICECHECK.SERVE1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)  
CONNNAME(SERVER1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-receiver channel')
```

3. Her kuyruk yöneticisinde bir CLUSSDR kanalı tanımlayın.

Kuyruk yöneticisini tüm havuz kuyruğu yöneticilerinden birine ya da birine bağlamak için her kuyruk yöneticisinde bir CLUSSDR tanımlaması yapın.

```
DEFINE CHANNEL(PRICECHECK.REPOS1) CHLTYPE(CLUSSDR) TRPTYPE(TCP)  
CONNNAME(REPOS1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-sender channel to  
repository queue manager')
```

4. Fiyat denetimi uygulamasını kurun.

5. Tüm sunucu kuyruğu yöneticilerindeki PRICEQ kuyruğunu tanımlayın.

Her biri için aşağıdaki komutu verin:

```
DEFINE QLOCAL(PRICEQ) CLUSTER(PRICECHECK)
```

6. Sorgular tarafından kullanılan kanal sayısını sınırla

Sorgu kuyruğu yöneticilerindeki, her birinde aşağıdaki komutları vererek, kullanılan etkin kanal sayısını kısıtlarız:

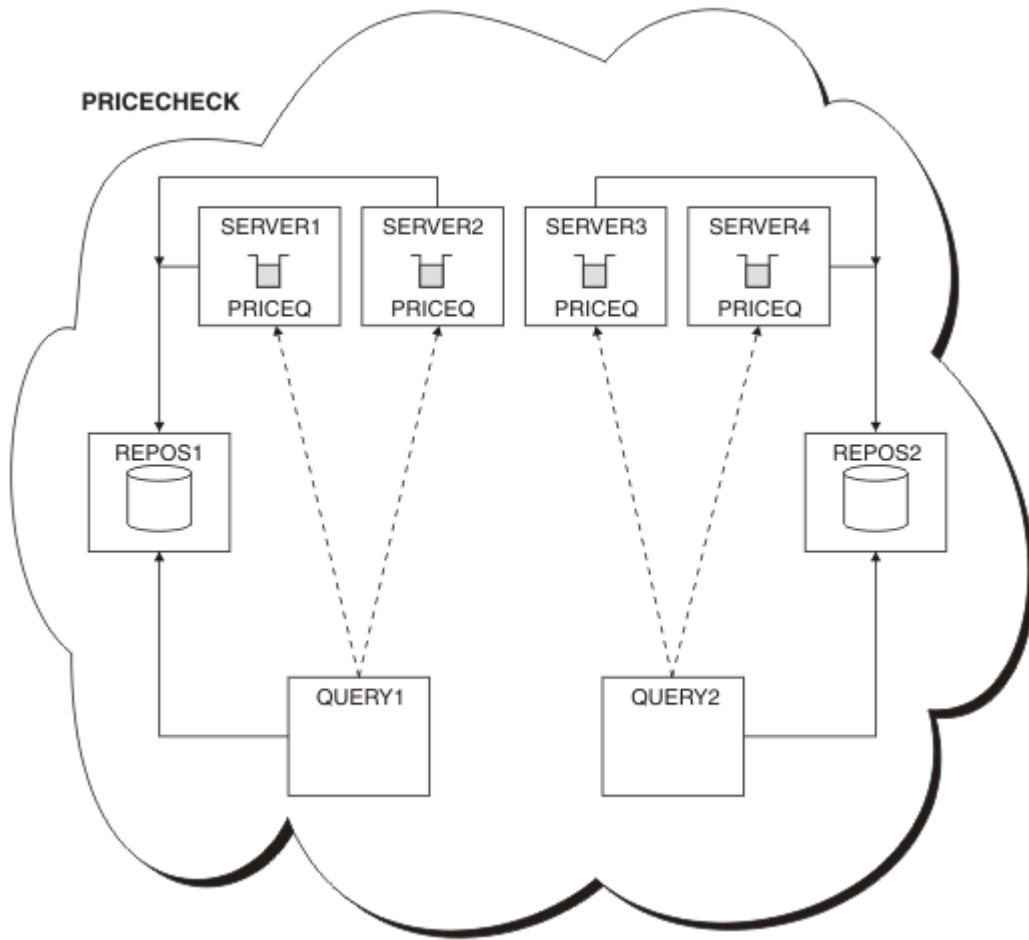
```
ALTER QMGR CLWLMRUC(2)
```

7. If you have not already done so, start the channel initiator on WebSphere MQ for z/OS. Tüm altyapılarda bir dinleyici programı başlatın.

Dinleyici programı gelen ağ isteklerini dinler ve gerektiğinde küme alıcı-alıcı kanalını başlatır.

Sonuçlar

[Şekil 53 sayfa 268](#) , bu görev tarafından ayarlanan kümeyi gösterir.



Şekil 53. Dört sunucu kuyruğu yöneticisi, iki havuz ve iki sorgu kuyruğu yöneticisi olan PRICECHECK kümesi

Although there are four instances of the PRICEQ queue available in the PRICECHECK cluster, each querying queue manager only uses two of two of them. Örneğin, QUERY1 kuyruk yöneticisinin SERVER1 ve SERVER2 kuyruk yöneticilerine yalnızca etkin kanalları vardır. SERVER1 kullanılamaz duruma geldiye, QUERY1 kuyruk yöneticisi başka bir kuyruk yöneticisini kullanmaya başlar (örneğin, SERVER3).

Sonraki adım

Although there are four instances of the PRICEQ queue available in the PRICECHECK cluster, each querying queue manager only uses two of two of them. Örneğin, QUERY1 kuyruk yöneticisinin SERVER1 ve SERVER2 kuyruk yöneticilerine yalnızca etkin kanalları vardır. SERVER1 kullanılamaz duruma geldiye, QUERY1 kuyruk yöneticisi başka bir kuyruk yöneticisini kullanmaya başlar (örneğin, SERVER3).

Kuyruk barındıran daha güçlü bir kuyruk yöneticisi eklenmesi

Los Angeles 'ın yanı sıra Los Angeles 'ın New York 'taki mesaj sayısını iki kez ele alabileceği New York 'un yanı sıra, envanter sistemini çalıştırarak ek kapasite sağlamak için bu yönergeleri izleyin.

Başlamadan önce

Not: Küme genelinde bir kümede geçirilecek değişiklikler için en az bir tam havuzun her zaman kullanılabilir olması gerekir. Bu görevi başlatmadan önce havuzlarınızın kullanılabilir olduğundan emin olun.

Senaryo:

- INVENTORY kümesi, “Küme için kuyruk yöneticisi eklenmesi” sayfa 191’inde açıklandığı şekilde ayarlanmıştır. It contains three queue managers: LONDON and NEWYORK both hold full repositories, PARIS holds a partial repository and puts messages from INVENTQ. Döküm uygulaması, NEWYORK kuyruk yöneticisine bağlı New York 'ta sistemde çalışır. The application is driven by the arrival of messages on the INVENTQ queue.
- Los Angeles 'ta yeni bir mağaza hazırlanıyor. Ek kapasite sağlamak için, New York 'un yanı sıra Los Angeles 'taki envanter sistemini de çalıştırmak istiyorsunuz. Yeni kuyruk yöneticisi, New York 'un iki katı kadar ileti işleyebilir.

Bu görev hakkında

Bir kuyruğu barındıran daha güçlü bir kuyruk yöneticisi eklemek için aşağıdaki adımları izleyin.

Yordam

1. LOSANGELES hangi tam havuzunun ilk önce başvurmaya karar vereceğine karar verin.
2. Bir kümedeki her kuyruk yöneticisinin, kümeyle ilgili bilgi toplamak için, tam havuzlardan birine ya da birbirine gönderme yapmalıdır. Kendi kısmi havuzunu oluşturur. Hangi havuzu seçtiğiniz belli bir öneme sahip değildir. Bu örnekte, NEWYORK seçilidir. Yeni kuyruk yöneticisi, her iki havuzla iletişim kurduğu kümeyle yeni bir kuyruk yöneticisi katıldıktan sonra.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from LOSANGELES to repository at NEWYORK')
```

3. Define the CLUSRCVR channel on queue manager LOSANGELES.

Bir kümedeki her kuyruk yöneticisinin, iletileri alabileceği bir küme alıcı kanalı tanımlamalıdır. LOSANGELES üzerinde şunları tanımlayın:

```
DEFINE CHANNEL(INVENTORY.LOSANGELES) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNNAME(LOSANGELES.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager LOSANGELES')
CLWLWGHT(2)
```

The cluster-receiver channel advertises the availability of the queue manager to receive messages from other queue managers in the cluster INVENTORY. CLWLWGHT ayarının iki değeri, Los Angeles kuyruk yöneticisinin New York (NEWYORK için kanal bir değere ayarlandığı zaman) kadar döküm iletilerinden iki kat daha fazla almasını sağlar.

4. Alter the CLUSRCVR channel on queue manager NEWYORK.

Los Angeles kuyruk yöneticisinin New York gibi döküm iletilerinden iki kat daha fazla aldığını doğrulayın. Küme alıcı kanalının tanımını değiştirin.

```
ALTER CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) CLWLWGHT(1)
```

5. İleti zenginlikleri için envanter uygulamasını gözden geçirin.

Devam etmeden önce, döküm uygulamasının iletilerin işlenmesi sırasında herhangi bir bağımlılıkları olmadığından emin olun.

6. Envanter uygulamasını Los Angeles 'taki sisteme kurun
7. Define the cluster queue INVENTQ.

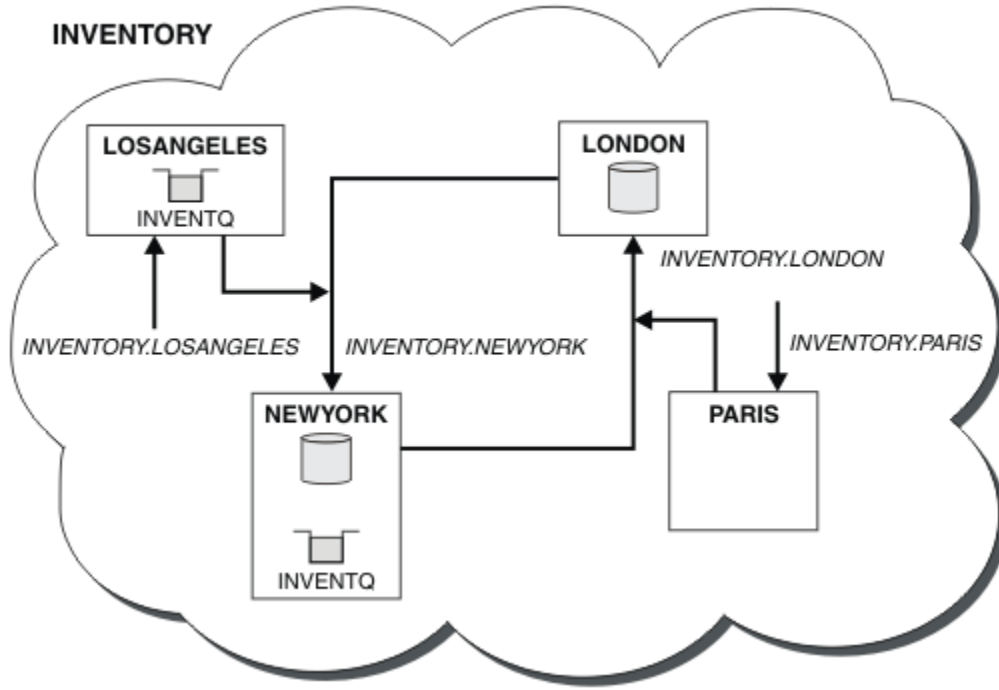
The INVENTQ queue, which is already hosted by the NEWYORK queue manager, is also to be hosted by LOSANGELES. Bunu LOSANGELES kuyruk yöneticisine aşağıdaki şekilde tanımlayın:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

Şimdi, tüm tanımları tamamladığınızı, henüz yapmadıysanız, kanal başlatıcısını WebSphere MQ for z/OS üzerinde başlatmış olmadıysanız. On all platforms, start a listener program on queue manager LOSANGELES. Dinleyici programı gelen ağ isteklerini dinler ve gerektiğinde küme alıcı-alıcı kanalını başlatır.

Sonuçlar

“Kuyruk barındıran daha güçlü bir kuyruk yöneticisi eklenmesi” sayfa 268 , bu görev tarafından ayarlanan kümeyi gösterir.



Şekil 54. Dört kuyruk yöneticisine sahip INVENTORY kümesi

This modification to the cluster was accomplished without you having to alter the queue managers LONDON and PARIS. The repositories in these queue managers are updated automatically with the information they need to be able to send messages to INVENTQ at LOSANGELES.

Sonraki adım

INVENTQ kuyruğu ve döküm uygulaması, kümedeki iki kuyruk yöneticisinde barındırılır. Yapılandırma, bunların kullanılabilirliğini artırır, iletilerin verimini hızlandırır ve iş yükünün iki kuyruk yöneticisi arasında dağıtılmasına olanak tanır. Messages put to INVENTQ by either LOSANGELES or NEWYORK are handled by the instance on the local queue manager whenever possible. Messages put by LONDON or PARIS are routed to LOSANGELES or NEWYORK, with twice as many messages being sent to LOSANGELES.

Uygulama programlama ve kümeler

Aynı kuyruğun birden çok örneğinden yararlanmak için herhangi bir programlama değişikliği yapmak zorunda değilsiniz. Ancak, bir kuyruğun aynı örneğine bir ileti dizisi gönderilmediği sürece bazı programlar doğru çalışmaz.

Uygulamalar MQOPEN çağrısını kullanarak bir kuyruk açabilir. Uygulamalar, iletileri açık bir kuyruğa yerleştirmek için MQPUT çağrısını kullanır. Uygulamalar, MQPUT1 çağrısını kullanarak, açık olmayan bir kuyruğa tek bir ileti yerleştirebilir.

Aynı kuyruğun birden çok eşgörünümü olan kümeleri ayarlıyorsanız, belirli bir uygulama programlama dikkate alınması gereken noktalar yoktur. Ancak, kümelemenin iş yükü yönetimi özelliklerinden yararlanabilmek için uygulamalarınızı değiştirmeniz gerekebilir. Aynı kuyruğun birden çok tanımının olduğu bir ağ ayarlıyorsanız, ileti zenginlikleri için uygulamalarınızı gözden geçirin.

Örneğin, soru ve yanıt formlarında aralarında akan bir dizi iletiye güvenen iki uygulamanız olduğunu varsayalım. Bir soru gönderen kuyruk yöneticisine geri dönmek için yanıtların büyük olasılıkla geri

gönderilmesini istiyorsunuz. İş yükü yönetimi yordamlarından, yanıt kuyruğunun bir kopyasını barındıran herhangi bir kuyruk yöneticisine ileti göndermemesi önemlidir.

İletilerin sırayla işlenmesini gerektiren uygulamalarınız olabilir (örneğin, sırayla alınması gereken iletilerin toplu iş iletilerini gönderen bir veritabanı eşleme uygulaması). Kesimlere ayrılmış iletilerin kullanımı, bir yakınlık sorununa da neden olabilir.

Hedef kuyruğun yerel ya da uzak bir sürümünün açılması

Kuyruk yöneticisinin hedef kuyruğun yerel ya da uzak bir sürümünün kullanılmasını nasıl seçeceğini dikkate al.

1. Kuyruk yöneticisi, iletileri okumak ya da kuyruğun özneliklerini ayarlamak için hedef kuyruğun yerel sürümünü açar.
2. Kuyruk yöneticisi, iletileri yazmak için hedef kuyruğun herhangi bir örneğini açar; aşağıdaki koşullardan en az biri doğru ise:
 - Hedef kuyruğun yerel bir sürümü yok.
 - Kuyruk yöneticisi, ALTER QMGRüzerinde CLWLUSEQ (ANY) değerini belirtiyor.
 - Kuyruk yöneticisindeki kuyruk CLWLUSEQ (ANY) değerini belirtiyor.

İleti zenginlikleri işleniyor

Mesaj zenginlikleri, nadiren iyi programlama tasarımının bir parçasıdır. Kümelemeyi tam olarak kullanmak için ileti yakınlığını kaldırmanız gerekir. İleti yakınlığını kaldıramıyorsanız, ilgili iletileri aynı kanal kullanılarak ve aynı kuyruk yöneticisine teslim etmek için zorlayabilirsiniz.

İleti zenginlikleri olan uygulamalarınız varsa, kümeleri kullanmaya başlamadan önce zenginlikleri kaldırın.

İleti zenginliğinin kaldırılması uygulamaların kullanılabilirliğini artırır. Bir uygulama, kuyruk yöneticisine ileti yakınlıkları olan bir ileti kümesi gönderir. Toplu işin yalnızca bir kısmını aldıktan sonra kuyruk yöneticisi başarısız olur. Gönderme kuyruk yöneticisi, daha fazla ileti göndermeden önce tamamlanmamış ileti kümesini kurtarmasını ve işlenmesini beklemelidir.

İletilerin bağlarını kaldırma, uygulamaların ölçeklenebilirliğini de geliştirir. Varlıklı iletiler içeren bir ileti kümesi, sonraki iletileri beklerken hedef kuyruk yöneticisinde kaynakları kilitleyebilir. Bu kaynaklar uzun süre kilitli kalabilir ve diğer uygulamaların çalışmalarını önlemektedir.

Ayrıca, ileti zenginlikleri, küme iş yükü yönetimi yordamlarının kuyruk yöneticisi seçimini en iyi şekilde yapmasını engeller.

Yakınlıkları kaldırmak için aşağıdaki olasılıkları göz önünde bulundurun:

- İletilerde durum bilgisi taşıma
- Herhangi bir kuyruk yöneticisinde (örneğin, bir Db2 veritabanında), geçici olmayan depolamada durum bilgilerinin sağlanması
- Birden çok kuyruk yöneticisi tarafından erişilebilir olması için salt okunur verileri eşleyen

Uygulamalarınızı ileti yakınlıkları kaldırmak için değiştirmek uygun değilse, soruna olası bir dizi çözüm vardır.

MQOPEN çağrısında belirli bir hedef adı yazın

Her bir MQOPEN çağrısında uzak kuyruk adını ve kuyruk yöneticisi adını belirtin ve o nesne tanıtıcısını kullanarak kuyruğa konan tüm iletiler aynı kuyruk yöneticisine (yerel kuyruk yöneticisi olabilir) gidin.

Her bir MQOPEN çağrısında uzak kuyruk adının ve kuyruk yöneticisi adının belirtilmesine ilişkin dezavantajları vardır:

- İş yükü dengelemesi gerçekleştirilmez. Küme iş yükü dengelemesinin avantajlarından yararlanmanıza gerek yoktur.
- Hedef kuyruk yöneticisi uzaksa ve birden çok kanal varsa, iletiler farklı rotalar alabilir ve ileti dizisi yine de korunmaz.

- Kuyruk yöneticinizin, hedef kuyruk yöneticisiyle aynı adı taşıyan bir iletim kuyruğu tanımlaması varsa, iletiler küme iletim kuyruğunda değil, o iletim kuyruğunda bulunur.

Yanıtlama kuyruğu yöneticisi alanında kuyruk yöneticisi adını döndürür

Bir toplu işteki ilk iletiyi alan kuyruk yöneticisine, kendi yanıtında adını döndürmesine izin verin. Bu, ileti tanımlayıcısının ReplyToQMgr alanını kullanarak yapar. Gönderme sonundaki kuyruk yöneticisi, yanıt kuyruğu yöneticisi adını çıkarabilir ve sonraki tüm iletilerde belirtebilirler.

Yanıttan ReplyToQMgr bilgilerinin kullanılması dezavantajlara neden olur:

- İstekte bulunan kuyruk yöneticisi, ilk iletilerine yanıt için beklemelidir
- Sonraki iletileri göndermeden önce ReplyToQMgr bilgilerini bulmak ve kullanmak için ek kod yazmanız gerekir.
- Kuyruk yöneticisine birden çok rota varsa, iletilerin sırası korunmayabilir.

MQOPEN çağrısında MQOO_BIND_ON_OPEN seçeneğini ayarlayın

Tüm iletilerinizin MQOPEN çağrısında MQOO_BIND_ON_OPEN seçeneğini kullanarak aynı hedefe konmasını zorunlu kılabilir. Gruptaki tüm iletilerin aynı hedefte işlendiğinden emin olmak için kümeler ile ileti grupları kullanıldığında MQOO_BIND_ON_OPEN ya da MQOO_BIND_ON_GROUP belirtilmelidir.

Bir kuyruk açıp MQOO_BIND_ON_OPEN belirtilerek, bu kuyruğa gönderilen tüm iletilerin kuyruğun aynı örneğine gönderilmesini zorunlu kısıyorsunuz. MQOO_BIND_ON_OPEN , tüm iletileri aynı kuyruk yöneticisine ve aynı rotada bağlar. Örneğin, aynı hedefe yönelik bir IP rotası ve bir NetBIOS rotası varsa, kuyruk açıldığında bunlardan biri seçilir ve bu seçim, elde edilen nesne tanıtıcısı kullanılarak aynı kuyruğa konması için bu seçim onurlandırılır.

MQOO_BIND_ON_OPEN belirtildiğinde, tüm iletilerin aynı hedefe yönlendirilmesini zorunlu kısıyorsunuz. Bu nedenle, ileti zenginlikleri olan uygulamalar kesintiye uğramaz. Hedef kullanılamıyorsa, iletiler yeniden kullanılabilir duruma gelinceye kadar iletim kuyruğunda kalır.

MQOO_BIND_ON_OPEN ayrıca, kuyruk açtığınızda nesne tanımlayıcısında kuyruk yöneticisi adı belirtildiğinde de geçerlidir. Adı belirtilen kuyruk yöneticisine birden çok rota olabilir. Örneğin, birden çok ağ yolu olabilir ya da başka bir kuyruk yöneticisi bir diğer ad tanımlanmış olabilir. MQOO_BIND_ON_OPEN belirtilirse, kuyruk açıldığında bir rota seçilir.

Not: Bu önerilen teknik. Ancak, kuyruk yöneticisinin bir küme kuyruğu için bir diğer adı ilan ettiği çoklu sekme yapılandırmalarında çalışmaz. Ayrıca, uygulamaların farklı ileti grupları için aynı kuyruk yöneticisinde farklı kuyruklar kullanmalarına da yardımcı olmaz.

MQOPEN çağrısında MQOO_BIND_ON_OPEN belirtilmesine alternatif olarak, kuyruk tanımlarınızı değiştirmeniz gerekir. Kuyruk tanımlarınızda, DEFBIND (OPEN) değerini belirtin ve MQOPEN çağrısındaki DefBind seçeneğinin varsayılan olarak MQOO_BIND_AS_Q_DEF' a izin vermelerini sağlayın.

MQOPEN çağrısında MQOO_BIND_ON_GROUP seçeneğini ayarlayın

Bir gruptaki tüm iletilerinizi, MQOPEN çağrısındaki MQOO_BIND_ON_GROUP seçeneğini kullanarak aynı hedefe yerleştirmek için zorlayın. Gruptaki tüm iletilerin aynı hedefte işlendiğinden emin olmak için kümeler ile ileti grupları kullanıldığında MQOO_BIND_ON_OPEN ya da MQOO_BIND_ON_GROUP belirtilmelidir.

Bir kuyruk açıp MQOO_BIND_ON_GROUP belirtilerek, bu kuyruğa gönderilen bir gruptaki tüm iletileri, kuyruğun aynı örneğine gönderilmek üzere zorlayın. MQOO_BIND_ON_GROUP , bir gruptaki tüm iletileri aynı kuyruk yöneticisine ve aynı yola bağlar. Örneğin, aynı hedefe yönelik bir IP rotası ve bir NetBIOS rotası varsa, kuyruk açıldığında bunlardan biri seçilir ve bu seçim, bir gruptaki tüm iletilerin, elde edilen nesne tanıtıcısını kullanarak aynı kuyruğa konması için bir onur olur.

MQOO_BIND_ON_GROUP belirtildiğinde, bir gruptaki tüm iletileri aynı hedefe yönlendirilmeye zorlayın. Bu nedenle, ileti zenginlikleri olan uygulamalar kesintiye uğramaz. Hedef kullanılamıyorsa, iletiler yeniden kullanılabilir duruma gelinceye kadar iletim kuyruğunda kalır.

MQ00_BIND_ON_GROUP ayrıca, kuyruk açtığınızda nesne tanımlayıcısında kuyruk yöneticisi adı belirtildiğinde de geçerlidir. Adı belirtilen kuyruk yöneticisine birden çok rota olabilir. Örneğin, birden çok ağ yolu olabilir ya da başka bir kuyruk yöneticisi bir diğer ad tanımlamış olabilir. MQ00_BIND_ON_GROUP belirtilirse, kuyruk açıldığında bir rota seçilir.

MQ00_BIND_ON_GROUP ' in etkili olması için, MQPUT üzerine MQPMO_LOGICAL_ORDER put seçeneğini eklemelisiniz. You can set **GroupId** in the MQMD of the message to MQGI_NONE, and you must include the following message flags within the MQMD **MsgFlags** field of the messages:

- Gruptaki son ileti: MQMF_LAST_MSG_IN_GROUP
- Gruptaki diğer tüm iletiler: MQMF_MSG_IN_GROUP

MQ00_BIND_ON_GROUP belirtilirse, ancak iletiler gruplanmadıysa, davranış MQ00_BIND_NOT_FIXED değerine eşdeğerdir.

Not: Bu, bir gruptaki iletilerin aynı hedefe gönderildiğinden emin olmak için önerilen bir tekniktir. Ancak, kuyruk yöneticisinin bir küme kuyruğu için bir diğer adı ilan ettiği çoklu sekme yapılandırmalarında çalışmaz.

MQOPEN çağrısında MQ00_BIND_ON_GROUP belirtilmesine alternatif olarak, kuyruk tanımlarınızı değiştirmeniz gerekir. Kuyruk tanımlarınızda, DEFBIND (GROUP) değerini belirtin ve MQOPEN çağrısındaki DefBind seçeneğinin varsayılan olarak MQ00_BIND_AS_Q_DEF' a izin vermelerini sağlayın.

Özelleştirilmiş bir küme iş yükü çıkış programı yazın

Uygulamalarınızı değiştirmek yerine, bir küme iş yükü çıkış programı yazarak ileti yakınlıkları sorununu atlatabilirsiniz. Bir küme iş yükü çıkış programı yazmak kolay değildir ve önerilen bir çözüm değildir. Programın, iletilerin içeriğini inceleyerek benzeşimi tanımak için tasarlanacağı yer. Benzeşimi tanıdıysanız, program, iş yükü yönetimi yardımcı programını tüm ilgili iletileri aynı kuyruk yöneticisine yöneltmek için zorlamalı.

Kümeleme: En iyi uygulamalar

Kümeleme, kuyruk yöneticilerine bağlanmak için bir mekanizma sağlar. Bu bölümde açıklanan en iyi uygulamalar, müşterilerden gelen test ve geri bildirimlere dayanır.

Başarılı bir küme kurulumu, iyi bir planlama ve iyi uygulama yönetimi ve ağ tasarımı gibi temel IBM WebSphere MQ temel kavramlarıyla ilgili ayrıntılı bir anlayışla bağlıdır. Devam etmeden önce, aşağıda listelenen ilgili konulardaki bilgileri aşina olduğunuzu doğrulayın.

İlgili kavramlar

[Kümeleme](#)

[İletişim kavramları](#)

[Kümeleme nasıl çalışır](#)

Kümeleme: çakışan kümeler için dikkat edilmesi gereken özel noktalar

Bu konuda, IBM WebSphere MQ kümelerini planlamaya ve yönetmeye ilişkin yönergeler sağlanır. Bu bilgiler, müşterilerden gelen testlere ve geribildirime dayalı bir kılavuzdur.

Küme sahipliği

Aşağıdaki bilgileri okumadan önce çakışan kümeleri tanıyın. Gerekli bilgiler için bkz. "[Çakışan kümeler](#)" sayfa 176 ve "[Kümeleme arasındaki ileti yollarının yapılandırılması](#)" sayfa 253 .

Çakışan kümelerden oluşan bir sistemi yapılandırırken ve yönetirken, en iyisi aşağıdakilere uymaktır:

- IBM WebSphere MQ kümeleri daha önce açıklandığı gibi 'gevşek bir şekilde eşlenmiş' olsalar da, bir kümeyi tek bir yönetim birimi olarak kabul etmek yararlıdır. Bu kavram, tek tek kuyruk yöneticilerindeki tanımlamalar arasındaki etkileşim, kümenin düzgün çalışması için kritik önem taşıdığından kullanılır. Örneğin: İş yükü dengeli küme kuyrukları kullanılırken, tek bir yöneticinin ya da ekibin, küme genelinde

dağıtılan tanımlamalara bağlı olarak iletilere ilişkin olası hedeflerin tam kümesini anlaması önemlidir. Daha önemsiz olarak, küme gönderen/alıcı kanal çiftleri boyunca uyumlu olmalıdır.

- Bu önceki kavram göz önünde bulundurulur; birden çok kümenin (ayrı ekipler/kişiler tarafından yönetileceği) bir araya geldiği durumlarda, ağ geçidi kuyruk yöneticilerinin yerinde denetim yönetiminde net ilkelere sahip olmak önemlidir.
- Çakışan kümeleri tek bir ad alanı olarak kabul etmek yararlıdır: Kanal adları ve kuyruk yöneticisi adları tek bir küme boyunca benzersiz olmalıdır. Yönetim, topolojinin tamamında benzersiz olduğunda çok daha kolaydır. Uygun bir adlandırma kuralına uymanız en iyi yoldur; olası kurallar "[Küme adlandırma kuralları](#)" sayfa 175 içinde açıklanmıştır.
- Bazen idari ve sistem yönetimi işbirliği çok önemlidir/önlenebilir: Örneğin, örtüşmesi gereken farklı kümelere sahip kuruluşlar arasında işbirliği. Kimin neye sahip olduğuna ilişkin net bir anlayış ve uygulanabilir kurallar/kurallar, kümelemenin çakışan kümeler sırasında sorunsuz bir şekilde çalışmasına yardımcı olur.

Çakışan kümeler: Ağ Geçitleri

Genel olarak, tek bir kümeyi yönetmek birden çok kümeye göre daha kolaydır. Bu nedenle çok sayıda küçük küme oluşturmak (örneğin her uygulama için bir tane) genel olarak kaçınılması gereken bir şeydir.

Ancak, hizmet sınıfları sağlamak için çakışan kümeler uygulayabilirsiniz. Örneğin:

- Daha küçük olanın Yayınlama/Abone Olma için olduğu eş merkezli kümeleriniz varsa. Ek bilgi için [Sistemleri nasıl boyutlandırılır](#) başlıklı konuya bakın.
- Bazı kuyruk yöneticileri farklı ekipler tarafından yönetiliyorsa. Daha fazla bilgi için önceki "[Küme sahipliği](#)" sayfa 273 bölümüne bakın.
- Bir organizasyonel ya da coğrafi bakış açısından mantıklı ise.
- Eşdeğer kümeler ad çözümlemesi ile çalışıyorsa, örneğin var olan bir kümede SSL ya da TLS uygulanıyor olabilir.

Örtüşen kümelere güvenlik avantajı yoktur; iki farklı ekip tarafından yönetilen kümelerin örtüşmesine izin verir, topolojinin yanı sıra ekiplere de etkili bir şekilde katılır. Herhangi biri:

- Böyle bir kümede duyurulan ad, diğer küme tarafından erişilebilir.
- Bir kümede duyurulan ad, uygun mesajları çekmek için diğerinde duyurular olabilir.
- Ağ geçidinin bitişiğindeki bir kuyruk yöneticisinde duyurulmayan nesne, ağ geçidinin üyesi olduğu kümelere çözümenebilir.

Ad alanı, her iki kümenin birleşmesi ve tek bir ad alanı olarak ele alınmalıdır. Bu nedenle, çakışan bir kümenin sahipliği her iki kümenin tüm yöneticileri arasında paylaşılır.

Bir sistem birden çok küme içerdiğinde, iletileri bir kümedeki kuyruk yöneticilerinden başka bir kümedeki kuyruk yöneticilerine yönlendirme gereksinimi olabilir. Bu durumda, birden çok kümenin bir şekilde birbirine bağlanması gerekir: İzlenmesi gereken iyi bir kalıp, kümeler arasında ağ geçidi kuyruk yöneticilerinin kullanılmasıdır. Bu düzenleme, noktadan noktaya iletişim kanallarının yönetilmesi zor bir ağ oluşturulmasını önler ve güvenlik politikaları gibi sorunları yönetmek için iyi bir yer sağlar. Bu düzenlemeyi gerçekleştirmenin iki farklı yolu vardır:

1. İkinci bir küme alıcı tanımlaması kullanarak her iki kümeye bir (ya da daha çok) kuyruk yöneticisi yerleştirin. Bu düzenleme daha az yönetim tanımı içerir, ancak daha önce belirtildiği gibi, çakışan bir kümenin sahipliğinin her iki kümenin tüm yöneticileri arasında paylaşıldığı anlamına gelir.
2. Küme 1 'deki bir kuyruk yöneticisini, geleneksel noktadan noktaya kanalları kullanarak küme teo içindeki bir kuyruk yöneticisiyle eşleştirin.

Bu iki durumda da, trafiği uygun şekilde yönlendirmek için çeşitli araçlar kullanılabilir. Özellikle, kuyruk ya da kuyruk yöneticisi diğer adları diğer kümeye yönelmek için kullanılabilir ve boş **RQMNAME** özelliğine sahip bir kuyruk yöneticisi diğer adı, iş yükü dengelemesini istendiği yerde yeniden yönlendirir.

İlgili kavramlar

["Küme adlandırma kuralları" sayfa 175](#)

Kuyruk yöneticisinin ait olduğu kümeyi tanıtan bir adlandırma kuralı kullanarak aynı kümedeki kuyruk yöneticilerini adlandırmayı düşünün. Kanal adları için benzer bir adlandırma kuralı kullanın ve kanal özelliklerini açıklayacak şekilde genişletin.

Kümeleme: Topoloji tasarımıyla ilgili dikkat edilecek noktalar

Bu konuda, IBM WebSphere MQ kümelerini planlamaya ve yönetmeye ilişkin yönergeler sağlanır. Bu bilgiler, müşterilerden gelen testlere ve geribildirime dayalı bir kılavuzdur.

Kullanıcı uygulamalarının ve iç idari süreçlerin önceden nerede bulunacağı düşünülüp, birçok sorun ya önlenir ya da daha sonraki bir tarihte en aza indirilebilir. Bu konu, performansı artırabilen tasarım kararlarıyla ilgili bilgileri içerir ve küme ölçeklendirdikçe bakım görevlerini basitleştirir.

- [“Kümeleme altyapısının performansı” sayfa 275](#)
- [“Tam havuzlar” sayfa 276](#)
- [“Uygulamalar tam havuzlardaki kuyrukları kullanmalı mı?” sayfa 276](#)
- [“Kanal tanımlarının yönetilmesi” sayfa 277](#)
- [“Birden çok kanal üzerinden iş yükü dengeleme” sayfa 277](#)

Kümeleme altyapısının performansı

Bir uygulama kümedeki bir kuyruk yöneticisinde bir kuyruk açmaya çalışıldığında, kuyruk yöneticisi, kuyruğun kümede nerede bulunduğunu öğrenmesi için ilgili kuyruğun tüm havuzlarına ilgisini kaydeder. Kuyruk konumu ya da yapılandırmasında yapılan güncellemeler, tüm havuzlar tarafından otomatik olarak ilgili kuyruk yöneticisine gönderilir. İlginizi çeken bu kayıt, dahili olarak abonelik olarak bilinir (bu abonelikler, IBM WebSphere MQ’ünde yayınlama/abone olma ileti sistemi için kullanılan IBM WebSphere MQ abonelikleriyle aynı değildir)

Bir kümeyle ilgili tüm bilgiler her tam havuzdan geçer. Bu nedenle, tam havuzlar her zaman yönetimle ilgili ileti trafiği için bir kümede kullanılır. Bu abonelikleri yönetirken sistem kaynaklarının yüksek kullanımı ve bunların iletimi ve sonuçta ortaya çıkan yapılandırma iletileri, kümeleme altyapısında önemli bir yüke neden olabilir. Bu yükün anlaşılmasını ve mümkün olan yerlerde en aza indirilmesini sağlarken göz önünde bulundurulması gereken birçok şey vardır:

- Küme kuyruğunu kullanan tek tek kuyruk yöneticileri ne kadar çok olursa, sistemde o kadar çok abonelik olur ve bu nedenle, değişiklikler oluştuğunda ve ilgili abonelere, özellikle de tam havuz kuyruğu yöneticilerine ilişkin bildirim gönderilmesi gereken yönetim ek yükü o kadar büyük olur. Gereksiz trafiği ve tam havuz yükünü en aza indirmenin bir yolu, benzer uygulamaları (yani aynı kuyruklarla çalışan uygulamaları) daha az sayıda kuyruk yöneticilerine bağlamaktır.
- Sistemdeki abonelik sayısının başarımı etkilemesine ek olarak, kümelenmiş nesnelere konfigürasyonundaki değişiklik hızı başarımı etkileyebilir; örneğin, kümelenmiş bir kuyruk konfigürasyonunun sık sık değiştirilmesi gibi.
- Bir kuyruk yöneticisi birden çok kümenin üyesiye (yani, çakışan bir küme sisteminin parçasıysa), aynı kuyruk yöneticileri birden çok kümenin tam havuzları olsa bile, bir kuyruktaki herhangi bir ilgi, üyesi olduğu her küme için bir abonelikle sonuçlanır. Bu düzenleme, sistemdeki yükü artırır ve tek bir küme yerine birden çok çakışan kümenin gerekli olup olmadığını düşünmek için bir nedendir.
- Uygulama ileti trafiği (yani, IBM WebSphere MQ uygulamaları tarafından küme kuyruklarına gönderilen iletiler), hedef kuyruk yöneticilerine ulaşmak için tam havuzlar üzerinden gitmez. Bu ileti trafiği, doğrudan iletinin kümeye girdiği kuyruk yöneticisi ile küme kuyruğunun bulunduğu kuyruk yöneticisi arasında gönderilir. Bu nedenle, tam havuz kuyruğu yöneticileri söz konusu iki kuyruk yöneticisinden biri olmadığı sürece, tam havuz kuyruğu yöneticilerine göre uygulama ileti trafiğinin yüksek oranlarına uyulması gerekli değildir. Bu nedenle, kümeleme altyapısı yükünün önemli olduğu kümelerdeki uygulama ileti trafiği için tam havuz kuyruğu yöneticilerinin kullanılmaması önerilir.

Tam havuzlar

Havuz, bir kümenin üyesi olan kuyruk yöneticilerine ilişkin bilgilerin bir toplamıdır. Kümedeki her kuyruk yöneticisine ilişkin eksiksiz bir bilgi kümesini barındıran bir kuyruk yöneticisi tam bir havuza sahiptir. Tam havuzlar ve kısmi havuzlar hakkında daha fazla bilgi için bkz. ["Küme havuzu" sayfa 161.](#)

Güvenilir ve yüksek düzeyde kullanılabilir olan sunucularda tam havuzlar tutulmalıdır ve tek hata noktalarından kaçınılmalıdır. Küme tasarımının her zaman iki tam havuzu olmalıdır. Tam havuzda bir hata oluşursa, küme çalışmaya devam edebilir.

Kümedeki bir kuyruk yöneticisi tarafından yapılan küme kaynaklarında yapılan güncellemelerin ayrıntıları; örneğin, kümelenecek kuyruklar, o kümedeki en çok iki tam havuza (ya da kümede yalnızca bir tam havuz kuyruk yöneticisi varsa) gönderilir. Bu tam havuzlar, bilgileri tutar ve kümede ilgili olan herhangi bir kuyruk yöneticisine geçirir (yani, ona abone olurlar). Kümenin her üyesinin küme kaynaklarının güncel bir görünümüne sahip olmasını sağlamak için, her kuyruk yöneticisinin bir kerede en az bir tam havuz kuyruk yöneticisiyle iletişim kurulması gerekir.

Herhangi bir nedenle, bir kuyruk yöneticisi herhangi bir tam havuzla iletişim kuramıyorsa, bir dönem için önceden önbelleğe alınmış bilgi düzeyine dayalı olarak kümede çalışmaya devam edebilir, ancak daha önce kullanılmamış küme kaynaklarına yeni güncelleme ya da erişim sağlanmaz.

Bu nedenle, iki tam havuzu her zaman kullanılabilir tutmayı hedeflemelisiniz. Bununla birlikte, bu düzenleme, kümenin tam bir havuz olmadan kısa bir süre için yeterli şekilde işlev göstermesi nedeniyle aşırı önlemlerin alınması gerektiği anlamına gelmez.

Bir kümenin, küme bilgilerinin kullanılabilirliği dışında iki tam havuz kuyruğu yöneticisinin olması gerekmesinin başka bir nedeni vardır: Bu neden, kurtarma amacıyla, tam havuz önbelleğinde tutulan küme bilgilerinin iki yerde bulunduğundan emin olmaktır. Yalnızca bir tam havuz varsa ve kümeyle ilgili bilgilerini kaybederse, kümenin yeniden çalışması için küme içindeki tüm kuyruk yöneticilerine el ile müdahale gerekir. Bununla birlikte, iki tam havuz varsa, bilgiler her zaman iki tam havuza yayınlanır ve bunlara abone olduğundan, başarısız olan tam havuz en az çabayla kurtarılabilir.

- Bu kümenin kullanıcılarını etkilemeden, iki tam havuz kümesi tasarımında tam havuz kuyruğu yöneticilerine bakım gerçekleştirilebilir: Küme, yalnızca bir havuzla çalışmaya devam eder; böylece, mümkün olan yerlerde havuzları aşağı indirir, bakım uygular ve her defasında bir tane yedekler. İkinci tam havuzda bir kesinti olsa bile, çalışan uygulamalar en az üç gün boyunca etkilenmez.
- Coğrafi nedenlerle coğrafi olarak yerel bir tam havuz kullanmak gibi üçüncü bir havuz kullanmak için iyi bir neden yoksa, iki havuz tasarımını kullanın. Üç tam havuz olması, şu anda kullanımda olan ikisinin hangilerinin kullanıldığını hiçbir zaman bilmediğiniz anlamına gelir ve birden çok iş yükü yönetimi parametresi arasındaki etkileşimlerin neden olduğu yönetimle ilgili sorunlar olabilir. İkide fazla tam havuz olması önerilmez.
- Hala daha iyi kullanılabilirliğe gereksiniminiz varsa, tüm havuz kuyruğu yöneticilerini çok eşgörünümlü kuyruk yöneticileri olarak barındırmayı ya da kullanılabilirliği artırmak için platforma özgü yüksek kullanılabilirlik desteğini kullanmayı düşünün.
- El ile tanımlanan küme gönderen kanallarıyla tüm havuz kuyruğu yöneticilerini tam olarak birbirine bağlamanız gerekir. Küme, haklı bir nedenle ikiden fazla tam havuza sahip olduğunda özellikle dikkatli olunmalıdır. Bu durumda genellikle bir veya daha fazla kanalı gözden kaçırmak ve hemen anlaşılması mümkün değildir. Tam ara bağlantı gerçekleşmediğinde, sorunların tanınması genellikle zordur. Bazı tam havuzlar tüm havuz verilerini tutmadığından ve bu nedenle bağlı oldukları tam havuzlara bağlı olarak kümedeki kuyruk yöneticilerinin farklı görünümlere sahip olmalarına neden olduğundan, bu havuzların tanınması zordur.

Uygulamalar tam havuzlardaki kuyrukları kullanmalı mı?

Tam havuz çoğu şekilde diğer kuyruk yöneticilerine benzer ve bu nedenle uygulama kuyruklarını tam havuzda barındırmak ve uygulamaları doğrudan bu kuyruk yöneticilerine bağlamak mümkündür. Uygulamalar tam havuzlardaki kuyrukları kullanmalı mı?

Yaygın olarak kabul edilen cevap "Hayır?" dir. Bu yapılandırma mümkün olsa da, birçok müşteri bu kuyruk yöneticilerini tam havuz kümesi önbelleğinin bakımı için özel olarak ayırmayı tercih eder. Her iki seçeneğe

karar verirken dikkate alınması gereken noktalar burada açıklanmıştır, ancak sonuçta küme mimarisi ortamın belirli taleplerine uygun olmalıdır.

- Yükseltmeler: Genellikle, IBM WebSphere MQ ' un yeni yayınlarında yeni küme özelliklerini kullanmak için öncelikle o kümenin tam havuz kuyruğu yöneticilerinin yükseltilmesi gerekir. Kümedeki bir uygulama yeni özellikler kullanmak istediğinde, birlikte bulunan bir dizi uygulamayı test etmeden tam havuzları (ve kısmi havuzların bazı alt kümelerini) güncellemek yararlı olabilir.
- Bakım: Benzer bir şekilde, tam havuzlara acil bakım uygulamanız gerekiyorsa, bunlar **REFRESH** komutuyla uygulamalara dokunmadan yeniden başlatılabilir ya da yenilenebilir.
- Performans: Kümeler büyüdükçe ve tam havuz kümesi önbelleği bakımına ilişkin talepler büyüdükçe, uygulamaların ayrı tutulması, sistem kaynakları için çekişme yoluyla uygulama performansını etkileyen riski azaltır.
- Donanım gereksinimleri: Tipik olarak, tam havuzların güçlü olması gerekmez; örneğin, iyi bir kullanılabilirlik beklentisi olan basit bir UNIX sunucusu yeterlidir. Alternatif olarak, çok büyük veya sürekli değişen kümeler için, tam havuz bilgisayarının performansı dikkate alınmalıdır.
- Yazılım gereksinimleri: Gereksinimler genellikle uygulama kuyruklarını tam bir havuzda barındırmak için ana nedendir. Küçük bir kümede, iş birliği (collocation), tüm kuyruk yöneticilerinin/sunucularının daha az olması için bir gereksinim anlamına gelebilir.

Kanal tanımlarının yönetilmesi

Tek bir küme içinde bile, iki kuyruk yöneticisi arasında birden çok rota sağlayan birden çok kanal tanımı olabilir.

Bazen tek bir küme içinde paralel kanallara sahip olmanın bir avantajı vardır, ancak bu tasarım kararının iyice düşünülmesi gerekir; karmaşıklık eklemenin yanı sıra, bu tasarım, performansı azaltan kanalların yeterince kullanılmamasına neden olabilir. Bu durum, testlerin genellikle sabit bir hızda çok sayıda ileti göndermeyi içermesi nedeniyle ortaya çıkar, bu nedenle paralel kanallar tamamen kullanılır. Ancak sürekli olmayan ileti akışının gerçek dünya koşullarıyla, iş yükü dengeleme algoritması, ileti akışı kanaldan kanala geçildikçe performansın düşmesine neden olur.

Bir kuyruk yöneticisi birden çok kümenin üyesiye, her küme için ayrı bir CLUSRCVR kanalı tanımlamak yerine küme ad listesi olan tek bir kanal tanımı kullanma seçeneği vardır. Ancak, bu kurulum daha sonra yönetim sorunlarına neden olabilir; örneğin, SSL ' nin bir kümeye uygulanacağı, ancak bir saniyeye uygulanmayacağını göz önünde bulundurun. Bu nedenle ayrı tanımlar yaratılması tercih edilir ve [“Küme adlandırma kuralları” sayfa 175](#) içinde önerilen adlandırma kuralı bunu destekler.

Birden çok kanal üzerinden iş yükü dengeleme

Bu bilgi, konunun gelişmiş bir şekilde anlaşılması amacıyla hazırlanmıştır. Bu konuya ilişkin temel açıklamalar için (buradaki bilgileri kullanmadan önce anlaşılması gereken) bkz. [“İş yükü yönetimi için kümeleri kullanma” sayfa 258](#), [İş yükü dengeleme](#) ve [Küme iş yükü yönetimi algoritması](#).

Küme iş yükü yönetimi algoritması büyük bir araç kümesi sağlar, ancak bunların tümü nasıl çalıştıklarını ve etkileşimde bulduklarını tam olarak anlamadan birbirleriyle birlikte kullanılmamalıdır. İş yükü dengeleme süreci için kanalların ne kadar önemli olduğu hemen belli olmayabilir: İş yükü yönetimi çevrimsel sıralı algoritması, kümelenebilir bir kuyruğu olan bir kuyruk yöneticisine birden çok küme kanalı gibi davranır ve bu kuyruğun birden çok eşgörünümü olarak kabul edilir. Bu işlem aşağıdaki örnekte daha ayrıntılı olarak açıklanmıştır:

1. Bir kümede kuyruk barındıran iki kuyruk yöneticisi vardır: QM1 ve QM2.
2. QM1 için beş küme alıcı kanalı vardır.
3. QM2 için yalnızca bir küme alıcı kanalı vardır.
4. QM3 üzerinde **MQPUT** ya da **MQOPEN** bir eşgörünüm seçtiğinde, algoritmanın iletiyi QM1 'e gönderme olasılığı QM2' e göre beş kat daha fazladır.
5. 4 numaralı adımdaki durum, algoritma beş kanaldan QM1 'e ve tek kanaldan QM2' e kadar altı seçeneği (5 + 1) ve yuvarlamalı robinler arasından seçim yapabileceği için ortaya çıkar.

Diğer bir ince davranış, iletileri yerel kuyruk yöneticisinde tek bir yönetim ortamının yapılandırılmış olduğu kümelenmiş bir kuyruğa koyarken bile, IBM WebSphere MQ iletilerin kuyruğun yerel eşgörünümüne mi, yoksa kuyruğun uzak eşgörünümüne mi konacağına karar vermek için yerel küme alıcı kanalının durumunu kullanmasıdır. Bu senaryoda:

1. İş yükü yönetimi algoritmasının tek tek küme kuyruklarına bakmadığı iletiler konurken, bu hedeflere ulaşabilen küme kanallarına bakar.
2. Yerel hedeflere ulaşmak için, yerel alıcı kanalları bu listeye eklenir (ileti göndermek için kullanılmamalarına rağmen).
3. Bir yerel alıcı kanal durdurulduğunda, iş yükü yönetimi algoritması, CLUSRCVR durdurulmamışsa varsayılan olarak alternatif bir örneği tercih eder. Hedef için birden çok yerel CLUSRCVR yönetim ortamı varsa ve en az biri durdurulmazsa, yerel yönetim ortamı uygun kalır.

Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

Bir uygulamayı konuşlandırdığınızda, diğer uygulamalarla paylaştığı IBM WebSphere MQ kaynaklarının ve bu kaynakların paylaşılmayacağı kaynaklara ilişkin bir seçiminiz olur. Paylaşılabilir kaynakların sayısı, sunucunun kendisi, kuyruk yöneticisi, kanal ve kuyruklar olmak üzere birçok kaynak tipi vardır. Uygulamaları daha az paylaşılan kaynakla yapılandırmayı seçebilir; ayrı kuyruklar, kanallar, kuyruk yöneticileri ya da hatta tek tek uygulamalar için sunucular yerleştirebilirsiniz. Bunu yapmazsanız, genel sistem yapılandırması daha büyük ve daha karmaşık hale gelir. IBM WebSphere MQ kümelerin kullanılması, daha fazla sunucunun, kuyruk yöneticilerinin, kuyrukların ve kanalların yönetilmesine ilişkin karmaşıklığı azaltır, ancak başka bir paylaşılan kaynak, küme iletim kuyruğu, SYSTEM . CLUSTER . TRANSMIT . QUEUE tanıtır.

Şekil 55 sayfa 279 is a slice out of a large IBM WebSphere MQ deployment that illustrates the significance of sharing SYSTEM . CLUSTER . TRANSMIT . QUEUE. In the diagram, the application, Client App, is connected to the queue manager QM2 in cluster CL1. A message from Client App is processed by the application, Server App. The message is retrieved by Server App from the cluster queue Q1 on the queue manager QM3 in CLUSTER2. İstemci ve sunucu uygulamaları aynı kümede olmadığından, ileti ağ geçidi kuyruk yöneticisi QM1 tarafından aktarılır.

Bir küme ağ geçidini yapılandırmanın olağan yolu, ağ geçidi kuyruk yöneticisinin tüm kümelerin bir üyesini yapmasıdır. Ağ geçidi kuyruk yöneticisinde, tüm kümelerdeki küme kuyrukları için kümelenmiş diğer ad kuyrukları tanımlanır. Kümelenmiş kuyruk diğer adları tüm kümelerde kullanılabilir. Küme kuyruğu diğer adlarına gönderilen iletiler, ağ geçidi kuyruk yöneticisi aracılığıyla doğru hedefleriyle yönlendirilir. Ağ geçidi kuyruk yöneticisi, kümelenmiş diğer ad kuyruklarına gönderilen iletileri QM1 üzerindeki ortak SYSTEM . CLUSTER . TRANSMIT . QUEUE ' e yerleştirir.

Merkez ve konuşan mimari, kümeler arasındaki tüm iletilerin ağ geçidi kuyruk yöneticisinden geçilmesini gerektirir. Sonuç olarak, tüm iletiler QM1, SYSTEM . CLUSTER . TRANSMIT . QUEUE üzerindeki tek küme iletim kuyruğundan geçer.

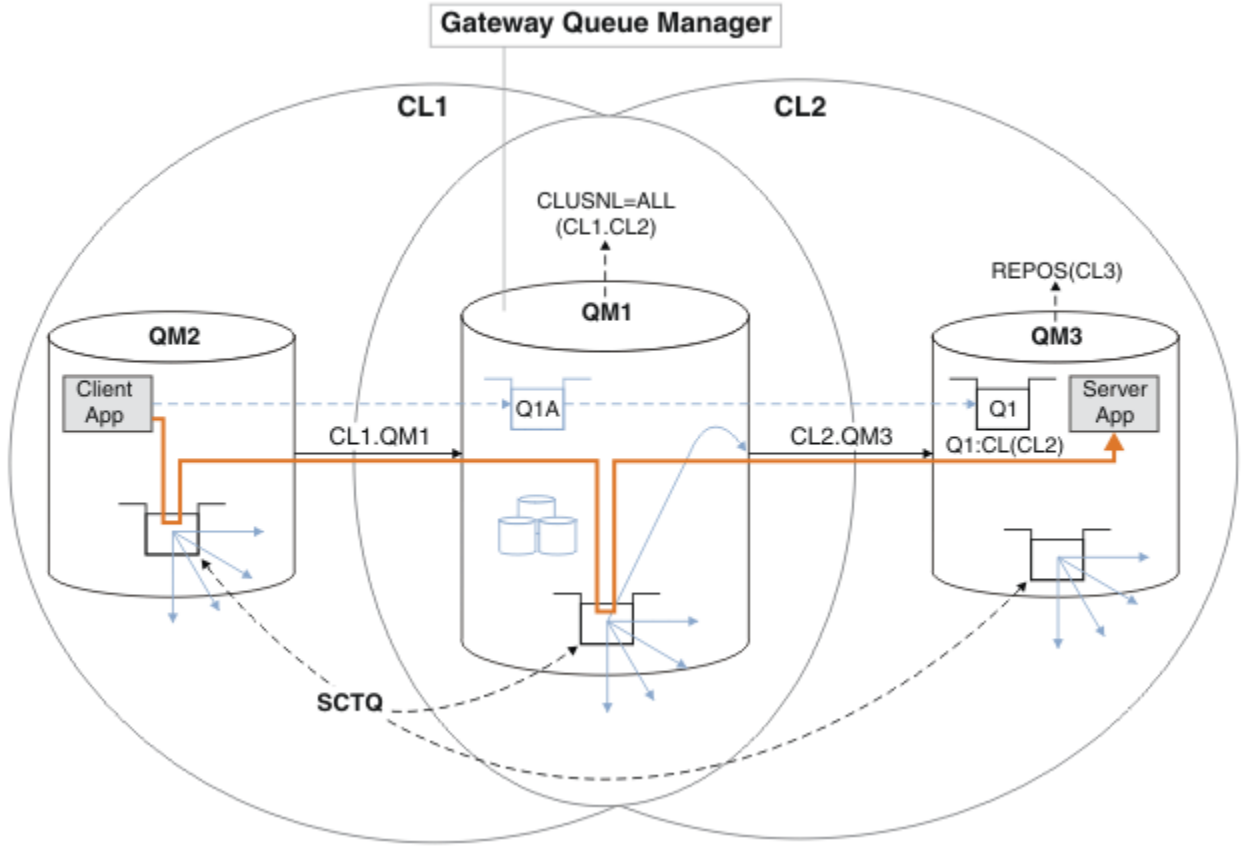
Başarım perspektifinden, tek bir kuyruk sorun değildir. Genel olarak yaygın bir iletim kuyruğu, performans darboğazını göstermez. Ağ geçidindeki ileti verimi büyük ölçüde, ona bağlanan kanalların performansı ile belirlenir. Üretilen iş genellikle kuyruk sayısından ya da kanalların kullandığı kuyruklardaki ileti sayısından etkilenmez.

Diğer bazı perspektiflerden, birden çok uygulama için tek bir iletim kuyruğunu kullanarak sakıncaları geri gelir:

- İletilerin akışını, ileti akışından başka bir hedefe ayıramazsınız. İletilerin farklı kuyruk yöneticilerindeki farklı kümelerde olması durumunda bile, iletilerin iletilmeden önce depolarını ayıramazsınız.

Bir küme hedefi kullanılamaz duruma gelirse, tek iletim kuyruğunda o hedef birikimine ilişkin iletiler ve sonunda iletiler bu bilgileri doldurur. İletim kuyruğu dolunca, iletilerin herhangi bir küme hedefi için iletim kuyruğuna konması durdurulur.

- İletilerin farklı küme hedeflerine aktarılmalarını izlemek kolay değildir. Tüm iletiler tek iletim kuyruğunda yer alıyor. İletim kuyruğunun derinliğinin görüntülenmesi, iletilerin tüm hedeflere aktarılıp aktarılmadığına ilişkin az bilgi verir.



Not: Şekil 55 sayfa 279 içindeki oklar ve aşağıdaki rakamlar farklı tiplerdir. Kesiksiz oklar ileti akışlarını gösterir. Kesiksiz oklardaki etiketler, ileti kanalı adlarıdır. The gray solid arrows are potential message flows from the SYSTEM.CLUSTER.TRANSMIT.QUEUE onto cluster-sender channels. Siyah çizgili çizgiler, etiketleri hedefleriyle ilişkilendirir. Gray dashed arrows are references; for example from an MQOPEN call by Client App to the cluster alias queue definition Q1A.

Şekil 55. Client-server application deployed to hub and spoke architecture using IBM WebSphere MQ clusters

Şekil 55 sayfa 279'ta, Server App 'un istemcileri kuyruğu Q1A' yi açar. Messages are put to SYSTEM.CLUSTER.TRANSMIT.QUEUE on QM2, transferred to SYSTEM.CLUSTER.TRANSMIT.QUEUE on QM1, and then transferred to Q1 on QM3, where they are received by the Server App application.

The message from Client App passes through system cluster transmission queues on QM2 and QM1. Şekil 55 sayfa 279 içinde, hedef ağ geçidi kuyruk yöneticisinde ileti akışını istemci uygulamasından yalıtılmak ve böylece iletilerin SYSTEM.CLUSTER.TRANSMIT.QUEUE' ta saklanmaması gerekir. Diğer kümelenmiş kuyruk yöneticilerinden herhangi birindeki akışları yalıtabilirsiniz. Akışları, diğer yöndeki istemciye geri giden diğer yönde de yalıtabilirsiniz. Çözümlerin açıklamalarını kısa tutmak için, açıklamalar istemci uygulamasından yalnızca tek bir akış olarak değerlendirilir.

Küme ileti trafiğini küme ağ geçidi kuyruk yöneticisinde yalıtmanın çözümleri

Sorunu çözenin bir yolu, kümeler arasında köprü yapmak için kuyruk yöneticisi diğer adlarını ya da uzak kuyruk tanımlamalarını kullanmandır. Ağ geçidi kuyruk yöneticisindeki her bir ileti akışını ayırmak için kümelenmiş bir uzak kuyruk tanımlaması, bir iletim kuyruğu ve bir kanal yaratın; bkz. "Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtılmak için uzak kuyruk tanımlaması eklenmesi" sayfa 196.

From Version 7.5 onwards, cluster queue managers are not limited to a single cluster transmission queue. İki seçeneğiniz vardır:

1. Ek küme iletim kuyruklarını el ile tanımlayın ve her bir iletim kuyruğundan hangi küme gönderici kanallarının iletileri aktarıp aktardığı tanımlayın; bkz. [“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199.](#)
2. Kuyruk yöneticisinin otomatik olarak ek küme iletim kuyrukları yaratmasına ve yönetmesine izin verin. Her bir küme gönderici kanalı için farklı bir küme iletim kuyruğu tanımlıyor; bkz. [“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219](#)

Bazı küme gönderici kanalları için el ile tanımlanmış küme iletim kuyruklarını, gerisini kuyruk yöneticisi yöneten kuyruk yöneticisiyle birleştirebilirsiniz. İletim kuyruklarının birleşimi, [“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199'](#) ta alınan yaklaşımdır. Bu çözümde, kümeler arasındaki çoğu ileti ortak `SYSTEM . CLUSTER . TRANSMIT . QUEUE'` ı kullanır. Bir uygulama kritik önem taşıyor ve tüm ileti akışları, el ile tanımlanmış bir küme iletim kuyruğu kullanılarak diğer akışlardan yalıtılıyor.

[“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199](#) içindeki yapılandırma sınırlanmıştır. İleti trafiğini, aynı kümedeki bir küme kuyruğuna, başka bir küme kuyruğunda aynı kuyruk yöneticisinde ayırmaz. Dağıtılmış kuyruğun bir parçası olan uzak kuyruk tanımlarını kullanarak, ileti trafiğini tek tek kuyruklara ayırabilirsiniz. Kümelerle, birden çok küme iletim kuyruğu kullanarak, farklı küme gönderen kanallarına giden ileti trafiğini birbirinden ayırabilirsiniz. Aynı kümedeki birden çok küme kuyruğu, aynı kuyruk yöneticisinde, bir küme gönderen kanalı paylaşır. Bu kuyruklara ilişkin iletiler, ağ geçidi kuyruk yöneticisinden iletilmeden önce, aynı iletim kuyruğunda saklanır. [“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202'](#) taki yapılandırma, başka bir küme eklenerek ve kuyruk yöneticisini ve küme kuyruğunu yeni kümenin bir üyesi haline getirerek taraf olarak adım adım atmaktadır. Yeni kuyruk yöneticisi, kümedeki tek kuyruk yöneticisi olabilir. Kümeye daha fazla kuyruk yöneticisi ekleyebilir ve aynı kümeyi kullanarak, bu kuyruk yöneticilerindeki küme kuyruklarını yalıtabilirsiniz.

İlgili kavramlar

[“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157](#)

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, `SYSTEM . CLUSTER . TRANSMIT . QUEUE'` a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanımlara karşı denetlenir.

[“Kümelenme: çakışan kümeler için dikkat edilmesi gereken özel noktalar” sayfa 273](#)

Bu konuda, IBM WebSphere MQ kümelerini planlamaya ve yönetmeye ilişkin yönergeler sağlanır. Bu bilgiler, müşterilerden gelen testlere ve geribildirime dayalı bir kılavuzdur.

[“Küme iletim kuyrukları ve küme-gönderici kanalları” sayfa 168](#)

Kümelenmiş kuyruk yöneticileri arasındaki iletiler, küme iletim kuyruklarında saklanır ve küme gönderici kanalları tarafından iletilir.

[“Çakışan kümeler” sayfa 176](#)

Çakışan kümeler, ek yönetim yetenekleri sağlar. Çakışan kümeleri denetlemek için gereken komut sayısını azaltmak için ad listelerini kullanın.

İlgili görevler

[Uzak küme kuyruklarına ileti koyma yetkisi](#)

[“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelenmiş bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

[“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden

bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafiğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanızı için bir yol sağlar.

“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

“Kümeler arasındaki ileti yollarının yapılandırılması” sayfa 253

Bir ağ geçidi kuyruk yöneticisini kullanarak kümeleri birbirine bağlayın. Ağ geçidi kuyruk yöneticisinde küme kuyruğu ya da küme kuyruğu yöneticisi diğer adlarını tanımlayarak kuyrukları ya da kuyruk yöneticilerini tüm kümeler tarafından görülebilir yapın.

“Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'deki altyapılar için geçerlidir.

İlgili başvurular

“Güvenlik” sayfa 422

Nesne Yetkisi Yöneticisi 'ne (OAM) ilişkin seçenekleri belirlemek için qm. ini dosyasındaki Security Stanza 'yı kullanın.

setmqaut

Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması

Küme iletim kuyrukları seçimleriyle yönlendirildiniz. Ortak bir varsayılan kuyruk, ayrı varsayılan kuyruklar ya da el ile tanımlanmış kuyruklar yapılandırabilirsiniz. Birden çok küme iletim kuyruğunun yapılandırılması, z/OS'deki altyapılar için geçerlidir.

Başlamadan önce

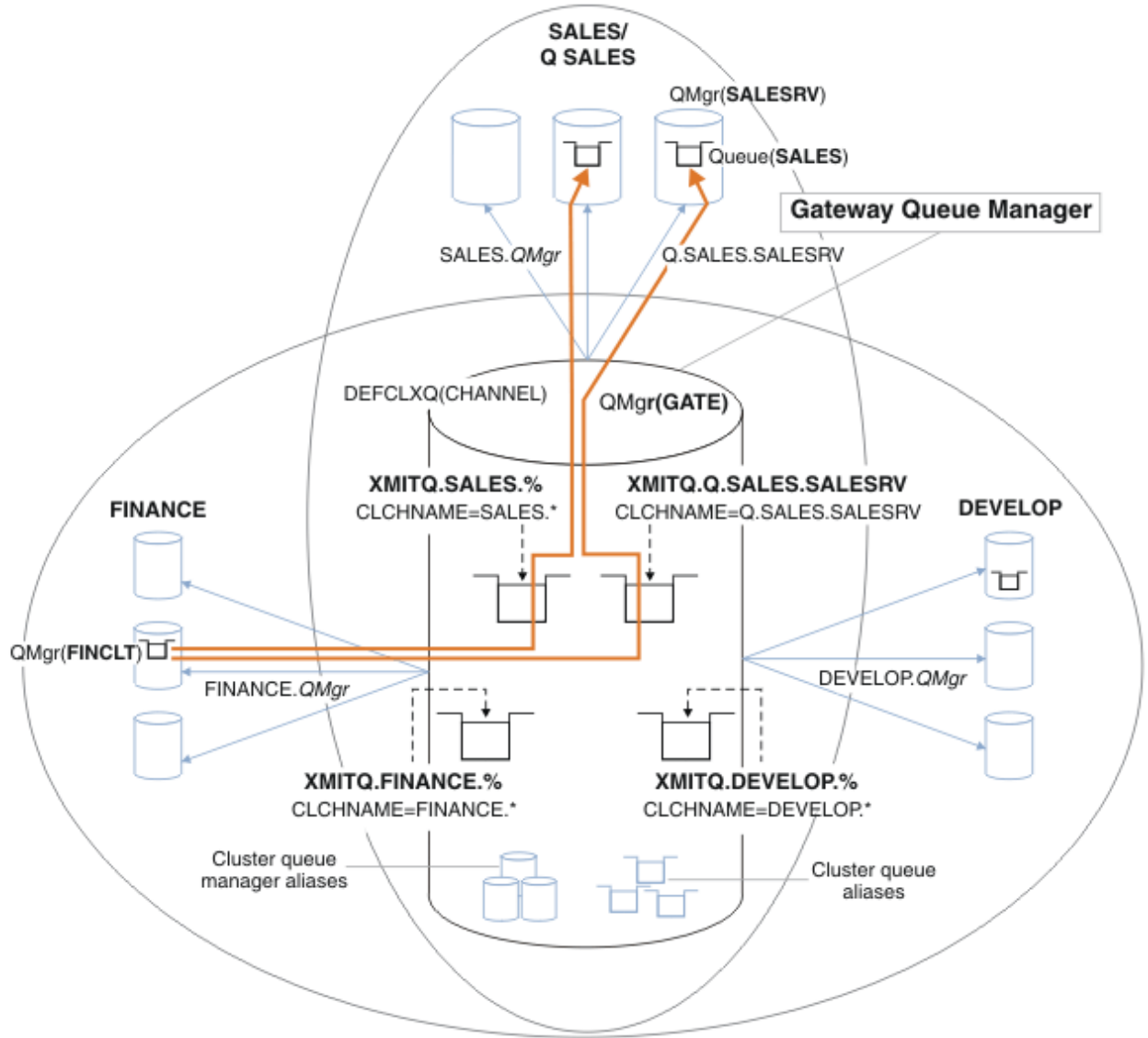
Review “Kullanılacak küme iletim kuyruğunun tipini nasıl seçecek?” sayfa 284.

Bu görev hakkında

Bir kuyruk yöneticisini bir küme iletim kuyruğu seçmek için nasıl yapılandıracağını planlarken yapmanız gereken bazı seçiminiz vardır.

1. Küme iletileri aktarımları için varsayılan küme iletim kuyruğu nedir?
 - a. Ortak bir küme iletim kuyruğu, SYSTEM . CLUSTER . TRANSMIT . QUEUE.
 - b. Ayrı küme iletim kuyrukları. Kuyruk yöneticisi, ayrı küme iletim kuyruklarını yönetir. Bu, bunları model kuyruğundan kalıcı dinamik kuyruklar olarak yaratır, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. Bu, kullandığı her bir küme gönderici kanalı için bir küme iletim kuyruğu oluşturur.
2. El ile yaratmaya karar verdiğiniz küme iletim kuyrukları için ek iki seçeneğiniz vardır:

- El ile yapılandırma kararı verdiğiniz her bir küme gönderici kanalı için ayrı bir iletim kuyruğu tanımlayın. Bu durumda, iletim kuyruğunun **CLCHNAME** kuyruk özneliğini, küme gönderen kanalının adına ayarlayın. Bu iletim kuyruğundan iletileri aktarmak için, küme gönderici kanalını seçin.
- Bir grup kümeli gönderici kanalının ileti trafiğini aynı küme iletim kuyruğunda birleştirin; bkz. Şekil 56 sayfa 282. Bu durumda, her ortak iletim kuyruğunun **CLCHNAME** kuyruk özneliğini soysal bir küme gönderen kanalı adına ayarlayın. Soysal bir kümeyle gönderici kanal adı, küme gönderen kanal adlarına süzgeç uygulanacak bir süzgeçtir. Örneğin, SALES . * , adları SALES . ile başlayan tüm küme gönderici kanallarını gruplarla gruplardır. Süzgeç dizisinin herhangi bir yerinde birden çok genel arama karakteri yerleştirebilirsiniz. Genel arama karakteri bir yıldız işaretidir ("*"). Sıfır ile herhangi bir sayıda karakter temsil eder.



Şekil 56. Farklı bölüm IBM WebSphere MQ kümeleri için belirli iletim kuyrukları örneği

Yordam

1. Kullanmak için varsayılan küme iletim kuyruğu tipini seçin.

- Tek bir küme iletim kuyruğu ya da her küme bağlantısı için ayrı kuyruklar seçin.

Varsayılan ayarı bırakın ya da **MQSC** komutunu çalıştırın:

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. Küme iletim kuyruğunu diğer akışlarla paylaşmaması gereken ileti akışlarını yalıtın.

- Bkz. [“Kümeleme: Birden çok küme iletim kuyruklarının yapılandırılması” sayfa 286](#). Örnek olarak, yalıtılması gereken SALES kuyruğu, SALESRV üzerinde SALES kümesinin bir üyesidir. To isolate the SALES queue, create a new cluster Q . SALES, make the SALESRV queue manager a member, and modify the SALES queue to belong to Q . SALES.
- SALES ' a ileti gönderen kuyruk yöneticileri de yeni kümenin üyesi olmalıdır. Örnekte olduğu gibi, kümelenmiş bir kuyruk diğer adı ve bir ağ geçidi kuyruk yöneticisi kullanıyorsanız, birçok durumda, ağ geçidi kuyruk yöneticisini yeni kümenin bir üyesi yapmak için gereken değişiklikleri sınırlayabilirsiniz.
- Ancak, ağ geçitinden hedefe akan akışlar, akışları kaynak kuyruk yöneticisinden ayrı olarak ayırmaz. ancak bazen ağ geçitten ayrı akışlar ayırmak ve ağ geçidine akmak için yeterli olduğu ortaya çıkıyor. Yeterli değilse, kaynak kuyruk yöneticisini yeni kümeye ekleyin. İletilerin ağ geçidi üzerinden seyahat etmesini istiyorsanız, küme diğer adını yeni kümeye taşıyın ve doğrudan hedef kuyruk yöneticisine değil, ağ geçidindeki küme diğer adlarına ileti göndermeye devam edin.

İleti akışlarını yalıtım için aşağıdaki adımları izleyin:

- a) Akışlar hedeflerini yapılandırın; böylece, her hedef kuyruk belirli bir kümedeki kuyrukta, o kuyruk yöneticisinde queue kuyrukta yer alan tek kuyruk olur.
- b) Sistematik bir adlandırma kuralından sonra yarattığınız yeni kümeler için küme gönderici ve küme alıcı kanallarını oluşturun.
 - Bkz. [“Kümeleme: çakışan kümeler için dikkat edilmesi gereken özel noktalar” sayfa 273](#).
- c) Hedef kuyruğa ileti gönderen her kuyruk yöneticisindeki her yalıtılmış hedef için bir küme iletim kuyruğu tanımlayın.
 - Küme iletim kuyrukları için bir adlandırma kuralı, küme kanalı adı özniteliğinin (CLCHNAME) değerini XMITQ . ile önekli olarak kullanmaktadır.

3. Yönetişim ya da izleme gereksinimlerini karşılamak için küme iletim kuyrukları yaratın.

- Tipik yönetim ve izleme gereksinimleri, küme başına iletim kuyruğunda ya da kuyruk yöneticisi başına iletim kuyruğu ile sonuçlanır. If you follow the naming convention for cluster channels, *ClusterName.QueueManagerName*, it is easy to create generic channel names that select a cluster of queue managers, or all the clusters a queue manager is a member of; see [“Kümeleme: Birden çok küme iletim kuyruklarının yapılandırılması” sayfa 286](#).
- Soysal kanal adları için, küme iletim kuyrukları için adlandırma kuralını, yıldız işaretini bir yüzde işaretiyle değiştirerek, soysal kanal adları için genişletir. Örneğin,

```
DEFINE QLOCAL(XMITQ.SALES.%) USAGE(XMITQ) CLCHNAME(SALES.*)
```

İlgili kavramlar

[“Küme iletim kuyrukları ve küme-gönderici kanalları” sayfa 168](#)

Kümelenmiş kuyruk yöneticileri arasındaki iletiler, küme iletim kuyruklarında saklanır ve küme gönderici kanalları tarafından iletilir.

[“Kümeleme: Birden çok küme iletim kuyruğu kullanılarak uygulama yalıtımı” sayfa 278](#)

Bir kümedeki kuyruk yöneticileri arasındaki ileti akışlarını yalıtabilirsiniz. Farklı küme gönderici kanalları tarafından, farklı küme iletim kuyruklarına iletilmekte olan iletileri yerleştirebilirsiniz. Yaklaşımı tek bir kümede ya da çakışan kümelerle kullanabilirsiniz. Bu konuda, kullanabileceğiniz bir yaklaşım seçiminde size yol göstermesi için bazı örnekler ve bazı en iyi uygulamalar sağlanır.

[“Erişim denetimi ve birden çok küme iletim kuyruğu” sayfa 157](#)

Bir uygulama uzak küme kuyruklarına ileti yerleştirdiğinde üç denetleme kipi arasında seçim yapın. Kipler, küme kuyruğunda uzaktan denetlenir, SYSTEM . CLUSTER . TRANSMIT . QUEUE' a yerel olarak denetlenir ya da küme kuyruğu ya da küme kuyruğu yöneticisi için yerel tanımlara karşı denetlenir.

[“Kümeleme: çakışan kümeler için dikkat edilmesi gereken özel noktalar” sayfa 273](#)

Bu konuda, IBM WebSphere MQ kümelerini planlamaya ve yönetmeye ilişkin yönergeler sağlanır. Bu bilgiler, müşterilerden gelen testlere ve geribildirimlere dayalı bir kılavuzdur.

[“Çakışan kümeler” sayfa 176](#)

Çakışan kümeler, ek yönetim yetenekleri sağlar. Çakışan kümeleri denetlemek için gereken komut sayısını azaltmak için ad listelerini kullanın.

İlgili görevler

[“Ağ geçidi kuyruk yöneticisinden gönderilen iletileri yalıtma için uzak kuyruk tanımlaması eklenmesi” sayfa 196](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. Çözüm, kümelenmiş bir kuyruk uzak tanımlamasını ve ayrı bir gönderen kanalı ve iletim kuyruğunu kullanır.

[“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri yalıtma için bir küme iletim kuyruğu ekleme” sayfa 199](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm, ileti trafiğini bir kümedeki tek bir kuyruk yöneticisinde ayırmak için ek bir küme iletim kuyruğu kullanır.

[“Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202](#)

Ağ geçidi kuyruk yöneticisini kullanan çakışan kümelerin yapılandırmasını değiştirin. Değişiklik iletileri, diğer küme iletileriyle aynı iletim kuyruğunu ya da kanalları kullanmadan, ağ geçidi kuyruk yöneticisinden bir uygulamaya aktarıldıktan sonra. ' den bir uygulamaya aktarıldıktan sonra, çözüm belirli bir küme kuyruğunda iletileri yalıtma için ek bir küme kullanır.

[“İleti trafiğini yalıtma için varsayılan olarak küme iletim kuyrukları için varsayılan olarak değiştirilmesi” sayfa 219](#)

Bir kuyruk yöneticisinin, bir iletim kuyruğunda kümelenmiş bir kuyruğa ya da konuya ilişkin iletileri sakladığını değiştirmenin varsayılan yolunu değiştirebilirsiniz. Varsayılan değeri değiştirmek, ağ geçidi kuyruk yöneticisiyle ilgili küme iletilerini yalıtmanızı için bir yol sağlar.

[“Ağ geçidi kuyruk yöneticisiyle çakışan iki küme oluşturma” sayfa 213](#)

Bir ağ geçidi kuyruk yöneticisiyle çakışan kümeler oluşturmak için görevdeki yönergeleri izleyin. Kümeleri, bir uygulamadaki iletileri bir kümedeki diğer uygulamalara iletilerden yalıtma örnekleri için başlangıç noktası olarak kullanın.

[“Kümeler arasındaki ileti yollarının yapılandırılması” sayfa 253](#)

Bir ağ geçidi kuyruk yöneticisini kullanarak kümeleri birbirine bağlayın. Ağ geçidi kuyruk yöneticisinde küme kuyruğu ya da küme kuyruğu yöneticisi diğer adlarını tanımlayarak kuyrukları ya da kuyruk yöneticilerini tüm kümeler tarafından görülebilir yapın.

Kullanılacak küme iletim kuyruğunun tipini nasıl seçecek?

Farklı küme iletim kuyruğu yapılandırma seçenekleri arasında seçim yapabilirsiniz.

Version 7.5 ' den başlayarak, bir küme gönderici kanalıyla hangi küme iletim kuyruğunu ilişkilendirdiğini seçebilirsiniz.

1. Tek varsayılan küme iletim kuyruğu olan SYSTEM . CLUSTER . TRANSMIT . QUEUE ile ilişkili tüm küme gönderici kanallarına sahip olabilirsiniz. This option is the default, and is the only choice for queue managers that run version Version 7.1, or earlier.
2. Tüm küme gönderici kanallarını otomatik olarak ayrı bir küme iletim kuyruğuyla ilişkilendirebilirsiniz. Kuyruklar, kuyruk yöneticisi tarafından SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE model kuyruğundan yaratılır ve SYSTEM . CLUSTER . TRANSMIT . ChannelName adlı kuyruğa alır. Kuyruk yöneticisi özniteliği **DEFCLXQ** KANAL olarak ayarlandıysa, kanallar benzersiz olarak adlandırılan küme iletim kuyruğunu kullanır.



Uyarı: Ürünün önceki bir sürümünden yükseltilecek bir kuyruk yöneticisiyle özel olarak ayrılmış SYSTEM . CLUSTER . TRANSMIT . QUEUE kullanıyorsanız, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE ' in SHARE/NOSHARE seçeneğinin **SHARE** olarak ayarlandığından emin olun.

3. Tek bir küme iletim kuyruğu tarafından hizmet vermek için küme gönderici kanalları belirleyebilirsiniz. Bir iletim kuyruğu yaratarak ve **CLCHNAME** özniteliğini, küme gönderen kanalının adına ayarlayarak bu seçeneği belirleyin.
4. Tek bir küme iletim kuyruğu tarafından hizmet vermek için küme gönderici kanallarının gruplarını seçebilirsiniz. Bir iletim kuyruğu yaratarak ve **CLCHNAME** özniteliğini *ClusterName.** gibi soysal bir kanal adına ayarlayarak bu seçeneği belirleyin. If you name cluster channels by following the naming conventions in “Kümeleme: çakışan kümeler için dikkat edilmesi gereken özel noktalar” sayfa 273, this name selects all cluster channels connected to queue managers in the cluster *ClusterName*.

Bazı küme gönderenler kanallarına ilişkin varsayılan küme iletim kuyruğu seçeneklerinden herhangi birini, herhangi bir sayıda belirli ve soysal küme iletim kuyruğu yapılandırmasıyla birleştirebilirsiniz.

En iyi uygulamalar

Çoğu durumda, var olan IBM WebSphere MQ kuruluşları için varsayılan yapılandırma en iyi seçenektir. Küme kuyruk yöneticisi, küme iletilerini tek bir küme iletim kuyruğunda saklar, SYSTEM . CLUSTER . TRANSMIT . QUEUE. Farklı kuyruk yöneticilerine ve ayrı iletim kuyruklarında farklı kümelere ilişkin iletileri saklamak ya da kendi iletim kuyruklarınızı tanımlamak için varsayılan değeri değiştirme seçeneğiniz vardır.

Çoğu durumda, yeni IBM WebSphere MQ kurulumları için varsayılan yapılandırma da en iyi seçenektir. Varsayılan yapılandırmadan, her bir küme gönderici kanalı için tek bir iletim kuyruğuna sahip olmanın alternatif varsayılan yapılandırmasından geçiş işlemi otomatik olarak işlenmektedir. Geri geçilmesi de otomatidir. Bir ya da diğerinin seçimi kritik değildir, bunu tersine çevirebilirsiniz.

Farklı bir yapılandırmanın seçilmesinin nedeni, yönetim ve yönetimle ilgili işlevlerden ya da performansla göre daha çok iş yapmasıdır. Birkaç kural dışı durumla, birden çok küme iletim kuyruklarının yapılandırılması, kuyruk yöneticisinin davranışına yararlanmamaktadır. Daha fazla kuyrukta sonuç alır ve önceden ayarladığınız izleme ve yönetim yordamlarıyla tek iletim kuyruğuna başvuruda bulunmanızı gerektirir. Bu nedenle, farklı bir seçim için güçlü bir yönetim veya yönetim nedeniniz yoksa, varsayılan yapılandırmayla birlikte kalan en iyi seçenek budur.

The exceptions are both concerned with what happens if the number of messages stored on SYSTEM . CLUSTER . TRANSMIT . QUEUE increases. Başka bir hedefe ilişkin iletilerden bir hedefe ilişkin iletileri ayırmak için her adımı atsanız, bir hedefle kanal ve teslim sorunları başka bir hedefe teslim edilmeyi etkilemez. However, the number of messages stored on SYSTEM . CLUSTER . TRANSMIT . QUEUE can increase due to not delivering messages fast enough to one destination. Bir hedef için SYSTEM . CLUSTER . TRANSMIT . QUEUE üzerindeki iletilerin sayısı, iletilerin diğer hedeflere teslimi olumsuz etkileyebilir.

Tek bir iletim kuyruğunu doldurmaktan kaynaklanan sorunları önlemek için, yapılandırmanızı kullanarak yeterli kapasite oluşturmayı hedefleyebilirsiniz. Daha sonra, bir hedef başarısız olursa ve bir ileti birikimi oluşturulmaya başlarsa, sorunu çözecek zamanınız olur.

İletiler, küme ağ geçidi gibi bir göbek kuyruk yöneticisiyle yönlendirilmişse, ortak bir iletim kuyruğunu paylaşırlar. SYSTEM . CLUSTER . TRANSMIT . QUEUE. Ağ geçidi kuyruk yöneticisiyle ilgili SYSTEM . CLUSTER . TRANSMIT . QUEUE üzerinde saklanan ileti sayısı derinlik üst sınırına ulaşırsa, kuyruk yöneticisi iletim kuyruğu için derinliği azalana kadar yeni iletileri reddetmeye başlar. Tıkanıklık, ağ geçidinden yönlendirilen tüm hedeflere ilişkin iletileri etkiler. İletiler, ağ geçidine ileti gönderen diğer kuyruk yöneticilerinin iletim kuyruklarını yedeklemektedir. Bu sorun, kuyruk yöneticisi hata günlüklerine, düşen ileti verisine ve iletiye bir iletinin varış noktasına geldiği süre arasında geçen zaman arasında yazılan iletilerde kendisini bildirmektedir.

Sıkışıklığın tek bir iletim kuyruğunda etkisi, tam olmadan önce bile belirginleşebilir. Bazı büyük ve kalıcı olmayan iletiler ve bazı küçük iletiler ile karışık ileti trafiği varsa, iletim kuyruğu doldururken küçük iletiler teslim etme süresi artar. Gecikme süresi, normalde diske yazılmayan, kalıcı olmayan büyük iletilerin diske yazılması nedeniyle ortaya çıkar. Zaman kritik ileti akışlarına sahipseniz, bir küme iletim kuyruğunu diğer karışık iletiler akışlarıyla paylaşırsanız, bu iletiyi diğer ileti akışlarından yalıtılmak için özel bir ileti yolu yapılandırmaya değer olabilir; bkz. “Ağ geçidi kuyruk yöneticisinden gönderilen küme ileti trafiğini yalıtılmak için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202

Ayrı küme iletim kuyruklarının yapılandırılmasına ilişkin diğer nedenler, yönetim gereksinimlerini karşılamakta ya da farklı küme hedeflerine gönderilen izleme iletilerini basitleştirmektedir. Örneğin, bir hedef için iletilerin başka bir hedef için ileti içeren bir iletim kuyruğunu hiçbir zaman paylaşmadığını göstermek zorunda kalabilirsiniz.

Varsayılan küme iletim kuyruğunu denetleyen **DEFCLXQ** kuyruk yöneticisi özniteliğini, her bir küme gönderici kanalı için farklı küme iletim kuyrukları yaratmak üzere değiştirin. Birden çok hedef, bir küme gönderici kanalını paylaşabilir, bu nedenle kümelerin bu amacı tam olarak karşılaması için planlamanız gerekir. “Ağ geçidi kuyruk yöneticisinden gönderilen küme iletileri trafikliğini yalıtma için bir küme ve bir küme iletim kuyruğu ekleme” sayfa 202 yöntemini tüm küme kuyruklarınıza sistematik olarak uygulayın. Hedeflediğiniz sonuç, bir küme gönderici kanalını başka bir küme hedefiyle paylaşmak için herhangi bir küme hedefi olmayışıdır. Sonuç olarak, bir küme hedefine ilişkin hiçbir ileti, başka bir hedef için bir iletiyle küme iletim kuyruğunu paylaşır.

Belirli bir ileti akışı için ayrı bir küme iletim kuyruğu yaratılması, bu hedefe yönelik ileti akışının izlenmesini kolaylaştırır. Yeni bir küme iletim kuyruğu kullanmak için, kuyruğu tanımlayın, bir kümeyle ilişkilendiren bir kanalla ilişkilendirin ve kanalı durdurup başlatın. Değişiklik kalıcı olmak zorunda değil. Bir süre için ileti akışını yalıtıp iletim kuyruğunu izleyebilir ve varsayılan iletim kuyruğunu yeniden kullanmaya geri dönebilirsiniz.

İlgili görevler

Kümeleme: Birden çok küme iletim kuyruklarının yapılandırılması

Bu görevde, birden çok küme iletim kuyruklarını üç üst üste çakışan kümeyle planlamak için adımları uygulayın. Gereksinimler, diğer tüm ileti akışlarından bir küme kuyruğuna ve farklı küme iletim kuyruklarında farklı kümelere ilişkin iletileri saklamak için, iletileri ayrı ayrı bir küme kuyruğuna yollamak içindir.

Kümeleme: Küme iletim kuyruklarının değiştirilmesi

Var olan bir üretim kuyruğu yöneticisinin küme iletim kuyruklarına ilişkin değişikliklerin yürürlüğe nasıl getirileceğini planlayın.

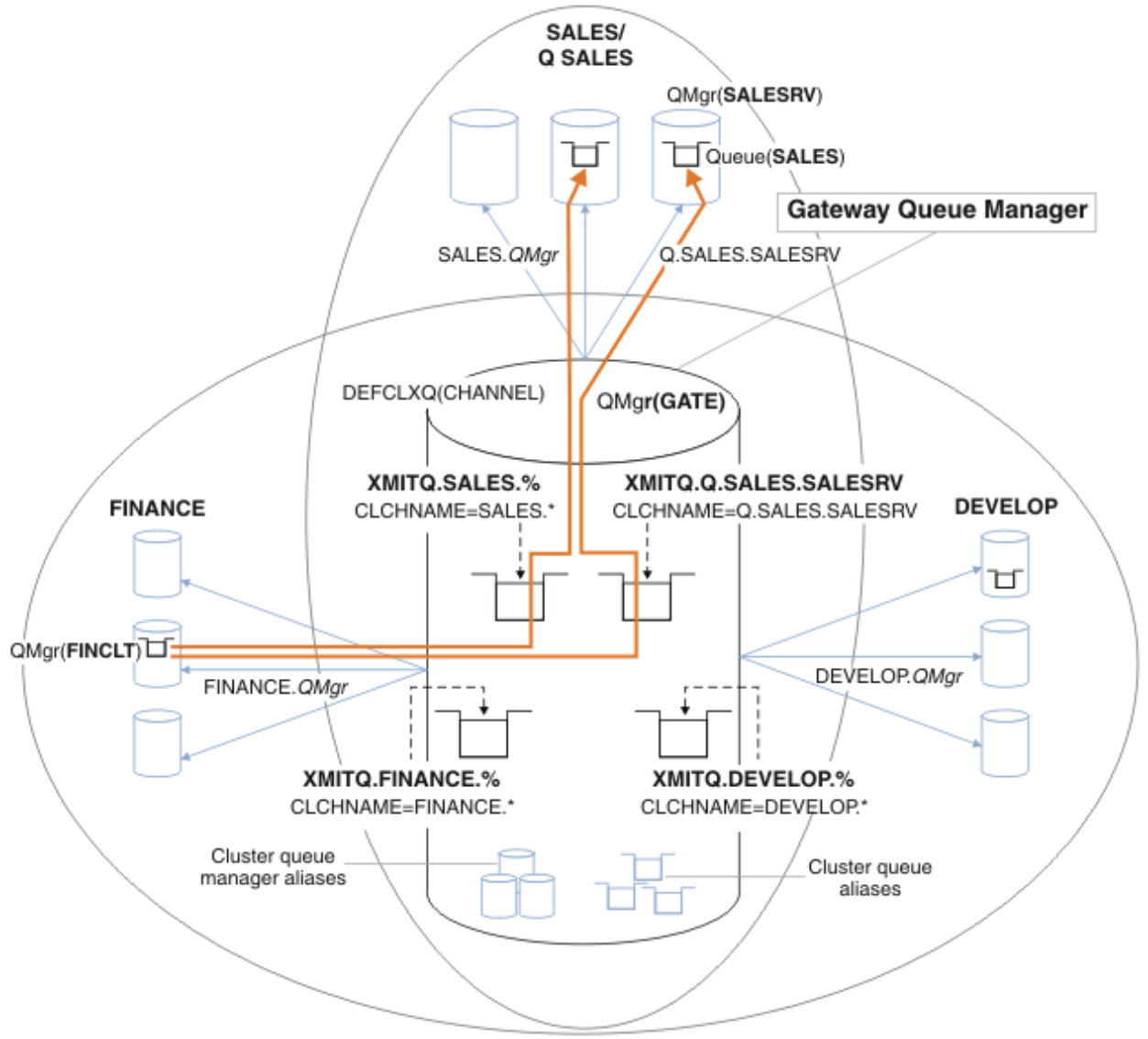
Kümeleme: Birden çok küme iletim kuyruklarının yapılandırılması

Bu görevde, birden çok küme iletim kuyruklarını üç üst üste çakışan kümeyle planlamak için adımları uygulayın. Gereksinimler, diğer tüm ileti akışlarından bir küme kuyruğuna ve farklı küme iletim kuyruklarında farklı kümelere ilişkin iletileri saklamak için, iletileri ayrı ayrı bir küme kuyruğuna yollamak içindir.

Bu görev hakkında

The steps in this task show how to apply the procedure in “Kümeleme: Küme iletim kuyruklarının nasıl yapılandırılacağı planlanması” sayfa 281 and arrive at the configuration shown in Şekil 57 sayfa 287. Bu, ayrı küme iletim kuyruklarıyla yapılandırılmış bir ağ geçidi kuyruk yöneticisi olan çakışan üç kümeden bir örnektir. Kümeleri tanımlamak için kullanılan MQSC komutları “Örnek kümeler oluşturma” sayfa 289’ünde açıklanmıştır.

Örneğin, iki gereksinim vardır. Bunlardan biri, ileti akışını, ağ geçidi kuyruk yöneticisinden satışları günlüğe kaydeden satış uygulamasına ayırmak olur. İkincisi, farklı departman alanlarına zaman içinde ne kadar mesaj gönderileceğini sorgulamak. SALES, FINANCE ve DEVELOP kümeleri önceden tanımlanır. Küme iletileri şu anda SYSTEM . CLUSTER . TRANSMIT . QUEUE' den iletilir.



Şekil 57. Farklı bölüm IBM WebSphere MQ kümeleri için belirli iletim kuyrukları örneği

Kümeleri değiştirme adımları aşağıdaki gibidir; tanımlamalar için Yeni bir kümede satış kuyruğunu yalıtma ve ağ geçidi kümesi iletim kuyruklarını ayırmak için yapılan değişiklikler başlıklı konuya bakın.

Yordam

1. İlk yapılandırma adımı "Kullanmak için varsayılan küme iletim kuyruğu tipini seçin" dir.

Bu karar, GATE kuyruk yöneticisine aşağıdaki **MQSC** komutunu çalıştırarak ayrı varsayılan küme iletim kuyrukları yaratılabilir.

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Bu varsayılan değer, küme iletim kuyruklarını el ile tanımlamakta olduğu için, bu varsayılan değeri seçmenin güçlü bir nedeni yoktur. Seçim zayıf bir tanımlama değerine sahip. El ile tanımlama yanlış yapılırsa ve bir ileti varsayılan küme iletim kuyruğunu aşağı doğru akıyorsa, bu ileti kalıcı dinamik küme iletim kuyruğu oluşturulmasında ortaya çıkar.

2. İkinci yapılandırma adımı "Küme iletim kuyruğunu diğer akışlarla paylaşmaması gereken ileti akışlarını yalıtın" dir.

Bu durumda, SALESRV üzerindeki SALES kuyruğundan ileti alan satış uygulaması yalıtma gerektirir. Yalnızca ağ geçidi kuyruk yöneticisinden gelen iletilerin yalıtılmış olması gerekir. Üç alt adım şunlardır:

- a) "Akışlar hedeflerini yapılandırın; böylece, her hedef kuyruk belirli bir kümedeki kuyrukta, o kuyruk yöneticisindequeuekuyrukta yer alan tek kuyruk olur."

The example requires adding the queue manager SALESRV to a new cluster within the sales department. Yalıtma gerektiren birkaç kuyruğunuz varsa, SALES kuyruğu için belirli bir küme yaratmaya karar verebilirsiniz. Küme adı için olası bir adlandırma kuralı, Q . *QueueName* gibi bu tür kümelerin adlandırılması (örneğin, Q . SALES). Yalıtılacak çok sayıda kuyruğunuz olması durumunda daha pratik olabilecek alternatif bir yaklaşım, gerektiğinde ve gerektiğinde yalıtılmış kuyruklar kümelerinin oluşturulabilirdir. Küme adları QUEUES . *not*abilir.

Örnekte, yeni küme Q . SALESolarak adlandırılır. Yeni kümeyi eklemek için, Yeni bir kümede satış kuyruklarını yalıtma ve ağ geçidi kümesi iletim kuyruklarını ayırmak için yapılan değişikliklerindeki tanımları görebilirsiniz. Tanımlama değişikliklerinin özeti aşağıdaki gibidir:

- Havuz kuyruğu yöneticilerindeki kümelerin ad listesine Q . SALES ekleyin. Ad listesi, kuyruk yöneticisi **REPOSNL** parametresine başvurur.
- Ağ geçidi kuyruk yöneticisinde bulunan kümelerin ad listesine Q . SALES ekleyin. Ad listesi, ağ geçidi kuyruk yöneticisinde tüm küme kuyruğu diğer adı ve küme kuyruğu yöneticisi diğer adı tanımlamalarında gönderme edilir.
- Create a namelist on the queue manager SALESRV , for both the clusters it is a member of, and change the cluster membership of the SALES queue:

```
DEFINE NAMLIST(CLUSTERS) NAMES(SALES, Q.SALES) REPLACE  
ALTER QLOCAL(SALES) CLUSTER(' ') CLUSNL(SALESRV.CLUSTERS)
```

SALES kuyruğu, geçiş için her iki kümenin de bir üyesidir. Once the new configuration is running, you remove the SALES queue from the SALES cluster; see [Şekil 58 sayfa 292](#).

- b) "Sistemik bir adlandırma kuralından sonra yarattığınız yeni kümeler için küme gönderici ve küme alıcı kanallarını oluşturun".
- Havuz günlük nesnesi kanalını Q . SALES . *RepositoryQMgr* havuz kuyruğu yöneticilerinin her birine ekleyin
 - Diğer havuz yöneticisine bağlanmak için, küme gönderici kanalını Q . SALES . *OtherRepositoryQMgr* havuz kuyruğu yöneticilerine ekleyin. Bu kanalları başlatın.
 - Q . SALES . SALESRV ve Q . SALES . GATE küme alıcı kanallarını çalışmakta olan havuz kuyruğu yöneticilerinden birine ekleyin.
 - Add the cluster-sender channels Q . SALES . SALESRV , and Q . SALES . GATE to the SALESRV and GATE queue managers. Küme gönderen kanalını, küme alıcı kanallarını oluşturduğunuz havuz kuyruk yöneticisine bağlayın.
- c) "Hedef kuyruğa ileti gönderen her kuyruk yöneticisindeki her yalıtılmış hedef için bir küme iletim kuyruğu tanımlayın".

On the gateway queue manager define the cluster transmission queue XMITQ . Q . SALES . SALESRV for the Q . SALES . SALESRV cluster-sender channel:

```
DEFINE QLOCAL(XMITQ.Q.SALES.SALESRV) USAGE(XMITQ) CLCHNAME(Q.SALES.SALESRV) REPLACE
```

3. Üçüncü yapılandırma adımı, "Yönetişim ya da izleme gereksinimlerini karşılamak için küme iletim kuyrukları yaratın" ile ilgili bir adımdır.

Ağ geçidi kuyruk yöneticisinde, küme iletim kuyruklarını tanımlayın:

```
DEFINE QLOCAL(XMITQ.SALES) USAGE(XMITQ) CLCHNAME(SALES.*) REPLACE  
DEFINE QLOCAL(XMITQ.DEVELOP) USAGE(XMITQ) CLCHNAME(DEVELOP.*) REPLACE  
DEFINE QLOCAL(XMITQ.FINANCE) USAGE(XMITQ) FINANCE(SALES.*) REPLACE
```

Sonraki adım

Ağ geçidi kuyruk yöneticisinde yeni yapılandırmaya geçin.

Anahtar, yeni kanallar başlatılarak ve artık farklı iletim kuyruklarıyla ilişkili kanalların yeniden başlatılarak tetiklenir. Diğer bir seçenek olarak, ağ geçidi kuyruk yöneticisini durdurabilir ve başlatabilirsiniz.

1. Ağ geçidi kuyruk yöneticisinde aşağıdaki kanalları durdurun:

```
SALES.Qmgr
DEVELOP.Qmgr
FINANCE.Qmgr
```

2. Ağ geçidi kuyruk yöneticisinde aşağıdaki kanalları başlatın:

```
SALES.Qmgr
DEVELOP.Qmgr
FINANCE.Qmgr
Q.SALES.SAVESRV
```

Anahtar tamamlandığında, SALES kuyruğunu SALES kümesinden çıkarın; bkz. [Şekil 58 sayfa 292](#).

İlgili kavramlar

Kullanılacak küme iletim kuyruğunun tipini nasıl seçecek?

Farklı küme iletim kuyruğu yapılandırma seçenekleri arasında seçim yapabilirsiniz.

İlgili görevler

Kümeleme: Küme iletim kuyruklarının değiştirilmesi

Var olan bir üretim kuyruğu yöneticisinin küme iletim kuyruklarına ilişkin değişikliklerin yürürlüğe nasıl getirileceğini planlayın.

Örnek kümeler oluşturma

The definitions and instructions to create the example cluster, and modify it to isolate the SALES queue and separate messages on the gateway queue manager.

Bu görev hakkında

FINANCE, SALES ve Q.SALES kümelerini oluşturmak için tam **MQSC** komutları, Temel kümelere ilişkin tanımlar, Yeni bir kümede satış kuyruğunu yalıtım ve ağ geçidi kümesi iletim kuyruklarını ayırmak için yapılan değişikliklerle Satış kümesinden kuyruk yöneticisi SALESRV kuyruk yöneticisi üzerindeki satış kuyruğunu kaldırmak için sağlanır. Tanımları daha kısa tutmak için, DEVELOP kümesi tanımlamalardan çıkarılır.

Yordam

1. SALES ve FINANCE kümelerini ve ağ geçidi kuyruk yöneticisini yaratın.

a) Kuyruk yöneticilerini yaratın.

Run the command: `crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QmgrName` for each of the queue manager names in [Çizelge 27 sayfa 289](#).

Çizelge 27. Kuyruk yöneticisi adları ve kapı numaraları		
Tanım	Kuyruk yöneticisi adı	Kapı numarası
Finans havuzu	FINR1	1414
Finans havuzu	FINR2	1415
Finans istemcisi	FINCLT	1418
Satış havuzu	SALER1	1416
Satış havuzu	SALER2	1417
Satış sunucusu	SALESRV	1419
Ağ geçidi	GATE	1420

b) Tüm kuyruk yöneticilerini başlat

Run the command: `strmqm QmgrName` for each of the queue manager names in [Çizelge 27 sayfa 289](#).

c) Kuyruk yöneticilerinin her biri için tanımlamaları yarat

Şu komutu çalıştırın: `runmqsc QmgrName < filename` , dosyalar [Temel kümelere ilişkin tanımları](#) içinde listelenmekte ve dosya adı kuyruk yöneticisi adıyla eşleşir.

Temel kümelere ilişkin tanımlar

finr1.txt

```
DEFINE LISTENER(1414) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1414) REPLACE
START LISTENER(1414)
ALTER QMGR REPOS(FINANCE)
DEFINE CHANNEL(FINANCE.FINR2) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)')
CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1414)')
CLUSTER(FINANCE) REPLACE
```

finr2.txt

```
DEFINE LISTENER(1415) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1415) REPLACE
START LISTENER(1415)
ALTER QMGR REPOS(FINANCE)
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSSDR) CONNAME('localhost(1414)')
CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.FINR2) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)')
CLUSTER(FINANCE) REPLACE
```

finclt.txt

```
DEFINE LISTENER(1418) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1418) REPLACE
START LISTENER(1418)
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSSDR) CONNAME('localhost(1414)')
CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.FINCLT) CHLTYPE(CLUSRCVR) CONNAME('localhost(1418)')
CLUSTER(FINANCE) REPLACE
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

saler1.txt

```
DEFINE LISTENER(1416) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1416) REPLACE
START LISTENER(1416)
ALTER QMGR REPOS(SALES)
DEFINE CHANNEL(SALES.SALER2) CHLTYPE(CLUSSDR) CONNAME('localhost(1417)')
CLUSTER(SALES) REPLACE
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1416)')
CLUSTER(SALES) REPLACE
```

saler2.txt

```
DEFINE LISTENER(1417) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1417) REPLACE
START LISTENER(1417)
ALTER QMGR REPOS(SALES)
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)')
CLUSTER(SALES) REPLACE
DEFINE CHANNEL(SALES.SALER2) CHLTYPE(CLUSRCVR) CONNAME('localhost(1417)')
CLUSTER(SALES) REPLACE
```

salesrv.txt

```
DEFINE LISTENER(1419) TRPTYPE(TCP) IPADDR(localhost) CONTROL(QMGR) PORT(1419) REPLACE
START LISTENER(1419)
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)')
CLUSTER(SALES) REPLACE
DEFINE CHANNEL(SALES.SALESRV) CHLTYPE(CLUSRCVR) CONNAME('localhost(1419)')
CLUSTER(SALES) REPLACE
DEFINE QLOCAL(SALES) CLUSTER(SALES) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE)
PROCESS(ECHO) REPLACE
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

gate.txt

```
DEFINE LISTENER(1420) TRPTYPE(TCP) IPADDR(LOCALHOST) CONTROL(QMGR) PORT(1420) REPLACE
START LISTENER(1420)
DEFINE NAMELIST(ALL) NAMES(SALES, FINANCE)
DEFINE CHANNEL(FINANCE.FINR1) CHLTYPE(CLUSSDR) CONNAME('LOCALHOST(1414)')
CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(FINANCE.GATE) CHLTYPE(CLUSRCVR) CONNAME('LOCALHOST(1420)')
CLUSTER(FINANCE) REPLACE
DEFINE CHANNEL(SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('LOCALHOST(1416)')
CLUSTER(SALES) REPLACE
DEFINE CHANNEL(SALES.GATE) CHLTYPE(CLUSRCVR) CONNAME('LOCALHOST(1420)')
CLUSTER(SALES) REPLACE
DEFINE QALIAS(A.SALES) CLUSNL(ALL) TARGET(SALES) TARGTYPE(Queue) DEFBIND(NOTFIXED)
REPLACE
DEFINE QREMOTE(FINCLT) RNAME(' ') RQMNAME(FINCLT) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(SALESRV) RNAME(' ') RQMNAME(SALESRV) CLUSNL(ALL) REPLACE
```

2. Örnek istek programını çalıştırarak yapılandırmayı test edin.

a) Start the trigger monitor program on the SALESRV queue manager

Windowsüzerinde bir komut penceresi açın ve `runmqtrm -m SALESRV`komutunu çalıştırın.

b) Örnek istek programını çalıştırın ve bir istek gönderin.

Windowsüzerinde bir komut penceresi açın ve `amqsreq A.SALES FINCLT`komutunu çalıştırın.

İstek iletişi yeniden yankılanır ve 15 saniye sonra örnek programın sona ermesi gerekir.

3. Create the definitions to isolate the SALES queue in the Q.SALES cluster and separate cluster messages for the SALES and FINANCE cluster on the gateway queue manager.

Şu komutu çalıştırın: `runmqsc QmgrName <filename>`, dosyalar aşağıdaki listede listelenmekte ve dosya adı kuyruk yöneticisi adıyla neredeyse eşleşiyor.

Satış kuyruğunu yeni bir kümede yalıtmak ve ağ geçidi kümesi iletim kuyruklarını ayırmak için yapılan değişiklikler
chgsaler1.txt

```
DEFINE NAMELIST(CLUSTERS) NAMES(SALES, Q.SALES)
ALTER QMGR REPOS(' ') REPOSNL(CLUSTERS)
DEFINE CHANNEL(Q.SALES.SALER2) CHLTYPE(CLUSSDR) CONNAME('localhost(1417)')
CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1416)')
CLUSTER(Q.SALES) REPLACE
```

chgsaler2.txt

```
DEFINE NAMELIST(CLUSTERS) NAMES(SALES, Q.SALES)
ALTER QMGR REPOS(' ') REPOSNL(CLUSTERS)
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)')
CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.SALER2) CHLTYPE(CLUSRCVR) CONNAME('localhost(1417)')
CLUSTER(Q.SALES) REPLACE
```

chgsalesrv.txt

```
DEFINE NAMELIST (CLUSTERS) NAMES(SALES, Q.SALES)
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)')
CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.SAVESRV) CHLTYPE(CLUSRCVR) CONNAME('localhost(1419)')
CLUSTER(Q.SALES) REPLACE
ALTER QLOCAL (SALES) CLUSTER(' ') CLUSNL(CLUSTERS)
```

chgate.txt

```
ALTER NAMELIST(ALL) NAMES(SALES, FINANCE, Q.SALES)
ALTER QMGR DEFCLXQ(CHANNEL)
DEFINE CHANNEL(Q.SALES.SALER1) CHLTYPE(CLUSSDR) CONNAME('localhost(1416)')
CLUSTER(Q.SALES) REPLACE
DEFINE CHANNEL(Q.SALES.GATE) CHLTYPE(CLUSRCVR) CONNAME('localhost(1420)')
CLUSTER(Q.SALES) REPLACE
DEFINE QLOCAL (XMITQ.Q.SALES.SALESRV) USAGE(XMITQ) CLCHNAME(Q.SALES.SALESRV) REPLACE
```

```
DEFINE QLOCAL (XMITQ.SALES)          USAGE(XMITQ) CLCHNAME(SALES.*) REPLACE
DEFINE QLOCAL (XMITQ.FINANCE)        USAGE(XMITQ) CLCHNAME(FINANCE.*) REPLACE
```

4. SALES kuyruğunu SALES kümesinden kaldırın.

Run the **MQSC** command in [Şekil 58 sayfa 292](#):

```
ALTER QLOCAL(SALES) CLUSTER('Q.SALES') CLUSNL(' ')
```

Şekil 58. Satış kümesinden kuyruk yöneticisi SALESRV 'daki satış kuyruğunu kaldır

5. Kanalları yeni iletim kuyruklarına değiştirin.

Gereksinim, GATE kuyruk yöneticisinin kullanmakta olduğu tüm kanalları durdurup başlatmadır. Bunu en az sayıda komutla yapmak için, kuyruk yöneticisini durdurup başlatın.

```
endmqm -i GATE
stmqm GATE
```

Sonraki adım

1. Yeni yapılandırma çalışmalarını doğrulamak için örnek istek programını yeniden çalıştırın; bkz. [“2” sayfa 291](#). adım
2. GATE kuyruk yöneticilerindeki tüm küme iletim kuyruklarından akan iletileri izleyin:
 - a. Kuyruk izlemeyi açmak için, küme iletim kuyruklarının her birinin tanımlamasını değiştirin.

```
ALTER QLOCAL(SYSTEM.CLUSTER.TRANSMIT.
name) STATQ(ON)
```

- b. Check queue manager statistics monitoring is OFF , to minimize output, and set the monitoring interval to a lower value to perform multiple tests conveniently.

```
ALTER QMGR STATINT(60) STATCHL(OFF) STATQ(OFF) STATMQI(OFF) STATACLS(OFF)
```

- c. GATE kuyruk yöneticisini yeniden başlatın.
- d. Run the sample request program a few times to verify that an equal number of messages are flowing through SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SALESRV and SYSTEM.CLUSTER.TRANSMIT.QUEUE. İstekler, SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SALESRV aracılığıyla akış ve SYSTEM.CLUSTER.TRANSMIT.QUEUE aracılığıyla yanıt verir.

```
amqsmon -m GATE -t statistics
```

- e. Birkaç aralık içindeki sonuçlar aşağıdaki gibidir:

```
C:\Documents and Settings\Admin>amqsmon -m GATE -t statistics
MonitoringType: QueueStatistics
QueueManager: 'GATE'
IntervalStartDate: '2012-02-27'
IntervalStartTime: '14.59.20'
IntervalEndDate: '2012-02-27'
IntervalEndTime: '15.00.20'
CommandLevel: 700
ObjectCount: 2
QueueStatistics: 0
  QueueName: 'SYSTEM.CLUSTER.TRANSMIT.QUEUE'
  CreateDate: '2012-02-24'
  CreateTime: '15.58.15'
  ...
  Put1Count: [0, 0]
```

```

Put1FailCount: 0
PutBytes: [435, 0]
GetCount: [1, 0]
GetBytes: [435, 0]
...
QueueStatistics: 1
QueueName: 'SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SAVESRV'
CreateDate: '2012-02-24'
CreateTime: '16.37.43'
...
PutCount: [1, 0]
PutFailCount: 0
Put1Count: [0, 0]
Put1FailCount: 0
PutBytes: [435, 0]
GetCount: [1, 0]
GetBytes: [435, 0]
...
MonitoringType: QueueStatistics
QueueManager: 'GATE'
IntervalStartDate: '2012-02-27'
IntervalStartTime: '15.00.20'
IntervalEndDate: '2012-02-27'
IntervalEndTime: '15.01.20'
CommandLevel: 700
ObjectCount: 2
QueueStatistics: 0
QueueName: 'SYSTEM.CLUSTER.TRANSMIT.QUEUE'
CreateDate: '2012-02-24'
CreateTime: '15.58.15'
...
PutCount: [2, 0]
PutFailCount: 0
Put1Count: [0, 0]
Put1FailCount: 0
PutBytes: [863, 0]
GetCount: [2, 0]
GetBytes: [863, 0]
...
QueueStatistics: 1
QueueName: 'SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SAVESRV'
CreateDate: '2012-02-24'
CreateTime: '16.37.43'
...
PutCount: [2, 0]
PutFailCount: 0
Put1Count: [0, 0]
Put1FailCount: 0
PutBytes: [863, 0]
GetCount: [2, 0]
GetBytes: [863, 0]
...
2 Records Processed.

```

İlk aralıkta bir istek ve yanıt iletisi, ikincisinde iki kez gönderildi. You can infer that the request messages were placed on SYSTEM.CLUSTER.TRANSMIT.Q.SALES.SAVESRV, and the reply messages on SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Kümeleme: Küme iletim kuyruklarının değiştirilmesi

Var olan bir üretim kuyruğu yöneticisinin küme iletim kuyruklarına ilişkin değişikliklerin yürürlüğe nasıl getirileceğini planlayın.

Başlamadan önce

Anahtarlama işleminin yeni iletim kuyruğuna aktarılması gereken ileti sayısını azaltıyorsanız, geçiş işlemi daha hızlı tamamlanır. Devam etmeden önce iletim kuyruğunu boşaltma nedenleriyle ilgili [“Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır” sayfa 170](#) ' i okuyun.

Bu görev hakkında

Küme iletim kuyruklarında yapılan değişikliklerin yürürlüğe girmesi için iki yol seçeneğiniz vardır.

1. Kuyruk yöneticisinin değişiklikleri otomatik olarak gerçekleştirmesine izin verin. Bu varsayılandır. Kuyruk yöneticisi, bir küme gönderici kanalı sonraki başlatıldığında beklemedeki iletim kuyruğu değişikliklerine sahip küme gönderici kanallarını değiştirir.
2. Değişiklikleri el ile yapın. Bir küme gönderici kanalında, durdurulduğunda yapılan değişiklikleri yapabilirsiniz. Kümeyi, küme gönderici kanalı başlamadan önce bir küme iletim kuyruğundan başka bir kümeye geçirebilirsiniz.

Seçilecek iki seçenektan hangisine karar verirken hesaba katmanız gereken etkenler nelerdir ve anahtarı nasıl yönettiyorsunuz?

Yordam

- Seçenek 1: Kuyruk yöneticisinin değişiklikleri otomatik olarak gerçekleştirmesine izin verin; bkz. [“Etkin küme gönderici kanallarının başka bir küme aktarma kuyruklarına geçişi” sayfa 295](#).

Kuyruk yöneticisinin anahtarı sizin için gerçekleştirmesini istiyorsanız bu seçeneği belirleyin.

Bu seçeneği açıklamanın alternatif bir yolu da, kuyruk yöneticisinin, kanalı durdurmak için zorlamadan bir küme gönderici kanalını değiştirmesi olduğunu söylemeniz gerekir. Kanalı durdurmak için ve daha sonra kanalı başlatırken, anahtarı daha erken yapmak için kanalı zorlamaya gerek vardır. Anahtar, kanalı başlatıldığında başlar ve kanal çalışırken, 2. seçeneğinin farklı olduğu bir sırada çalışır. Seçenek 2 'de, kanalı durdurulduğunda anahtar gerçekleşir.

Bu seçeneği, anahtarın otomatik olarak gerçekleşmesine izin vererek seçerseniz, bir kümeden gönderici kanalı başlatıldığında geçiş işlemi başlar. Kanalı durdurulmamışsa, işlenecek bir ileti varsa, kanalı devre dışı duruma geldikten sonra başlar. Kanalı durdurulduysa, START CHANNEL komutunu kullanarak başlatın.

Anahtar işlemi, kanalı hizmet vermekte olduğu iletim kuyruğunda kümeden gönderici kanalı için bir ileti bırakılmaz tamamlanmaz tamamlanır. Bu durumda, kümelik gönderici kanalı için yeni gelen iletiler doğrudan yeni iletim kuyruğunda saklanır. O zamana kadar iletiler eski iletim kuyruğunda saklanır ve geçiş işlemi iletileri eski iletim kuyruğundan yeni iletim kuyruğuna aktarır. Küme gönderici kanalı, tüm anahtarlama işlemi sırasında yeni küme iletim kuyruğundan iletileri iletir.

Anahtar işlemi tamamlanınca sistemin durumuna bağlıdır. Değişiklikleri bir bakım penceresinde yapıyorsanız, geçiş sürecinin zamanında tamamlanıp tamamlanmayacağını önceden değerlendirin. Zaman içinde tamamlanıp tamamlanmayacağı, eski iletim kuyruğundan aktarımı bekleyen ileti sayısının sifıra ulaşmasına bağlıdır.

İlk yöntemin avantajı otomatidir. Bir dezavantaj, yapılandırma değişikliklerinin bir bakım penceresinden sınırlı olması durumunda, sistem, bakım penceresindeki anahtar sürecini tamamlamak için sistemi denetleyebileceğinden emin olmanız gerekir. Emin olamıyorsanız, seçenek 2 daha iyi bir seçim olabilir.

- Seçenek 2: Değişiklikleri el ile yapın; bkz. [“Durdurulmuş bir küme gönderici kanalının başka bir küme iletim kuyruğuna değiştirilmesi” sayfa 296](#).

Tüm anahtarlama işlemini el ile denetlemek ya da durdurulmuş ya da etkin olmayan bir kanalı değiştirmek istiyorsanız bu seçeneği belirleyin. Birkaç küme gönderici kanalı değiştiriyorsanız ve bir bakım penceresi sırasında anahtarı yapmak istiyorsanız, bu iyi bir seçimdir.

Bu seçeneğin alternatif bir açıklaması, kümeyi gönderen kanalı durdurulurken, kümeyi gönderen kanalı değiştirdiğinizi söylemeniz gerekir.

Bu seçeneği belirlerseniz, anahtar yerine geçtiğinde tam denetime sahip olur.

Bir bakım aralığı içinde, değiştirme işlemini sabit bir süre içinde tamamlamaktan emin olabilirsiniz. Anahtarın bir iletim kuyruğundan diğerine kaç ileti aktarılacağına başlanacağına saat. İleti gelmeye devam ederse, işlemin tüm iletileri aktararak aktarması biraz zaman alabilir.

Eski iletim kuyruğundan ileti aktarmadan kanalı değiştirme seçeneğiniz vardır. Anahtar "instant"(Anında).

Küme gönderici kanalını yeniden başlattığınızda, bu kanalı yeni atadığınız iletim kuyruğunda iletileri işlemeye başlar.

İkinci yöntemin avantajı, geçiş süreci üzerinde denetime sahip olmanız. Dezavantajı ise, anahtarlama olarak küme gönderici kanallarını belirlemeniz, gerekli komutları çalıştırmamız ve küme gönderen kanalının durdurulmasını önleyen tüm belirsiz kanalları çözmeniz gerekir.

İlgili kavramlar

Kullanılacak küme iletim kuyruğunun tipini nasıl seçecek?

Farklı küme iletim kuyruğu yapılandırma seçenekleri arasında seçim yapabilirsiniz.

İlgili görevler

Kümeleme: Birden çok küme iletim kuyruklarının yapılandırılması

Bu görevde, birden çok küme iletim kuyruklarını üç üst üste çakışan kümeye planlamak için adımları uygulayın. Gereksinimler, diğer tüm ileti akışlarından bir küme kuyruğuna ve farklı küme iletim kuyruklarında farklı kümelere ilişkin iletileri saklamak için, iletileri ayrı ayrı bir küme kuyruğuna yollamak içindir.

[“Etkin küme gönderici kanallarının başka bir küme aktarma kuyruklarına geçişi” sayfa 295](#)

Bu görev, etkin küme gönderici kanallarını değiştirmek için size üç seçenek sunar. Bir seçenek, kuyruk yöneticisinin anahtarı otomatik olarak gerçekleştirmesine izin vermek; bu da çalışan uygulamaları etkilemez. Diğer seçenekler, kanalları el ile durdurup başlatmak ya da kuyruk yöneticisini yeniden başlatmak için kullanılabilir.

[“Durdurulmuş bir küme gönderici kanalının başka bir küme iletim kuyruğuna değiştirilmesi” sayfa 296](#)

İlgili bilgiler

[“Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır” sayfa 170](#)

Etkin küme gönderici kanallarının başka bir küme aktarma kuyruklarına geçişi

Bu görev, etkin küme gönderici kanallarını değiştirmek için size üç seçenek sunar. Bir seçenek, kuyruk yöneticisinin anahtarı otomatik olarak gerçekleştirmesine izin vermek; bu da çalışan uygulamaları etkilemez. Diğer seçenekler, kanalları el ile durdurup başlatmak ya da kuyruk yöneticisini yeniden başlatmak için kullanılabilir.

Başlamadan önce

Küme iletim kuyruğu yapılandırmasını değiştirin. **DEFCLXQ** kuyruk yöneticisi özniteliğini değiştirebilir ya da iletim kuyruklarının **CLCHNAME** özniteliğini ekleyebilir ya da değiştirebilirsiniz.

Anahtarlama işleminin yeni iletim kuyruğuna aktarılması gereken ileti sayısını azaltıyorsanız, geçiş işlemi daha hızlı tamamlanır. Devam etmeden önce iletim kuyruğunu boşaltma nedenleriyle ilgili [“Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır” sayfa 170](#) 'i okuyun.

Bu görev hakkında

Küme iletim kuyruğu yapılandırması değişiklikleri yapmak için kendi planınızın üzerinde çalışmak için görevdeki adımları temel olarak kullanın.

Yordam

1. İsteğe bağlı: Geçerli kanal durumunu kaydet

Küme iletim kuyruklarına hizmet veren yürürlükteki ve saklanmış kanalların durumunu kaydedin. Aşağıdaki komutlarda, sistem kümesi iletim kuyruklarıyla ilişkili durum görüntülenir. Tanımladığınız küme iletim kuyruklarıyla ilişkili durumu görüntülemek için kendi komutlarınızı ekleyin. Bu iletim kuyruklarına ilişkin kanal durumunun görüntülenmesini kolaylaştırmak üzere tanımladığınız küme iletim kuyruklarına ad vermek için XMITQ. *ChannelName* gibi bir toplantı kullanın.

```
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
DISPLAY CHSTATUS(*) SAVED WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

2. Aktarım kuyruklarını değiştir.

- Hiçbir şey yapma. Kuyruk yöneticisi, durdurulduktan ya da devre dışı kaldıktan sonra yeniden başlatıldığında, küme gönderen kanallarını değiştirir.

Bir kuyruk yöneticisi yapılanışını değiştirme konusunda herhangi bir kuralınız ya da endişeniz yoksa bu seçeneği belirleyin. Çalışan uygulamalar değişikliklerden etkilenmez.

- Kuyruk yöneticisini yeniden başlatın. Tüm küme-gönderici kanalları istek üzerine otomatik olarak durdurulur ve yeniden başlatılır.

Tüm değişiklikleri hemen başlatmak için bu seçeneği belirleyin. Çalışmakta olan uygulamalar kuyruk yöneticisi tarafından durdurularak sona erdirilir ve yeniden başlatılır.

- Tek tek küme-gönderici kanallarını durdurun ve yeniden başlatın.

Birkaç kanalı hemen değiştirmek için bu seçeneği belirleyin. Çalışan uygulamalar, durdurulan ve ileti kanalını yeniden başlamanızın arasında ileti aktarımında kısa bir gecikme deneyimi yaşamanızı sağlar. Küme gönderici kanalı, durdurduğu süre dışında, çalışmakta olmaya devam eder. Anahtar işlemi sırasında iletiler eski iletim kuyruğuna teslim edilir, yeni iletim kuyruğuna geçiş işlemi tarafından aktarılır ve yeni iletim kuyruğundan, küme gönderen kanalına iletilir.

3. İsteğe bağlı: Kanalları değiştikçe izleyin

Anahtar sırasında kanal durumunu ve iletim kuyruğu derinliğini görüntüleyin. Aşağıdaki örnek, sistem kümesi iletim kuyruklarının durumunu görüntüler.

```
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
DISPLAY CHSTATUS(*) SAVED WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
DISPLAY QUEUE('SYSTEM.CLUSTER.TRANSMIT.*') CURDEPTH
```

4. İsteğe bağlı: Kuyruk yöneticisi hata günlüğüne yazılan "AMQ7341 The transmission queue for channel *ChannelName* switched from queue *QueueName* to *QueueName*" iletilerini izleyin.

Durdurulmuş bir küme gönderici kanalının başka bir küme iletim kuyruğuna değiştirilmesi

Başlamadan önce

Bazı yapılandırma değişiklikleri yapabilirsiniz ve bu değişiklikleri, etkilenen küme gönderen kanallarına başlamaksızın etkili yapmak isteyebilirsiniz. Diğer bir seçenek olarak, gerekli yapılandırma değişikliklerini, görevdeki adımlardan biri olarak gerçekleştirmenizi sağlar.

Anahtarlama işleminin yeni iletim kuyruğuna aktarılması gereken ileti sayısını azaltıyorsanız, geçiş işlemi daha hızlı tamamlanır. Devam etmeden önce iletim kuyruğunu boşaltma nedenleriyle ilgili "[Küme-gönderici kanalını farklı bir iletim kuyruğuna değiştirme işlemi nasıl çalışır](#)" sayfa 170 ' i okuyun.

Bu görev hakkında

Bu görev, durdurulmuş ya da etkin olmayan küme gönderen kanallarının hizmet verdiği iletim kuyruklarını değiştirir. Bu görevi, bir küme gönderici kanalı durdurulduğundan ve iletim kuyruğunu hemen değiştirmek istiyorsanız yapmanız gerekebilir. Örneğin, bir nedenden dolayı bir küme-gönderen kanalı başlatılmamıştır ya da başka bir yapılandırma sorunu vardır. Sorunu çözmek için, bir küme-gönderici kanalı yaratmaya

karar verirsiniz ve tanımladığınız yeni küme gönderici kanalına sahip eski kümeli gönderici kanalına ilişkin iletim kuyruğunu ilişkilendirmeye karar verirsiniz.

Küme iletim kuyruklarının yeniden yapılandırılırken denetlenmesini istediğiniz olası bir senaryo daha olası olur. Yeniden yapılandırmayı tam olarak denetlemek için, kanalları durdurursun, yapılandırmayı değiştirin ve daha sonra iletim kuyruklarını değiştirin.

Yordam

1. Değiştirmek istediğiniz kanalları durdurun.

- a) Değiştirmek istediğiniz, çalışmakta olan ya da etkin olmayan kanalları durdurun. Etkin olmayan bir küme gönderici kanalı durdurulurken, yapılandırma değişiklikleri yaparken bu kanal başlatılmasına engel olur.

```
STOP CHANNEL(ChannelName) MODE(QUIESCSE) STATUS(STOPPED)
```

2. İsteğe bağlı: Yapılandırma değişikliklerini yapın.

Örneğin, bkz. [“Kümeleme: Birden çok küme iletim kuyruklarının yapılandırılması” sayfa 286.](#)

3. Küme gönderici kanallarını yeni küme iletim kuyruklarına geçirmeniz gerekir.

```
runswchl -m QmgrName -c ChannelName
```

runswchl komutu, eski iletim kuyruğunda bulunan iletileri yeni iletim kuyruğuna aktarır. Bu kanala ilişkin eski iletim kuyruğunda bulunan iletilerin sayısı sifıra ulaştığında, anahtar tamamlanır. Komut zamanuyumlu. Komut değiştirme işlemi sırasında komut, aşama iletilerini pencereye yazar.

Aktarma aşaması sırasında, yeni iletim kuyruğuna, küme gönderen kanalına gönderilen yeni iletiler aktarılır.

Küme gönderen kanalı durdurulduğundan, iletiler yeni iletim kuyruğunda birikir. Durdurulan küme gönderen kanalının tersine, [“Etkin küme gönderici kanallarının başka bir küme aktarma kuyruklarına geçişi” sayfa 295](#) içinde [“2” sayfa 296](#) adımına adım atmasını sağlar. Bu adımda, küme-gönderici kanalı çalışıyor, bu nedenle iletilerin yeni iletim kuyruğunda oluşturulmaması olması gerekmez.

4. İsteğe bağlı: Kanalları değiştikçe izleyin

Farklı bir komut penceresinde, anahtar sırasında iletim kuyruğu derinliğini görüntüleyin. Aşağıdaki örnek, sistem kümesi iletim kuyruklarının durumunu görüntüler.

```
DISPLAY QUEUE('SYSTEM.CLUSTER.TRANSMIT.*') CURDEPTH
```

5. İsteğe bağlı: Kuyruk yöneticisi hata günlüğüne yazılan "AMQ7341 The transmission queue for channel *ChannelName* switched from queue *QueueName* to *QueueName*" iletilerini izleyin.

6. Durdurduğunuz küme gönderici kanallarını yeniden başlatın.

The channels do not start automatically, as you stopped them, placing them into DURDURULDU status.

```
START CHANNEL(ChannelName)
```

İlgili başvurular

[runswchl](#)

[KANALIN](#)

[KANAL DURDUR](#)

Kümeleme: Geçiş ve değişiklik en iyi uygulamaları

Bu konuda, IBM WebSphere MQ kümelerini planlama ve yönetme kılavuzu sağlanır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuzdur.

1. [“Bir Kümedeki Nesnelerin Taşınması” sayfa 298](#) (Herhangi bir düzeltme paketi ya da yeni IBM WebSphere MQ sürümü kurmadan, nesnelere bir kümenin içinde taşımaya ilişkin en iyi uygulamalar).

2. “Büyütmeler ve bakım kuruluşları” sayfa 299 (Bir çalışan küme mimarisini çalışır ve çalışır durumda tutmak için en iyi uygulamalar, bakım veya büyütmeler uygulanırken ve yeni mimariyi test etmek için).

Bir Kümedeki Nesnelerin Taşınması

Uygulamalar ve kuyrukları

Başka bir kuyruk yöneticisine ev sahipliği yapmak üzere bir kuyruk örneğini taşımanız gerekirken, düzgün bir geçiş sağlamak için iş yükü dengeleme parametreleriyle çalışabilirsiniz.

Yeni barındırılacağı bir kuyruk örneği yaratın, ancak uygulamanızın değişmeye hazır oluncaya kadar özgün yönetim ortamına ileti göndermeye devam etmek için küme iş yükü dengeleme ayarlarını kullanın. Bu, aşağıdaki adımlarla gerçekleştirilir:

1. Var olan kuyruğun **CLWL**RANK özelliğini yüksek bir değere (örneğin, beş) ayarlayın.
2. Kuyruğun yeni örneğini oluşturun ve **CLWL**RANK özelliğini sıfır olarak ayarlayın.
3. Yeni sistemin daha fazla yapılandırmasını tamamlayın; örneğin, uygulamaları kuyruğun yeni örneğine karşı konuşlandırmak ve tüketmeye başlayın.
4. Yeni kuyruk örneğinin **CLWL**RANK özelliğini, özgün örneğinden daha yüksek olacak şekilde ayarlayın; örneğin, dokuz.
5. Özgün kuyruk örneğinin sistemde kuyruğa alınan iletileri işlemesini ve kuyruğun silinmesini sağlayın.

Kuyruk yöneticilerinin tamamının taşınması

Kuyruk yöneticisi aynı anasistemde kalıyorsa, ancak IP adresi değişiyorsa, işlem şu şekildedir:

- DNS, doğru kullanıldığında, süreci basitleştirmeye yardımcı olabilir. Bağlantı adı (CONNAME) kanal özneliğini ayarlayarak DNS kullanımıyla ilgili bilgi için [ALTER CHANNEL](#) başlıklı konuya bakın.
- Bir tam havuzu taşıyorsanız, değişiklik yapmadan önce sorunsuz bir şekilde çalışan en az bir tam havuza (örneğin, örneğin kanal durumuyla ilgili sorun yok) sahip olduğundan emin olun.
- Trafik oluşturma işlemini önlemek için [SUSPEND QMGR](#) komutunu kullanarak kuyruk yöneticisini askıya alın.
- Bilgisayarın IP adresini değiştirin. CLUSTRVR kanal tanımınızda CONNAME alanında bir IP adresi kullanılıyorsa, bu IP adresi girişini değiştirin. Güncellemelerin her yerde kullanılabilir olduğundan emin olmak için DNS önbelleğinin temizlenmesi gerekebilir.
- Kuyruk yöneticisi tüm havuzlara yeniden bağlandığında, kanal otomatik tanımlamaları otomatik olarak kendini çözer.
- Kuyruk yöneticisi tam bir havuz barındırıyor ve IP adresi değişirse, el ile tanımlanan CLUSSDR kanallarını yeni konuma işaret etmek için tarafların en kısa zamanda bu tarafa geçtiğinden emin olun. Bu anahtar gerçekleştirilinceye kadar, bu kuyruk yöneticileri yalnızca geri kalan (değiştirilmemiş) tam havuzla iletişim kurabilirler ve yanlış kanal tanımlamasıyla ilgili uyarı iletileri görülebilir.
- Resume the queue manager using the [QMGR ' YI Sü](#) command.

Kuyruk yöneticisinin yeni bir anasisteme taşınması gerekiyorsa, kuyruk yöneticisi verilerinin kopyalanabileceği ve yedekten geri yüklenmesi mümkündür. Başka seçenek yoksa, bu işlem önerilmez; yeni bir makinede kuyruk yöneticisi yaratmak daha iyi olabilir ve önceki bölümde açıklandığı gibi, kuyrukları ve uygulamaları çoğaltmak daha iyi olabilir. Bu durum pürüzsüz bir baştan başlama/geri alma mekanizması sağlar.

Yedeklemeyi kullanarak tam bir kuyruk yöneticisini taşımaya karar verdiyseniz, aşağıdaki en iyi uygulamaları izleyin:

- Tüm süreci bir kuyruk yöneticisi geri yükleme işlemi olarak, işletim sistemi ortamınız için uygun olan sistem kurtarması için kullanabildiğiniz tüm işlemleri uygulayarak, yedekten geri yükleme işlemi geri yükleyin.

- Yerel olarak tutulan tüm küme bilgilerini (belirsiz olan otomatik tanımlı kanallar da içinde olmak üzere) atmak için geçişten sonra **REFRESH CLUSTER** komutunu kullanın ve bu bilgileri yeniden oluşturulmalıdır.

Not: Büyük kümeler için, **REFRESH CLUSTER** komutunun kullanımı devam ederken kümeyi kesintiye uğratabilir ve bundan sonra 27 gün aralıklarla küme nesnelere, ilgili tüm kuyruk yöneticilerine otomatik olarak durum güncellemeleri gönderdiğinde, bu işlem yine 27 gün aralıklarla kesintiye uğrayabilir. Bkz. [Büyük bir kümede yenilenme, kümenin performansını ve kullanılabilirliğini etkileyebilir.](#)

Bir kuyruk yöneticisi yaratırken kümedeki var olan bir kuyruk yöneticisinden kuruluş eşlenirken (daha önce bu konuda açıklandığı gibi), iki farklı kuyruk yöneticisini hiçbir zaman aynı şekilde davranmayın. Özellikle, yeni bir kuyruk yöneticisi için aynı kuyruk yöneticisi adını ve IP adresini vermeyin. Yeni bir kuyruk yöneticisinde 'bırakma' girişimi, IBM WebSphere MQ kümelerinde sık rastlanan bir sorun nedenidir. Önbellek, **QMID** özneliği de dahil olmak üzere güncellemeleri almayı bekler ve durum bozulabilir.

Aynı adı taşıyan iki farklı kuyruk yöneticisi yanlışlıkla oluşturulduysa, yanlış girişi kümeden çıkarmak için **RESET CLUSTER QMID** komutunu kullanmanız önerilir.

Büyütmeler ve bakım kuruluşları

"Büyük patlama senaryosundan" kaçının (örneğin, tüm küme ve kuyruk yöneticisi etkinliğini durdurmak, tüm kuyruk yöneticilerine tüm yükseltmeleri ve bakımı uygulamak, sonra her şeyi aynı anda başlatmak): Kümeler, kuyruk yöneticisi birlikte var olan birden çok sürümle çalışmaya devam edecek şekilde tasarlanmıştır, bu nedenle iyi planlanmış, aşamalı bir bakım yaklaşımı önerilir.

Yedek bir planın var mı:

- z/OS'ta geriye doğru geçiş PTF' leri uyguladınız mı?
- Yedekleri aldın mı?
- Hemen yeni küme işlevselliğini kullanmaktan kaçının: Tüm kuyruk yöneticilerinin yeni düzeye yükseltildiğinden emin oluncaya kadar bekleyin ve bunların hiçbirini geri almayacağından emin olun. Bazı kuyruk yöneticilerinin daha önceki bir düzeyin bulunduğu bir kümede yeni küme işlevinin kullanılması, tanımsız davranışlara yol açabilir. Örneğin, IBM WebSphere MQ Version 6.0'dan IBM WebSphere MQ Version 7.1 ' a yapılan taşıma sırasında, bir kuyruk yöneticisi bir küme konusu tanımlıyorsa, IBM WebSphere MQ Version 6.0 kuyruk yöneticileri tanımlamayı anlamaz ya da bu konuda yayınlayamaz.

Önce tam havuzları geçirin. Anlamadıkları bilgileri iletebilirlerse de, bunu sürdüremezler, bu nedenle kesinlikle gerekli olan bir yaklaşım değildir. Ek bilgi için [Kuyruk yöneticisi küme geçişbaşılıklı konuya](#) bakın.

İlgili kavramlar

[“Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma” sayfa 299](#)

Bir kümeye ilişkin olarak yerel olarak tutulan tüm bilgileri atmak ve bu bilgileri kümedeki tüm havuzlardan yeniden oluşturmak için **REFRESH CLUSTER** komutunu kullanıyorsunuz. Bu komutu, istisnai koşullar dışında, kullanmanıza gerek yoktur. Bunu kullanmanız gerekiyorsa, bunu nasıl kullanmanız gerektiği konusunda dikkat edilmesi gereken noktalar vardır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuzdur.

Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma

Bir kümeye ilişkin olarak yerel olarak tutulan tüm bilgileri atmak ve bu bilgileri kümedeki tüm havuzlardan yeniden oluşturmak için **REFRESH CLUSTER** komutunu kullanıyorsunuz. Bu komutu, istisnai koşullar dışında, kullanmanıza gerek yoktur. Bunu kullanmanız gerekiyorsa, bunu nasıl kullanmanız gerektiği konusunda dikkat edilmesi gereken noktalar vardır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuzdur.

Yalnızca REFRESH CLUSTER ' ı çalıştırın; bu işlemi gerçekten yapmanız gerekir

IBM WebSphere MQ küme teknolojisi, kümelenmiş bir kuyruğun değiştirilmesi gibi küme yapılandırmalarında herhangi bir değişikliğin otomatik olarak, bilgileri bilmesi gereken kümenin herhangi bir üyesi tarafından otomatik olarak bilinmesini sağlar. Bu bilgilerin yayılmasını sağlamak için daha fazla idari adıma gerek yoktur.

Bu tür bilgiler, gerekli olduğu kümedeki kuyruk yöneticilerine ulaşmazsa, örneğin kümelenmiş bir kuyruk, bir uygulama ilk kez açmayı denediğinde kümedeki başka bir kuyruk yöneticisi tarafından bilinmez; bu, küme altyapısında bir sorun anlamına gelir. Örneğin, kuyruk yöneticisi ile tam havuz kuyruk yöneticisi arasında bir kanal başlatılamaması mümkündür. bu nedenle tutarsızlıklar gözlenmekte olan her türlü durum araştırılmalıdır. If possible, resolve the situation without using the **REFRESH CLUSTER** command.

Bu ürün belgelerinin başka bir yerinde belgelenmiş nadir durumlarda ya da IBM desteği tarafından istendiğinde, bir kümeye ilişkin olarak yerel olarak tutulan tüm bilgileri atmak ve bu bilgileri kümedeki tüm havuzlardan yeniden oluşturmak için **REFRESH CLUSTER** komutunu kullanabilirsiniz.

Büyük bir kümede yenileme, kümenin performansını ve kullanılabilirliğini olumsuz etkileyebilir

Use of the **REFRESH CLUSTER** command can be disruptive to the cluster while it is in progress, for example by creating a sudden increase in work for the full repositories as they process the repropagation of queue manager cluster resources. Büyük bir kümede (yani, yüzlerce kuyruk yöneticisi) yeniliyorsanız, mümkünse günlük işteki komuttan kaçınmalısınız ve belirli tutarsızlıkları düzeltmek için alternatif yöntemler kullanırsınız. Örneğin, bir küme kuyruğu doğru bir şekilde geçirilmiyorsa, kümelenmiş kuyruk tanımlamasını güncellemeye ilişkin ilk araştırma tekniği (tanımını değiştirmek gibi), kuyruk yapılanışını küme genelinde yeniden yayılır. Bu işlem, sorunun saptanmasına ve geçici tutarsızlığını çözmeye yardımcı olabilir.

Alternatif yöntemler kullanılmıyorsa ve **REFRESH CLUSTER** komutunu büyük bir kümede çalıştırmanız gerekiyorsa, kullanıcı iş yüklerinin etkisini önlemek için yoğun zamanlarda ya da bir bakım penceresinde bunu yapmalısınız. Ayrıca, tek bir toplu işte büyük bir kümeyi yenilemekten kaçınmalı ve etkinliği "[Küme nesnelere otomatik güncellemeler gönderdiğinde performans ve kullanılabilirlik sorunlarını önle](#)" sayfa 300 içinde açıkladığı gibi basamaklandırmak da gerekir.

Küme nesnelere otomatik güncellemeler gönderdiğinde performans ve kullanılabilirlik sorunlarını önle

Kuyruk yöneticisinde yeni bir küme nesnesi tanımlandıktan sonra, bu nesne için bir güncelleme, tanımlama süresinden her 27 günde bir oluşturulur ve kümedeki her tam havuza gönderilir ve diğer ilgili kuyruk yöneticilerine iletilir. Bir kuyruk yöneticisine **REFRESH CLUSTER** komutunu verdiğinizde, belirtilen kümede yerel olarak tanımlanan tüm nesnelere bu otomatik güncellemeye ilişkin saati sıfırlayın.

Büyük bir kümeyi (yani, yüzlerce kuyruk yöneticisi) tek bir toplu işte ya da bir sistemi yapılandırma yedeklemeden yeniden oluşturma gibi diğer durumlarda, o kuyruk yöneticilerinin 27 gün sonra tüm nesne tanımlamalarını aynı anda tüm nesne tanımlarını tam havuzlara yeniden yayınlayacağı gibi başka bir durumda da yenilemişseniz. Bu yeniden, tüm güncellemeler tamamlanıncaya kadar, sistemin önemli ölçüde daha yavaş çalışmasına ya da kullanılamaz duruma gelmesine neden olabilir. Bu nedenle, büyük bir kümede birden çok kuyruk yöneticisini yenilemek ya da yeniden yaratmak için, etkinliği birkaç saat ya da birkaç gün boyunca kaydedemelisiniz; böylece, sonraki otomatik güncellemeler sistem başarımını düzenli olarak etkilemez.

Sistem kümesi geçmiş kuyruğu

Bir **REFRESH CLUSTER** gerçekleştirildiğinde, kuyruk yöneticisi, yenileme işleminden önce küme durumuna ilişkin bir anlık görüntü alır ve kuyruk yöneticisinde tanımlandıysa, **SYSTEM. CLUSTER. HISTORY. QUEUE (SCHQ)** üzerinde saklanır. Bu anlık görüntü, daha sonraki sorunlar durumunda, yalnızca IBM hizmet amaçlı olarak kullanılabilir. SCHQ, başlatma sırasında dağıtılmış kuyruk yöneticilerindeki varsayılan değer olarak tanımlanır. z/OS geçişi için, SCHQ el ile tanımlanmalıdır. SCHQ 'daki iletiler üç ay sonra sona erer.

İlgili kavramlar

[REFRESH CLUSTER çalıştırılırken uygulama sorunları görüldü](#)

[Yayınlama/abone olma kümeleri için Kümeyi Yenile](#)

İlgili başvurular

[MQSC Commands başvurusu: REFRESH CLUSTER](#)

Kümeleme: Kullanılabilirlik, çok eşgörünümler ve olağanüstü durumdan kurtarma

Bu konuda, IBM WebSphere MQ kümelerini planlama ve yönetme kılavuzu sağlanır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuzdur.

IBM WebSphere MQ Clustering 'in kendisi, High Availability (Yüksek Kullanılabilirlik) çözümü değildir, ancak bazı durumlarda, örneğin, farklı kuyruk yöneticilerinde bir kuyruğun birden çok örneğinin olması gibi, IBM WebSphere MQ'olanağını kullanarak hizmetlerin kullanılabilirliğini artırmak için kullanılabilir. Bu bölümde, IBM WebSphere MQ altyapısının bu tür bir mimaride kullanılabilmesi için mümkün olduğunca yüksek düzeyde kullanılabilir olmasını sağlama konusunda kılavuz bilgiler verilmektedir.

Küme kaynaklarının kullanılabilirliği

İki tam havuzu korumak için olağan önerinin nedeni, birinin kaybının kümenin düzgün bir şekilde çalıştırılması açısından kritik önem göstermemesidir. Her ikisi de kullanılamaz duruma gelse de, kısmi havuzlar tarafından tutulan mevcut bilgi için 60 günlük bir ek süre vardır; ancak, yeni ya da daha önce erişilmemiş kaynaklar (örneğin, örnek) bu etkinlikte kullanılamaz.

Uygulama kullanılabilirliğini artırmak için kümeleri kullanma

Bir küme, kuyruğun ve uygulamanın birden çok örneğini kullanarak yüksek düzeyde kullanılabilir uygulamalar tasarlamaya (örneğin, bir istek/yanıt tipi sunucu uygulaması) yardımcı olabilir.

Gerekliyse, öncelik öznitelikleri, örneğin bir kuyruk yöneticisi ya da kanal kullanılamaz hale gelmediği sürece 'canlı' uygulamaya tercih yapabilir. Bu, bir sorun ortaya çıktığında yeni iletileri işlemeye devam etmek için hızla geçiş yapmak için güçlü bir işlemdir.

Ancak, bir kümede belirli bir kuyruk yöneticisine teslim edilen iletiler yalnızca o kuyruk örneğinde tutulur ve kuyruk yöneticisi kurtarıncaya kadar işlenemez. Bu nedenle, gerçek veri yüksek kullanılabilirliği için, çok eşgörünümlü kuyruk yöneticileri gibi diğer teknolojileri de göz önünde bulundurmanız gerekebilir.

Çok eşgörünümlü kuyruk yöneticileri

Yazılım Yüksek Kullanılabilirliği (birden çok yönetim ortamı), var olan iletilerinizi sağlamak için en iyi yerleşik olanaktır. Daha fazla bilgi için bkz. "Yüksek kullanılabilirlik yapılandırmalarıyla WebSphere MQ ' un kullanılması" sayfa 310, "Çok eşgörünümlü bir kuyruk yöneticisi yarat" sayfa 338ve sonraki bölüm. Kümedeki tüm kuyruk yöneticileri en az IBM WebSphere MQ Version 7.0.1çalıştığı sürece, kümedeki herhangi bir kuyruk yöneticisi bu teknik kullanılarak yüksek kullanılabilirlikli kılınabilir. Kümedeki kuyruk yöneticileri önceki düzeylerde varsa, ikincil bir IP ' ye geçemezlerse, çok eşgörünümlü kuyruk yöneticileriyle bağlantı kaybedebilir.

Bu konuda daha önce ele alındığı gibi, iki tam havuz yapılandırıldığı sürece, bunlar hemen hemen nitelikleri tarafından kullanılabilir durumda olur. Gerekliyse, IBM WebSphere MQ yazılımı High Availability/Multi-instance kuyruk yöneticileri tüm havuzlar için kullanılabilir. Bu yöntemleri kullanmak için güçlü bir neden yoktur ve geçici kesintiler için bu yöntemler, hata durumunda yedek sisteme geçiş sırasında ek performans maliyetine neden olabilir. Örneğin, tek bir kanal kesintisi durumunda, iki tam havuz çalıştırmak yerine yazılım HA ' nın kullanılması önerilmez; ancak, bu bir kanal kesintisi durumunda başarısız olmaz, ancak kısmi havuzların küme kaynakları için sorgulayamaması durumunda bırakılabilir.

Olağanüstü durumdan kurtarma

Olağanüstü durumdan kurtarma, örneğin, bir kuyruk yöneticisinin verilerini saklayan disklerin bozulması nedeniyle kurtarma işlemi bozuk olduğu için, iyi olması zor; IBM WebSphere MQ yardım edebilir, ancak bunu otomatik olarak yapamayabilir. IBM WebSphere MQ içindeki tek 'true' olağanüstü durumdan kurtarma seçeneği (herhangi bir işletim sistemi ya da diğer temel kopyalama teknolojileri dışında) bir yedekten geri yüklenir. Bu durumlarda göz önünde bulundurulması gereken bazı kümeleme noktaları vardır:

- Olağanüstü durumdan kurtarma senaryolarını test ederken dikkatli olun. Örneğin, yedek kuyruk yöneticilerinin çalışmasını test etmek, yanlışlıkla canlı kümeye katılmak ve canlı küme kuyruğu yöneticilerindeki aynı adı taşıyan kuyrukları barındırarak 'çalma' iletileri başlatmak mümkün olduğu için, bunları çevrimiçi olarak aynı ağda çevrimiçi olarak getirirken dikkatli olun.
- Olağanüstü durumdan kurtarma testi, çalışmakta olan bir canlı kümeye müdahale etmemelidir. Paraziti önlemek için kullanılan teknikler şunlardır:
 - Ağ ayrılığı ya da ayırma güvenlik duvarı düzeyinde tamamdır.
 - Olağanüstü durumdan kurtarma sistemine kadar ya da gerçek bir olağanüstü durumdan kurtarma senaryosu ortaya çıkıncaya kadar, canlı SSL sertifikası yayınlanmaması.
- Bir kuyruk yöneticisinin yedeğini kümede geri yüklerken, yedeğin kümenin geri kalanıyla eşitlenmemiş olması mümkün olur. **REFRESH CLUSTER** komutu, güncellemeleri çözebilir ve kümeyle uyumlulaştırabilir; ancak, **REFRESH CLUSTER** komutu son çare olarak kullanılmalıdır. Bkz. “Kümeleme: REFRESH CLUSTER en iyi uygulamaları kullanma” sayfa 299. Komutu kullanmaya başvurmadan önce basit bir adımın kaçırılıp kaçırılmadığını görmek için kurum içi süreç belgelerini ve IBM WebSphere MQ belgelerini gözden geçirin.
- Herhangi bir kurtarma işlemi için, uygulamaların yeniden yürütülmesi ve veri kaybı ile ilgilenmesi gerekir. Kuyrukların bilinen bir duruma doğru temizlenip temizlenmeyeceğine ya da yeniden oynamaları yönetmek için başka bir yerde yeterli bilgi olup olmadığına karar verilmelidir.

Kümeleme: İzleme

Bu konuda, IBM WebSphere MQ kümelerinin planlanması ve yönetilmesi için kılavuz bilgiler yer alır. Bu bilgiler, müşterilerden gelen test ve geri bildirimlere dayalı bir kılavuzdur.

Küme içindeki uygulama iletilerinin izlenmesi

Genellikle, kuyruk yöneticisini terk eden tüm küme iletileri, iletiyi iletmek için hangi küme gönderen kanalının kullanıldığından bağımsız olarak SYSTEM . CLUSTER . TRANSMIT . QUEUE' den geçer. Her kanal, diğer tüm küme gönderen kanallarına paralel olarak bu kanal için hedeflenen iletileri boşaltıyor. Bu kuyruktaki iletilerin artan bir oluşturma işlemi, bir ya da daha çok kanalda sorun olduğunu gösterebilir ve araştırılmalıdır:

- Küme tasarımı için, kuyruğun derinliği uygun şekilde izlenmelidir.
- Aşağıdaki komut, iletim kuyruğunda bekleyen birden fazla iletiye sahip olan tüm kanalları döndürür:

```
DIS CHSTATUS(*) WHERE(XQMSGSA GT 1)
```

Tüm küme iletileri tek bir kuyrukla doldurulduğunda, hangi kanalda sorun olduğunu görmek her zaman kolay değildir. Bu komutun kullanılması, hangi kanaldan sorumlu olduğunu görmek için kolay bir yoldur.

Birden çok iletim kuyruğuna sahip olmak için bir küme kuyruk yöneticisi yapılandırabilirsiniz. If you change the queue manager attribute DEFCLXQ to CHANNEL, every cluster-sender channel is associated with a different cluster transmit queue. Diğer bir seçenek olarak, ayrı iletim kuyruklarını el ile de yapılandırabilirsiniz. Küme gönderen kanallarıyla ilişkili tüm küme iletim kuyruklarını görüntülemek için şu komutu çalıştırın:

```
DISPLAY CLUSQMGR (qmgrName) XMITQ
```

Küme iletim kuyruklarını, soldaki kuyruk adının sabit gövdesine sahip olma örüntülerini izletebilmesi için tanımlayın. Daha sonra, soysal bir kuyruk adı kullanarak, **DISPLAY CLUSMGR** komutu tarafından döndürülen tüm küme iletim kuyruklarının derinliğini sorgulayabilirsiniz:

```
DISPLAY QUEUE (qname*) CURDEPTH
```

Küme içindeki denetim iletilerini izleme

SYSTEM . CLUSTER . COMMAND . QUEUE kuyruğu, yerel kuyruk yöneticisi tarafından oluşturulan ya da kümedeki diğer kuyruk yöneticilerinden bu kuyruk yöneticisine gönderilen bir kuyruk yöneticisine ilişkin

tüm küme denetim iletilerini işlemek için kullanılır. Kuyruk yöneticisi doğru bir şekilde küme durumunu korurken, bu kuyruk sifıra doğru eğilimli olur. Ancak, bu kuyruktaki iletilerin derinliğinin geçici olarak büyüyebileceği durumlar vardır:

- Kuyrukta çok sayıda ileti olması, küme durumunda churn olduğunu gösterir.
- Önemli değişiklikler yaparken, kuyruğun bu değişiklikler arasında yerleşmesine izin verin. Örneğin, havuzları taşıırken, ikinci havuzu taşımadan önce kuyruğun sifıra ulaşmasına izin verin.

Bu kuyruktan bir ileti birikimi varken, küme durumu ya da kümeyle ilgili komutlara ilişkin güncellemeler işlenmez. İletiler uzun süre bu kuyruktan kaldırılmamışsa, daha fazla araştırma yapılması gerekir. Bu durumda, kuyruk yöneticisi hata günlüklerinin incelenmesi sırasında, bu duruma neden olan süreci açıklayabilir.

SYSTEM . CLUSTER . REPOSITORY . QUEUE , küme havuzu önbelleği bilgilerini bir ileti sayısı olarak tutar. Bu kuyruğun her zaman var olması, daha büyük kümeler için her zaman iletileceği iletiler içindir. Bu nedenle, bu kuyruktaki iletilerin derinliği endişe için bir sorun değildir.

İzleme günlükleri

Kümede ortaya çıkan sorunlar, bilgilerin önbelleğe alınması ve kümelenemenin dağıtılmış niteliği nedeniyle ortaya çıkan sorun nedeniyle, birçok gün (ve hatta aylar) boyunca dış belirtiler göstermeyebilir. Ancak, özgün sorun genellikle IBM WebSphere MQ hata günlüklerinde raporlanır. Bu nedenle, kümeleme ile ilgili olan herhangi bir ileti için bu günlüklerin etkin bir şekilde izlenmesi çok önemlidir. Bu iletiler, gerekli olduğu yerlerde herhangi bir işlem yapılması ile okunmalı ve anlaşılır olmalıdır.

Örneğin: Bir kümedeki bir kuyruk yöneticisiyle iletişimde bir kesme işlemi, kümelerin, bilgileri yeniden yayınlamak küme kaynaklarını düzenli olarak yeniden denetleyebilmesi nedeniyle silinmekte olan belirli küme kaynaklarının bilgisine neden olabilir. Bu tür bir olaya neden olabilecek bir uyarı, [AMQ9465](#) iletisi tarafından raporlanır. Bu ileti, sorunun soruşturulması gerektiğini belirtir.

Yük dengelemeye ilişkin özel noktalar

Küme, bir kuyruğun iki ya da daha fazla eşgürünümü arasında yük dengelemesi yaparken, uygulamaların tüketilmesi, her bir eşgürünümün her birinde ileti işlenmesi gerekir. Bu tür bir ya da daha çok uygulama ileti işlemeyi sona erdirirse ya da iletileri durdurursa, kümelemenin kuyruğun bu eşgürünümlerine ileti göndermeye devam etmesi mümkündür. Bu durumda, uygulamalar yeniden düzgün bir şekilde çalışmaya kadar bu iletiler işlenmez. Bu nedenle, uygulamaların izlenmesi çözümün önemli bir parçasıdır ve bu durumdaki iletilerin yeniden yönlendirilmesi için işlem yapılması gerekir. Bu örneğin, [The Cluster Queue Monitoring örnek programı \(AMQSCLM\)](#) örneğinde bu tür bir izlemeyi otomatikleştirecek bir mekanizmaya örnek olarak verilebilir.

Kullanılabilirlik, kurtarma ve yeniden başlatma

Kuyruk yöneticisi başarısız olursa ve sunucu ya da depolama hatasından sonra iletileri kurtarmak için kuyruk kullanılabilirliğini koruyarak uygulamalarınızı yüksek düzeyde kullanılabilir duruma getiriniz.

İstemci yeniden bağlanma olanağını kullanarak, bir istemciyi otomatik olarak bir kuyruk yöneticisi grubu arasında ya da bir kuyruk yöneticisi hatasından sonra çok eşgürümlü bir kuyruk yöneticisinin yeni etkin eşgürünümü arasında geçiş yapmak için istemci yeniden bağlantı özelliğini kullanın. Otomatik istemci yeniden bağlanması, Java için WebSphere MQ sınıfları tarafından desteklenmez.

Windows, UNIX, Linux ve IBM i platformları, sunucu uygulamalarını birden çok sunucuda tek bir kuyruk yöneticisi olarak çalışacak şekilde yapılandırılmış çok eşgürümlü bir kuyruk yöneticisine konuşturılmaktadır; etkin yönetim ortamını çalıştıran sunucu başarısız olursa, yürütme otomatik olarak farklı bir sunucudaki aynı kuyruk yöneticisinin yedek yönetim ortamına geçer. Sunucu uygulamalarını kuyruk yöneticisi hizmetleri olarak çalışacak şekilde yapılandırırsanız, yedek yönetim ortamı etkin olarak çalışan kuyruk yöneticisi yönetim ortamı olduğunda bunlar yeniden başlatılır.

You can configure WebSphere MQ as part of a platform-specific clustering solution such as Microsoft Cluster Server, or PowerHA for AIX (formerly HACMP on AIX) and other UNIX and Linux clustering solutions.

Sunucu uygulaması kullanılabilirliğini artırmanın başka bir yolu da, sunucu uygulamalarını bir kuyruk yöneticisi kümesinde birden çok bilgisayara konuşlandırmasıdır.

Bir ileti sistemi sistemi, sisteme girilen iletilerin hedef olarak teslim edilmesini sağlar. WebSphere MQ , bir ileti rotasını, **dspmqrte** komutunu kullanarak bir kuyruk yöneticisinden başka bir kuyruk yöneticisinden diğerine taşıyabilir. Bir sistem arızalanırsa, hata tipine ve sistemin konfigürasyonuna bağlı olarak iletiler çeşitli şekillerde kurtarılabilir.

WebSphere MQ , iletilerin giriş, iletim ve teslim edilmesini işleyen kuyruk yöneticilerinin etkinliklerinin kurtarma günlüklerini koruyarak iletilerin kaybedilmemesini sağlar.Üç tip kurtarma işlemi için bu günlükleri kullanır:

1. *Yeniden başlatma kurtarma*, when you stop WebSphere MQ in a planned way.
2. *Hata kurtarma*, when a failure stops WebSphere MQ.
3. *Ortam kurtarma*, zarar görmüş nesnelere geri yüklemek için.

Tüm durumlarda, kurtarma işlemi kuyruk yöneticisi durduğunda bulunduğu duruma geri yükler; ancak, uçuş içi işlemler geri döndürülmesi dışında, kuyruk yöneticisinin durduğu sırada uçuşta bulunan güncellemeler kuyruktan kaldırılır. Kurtarma işlemi, tüm kalıcı iletileri geri yükler; kalıcı olmayan iletiler işlem sırasında kaybedilebilir.

Otomatik istemci yeniden bağlantısı

Bir dizi bileşen yapılandırarak, istemci uygulamalarınızın otomatik olarak yeniden bağlanarak, ek kod yazmadan otomatik olarak yeniden bağlanabilmesini sağlayabilirsiniz.

Otomatik istemci yeniden bağlantısı *satır içi*. Bağlantı, istemci uygulama programındaki herhangi bir noktada otomatik olarak geri yüklenir ve nesnelere açmak için kullanılan tutamaçlar geri yüklenir.

Bunun tersine, el ile yeniden bağlantı, istemci uygulamasının MQCONN ya da MQCONNX kullanarak bir bağlantıyı yeniden yaratmasını ve nesnelere yeniden açabilmesini gerektirir. Otomatik istemci yeniden bağlantısı çok sayıda, ancak tüm istemci uygulamaları için uygun değil.

Çizelge 28 sayfa 305 , bir istemci iş istasyonuna kurulması gereken en eski IBM WebSphere MQ istemcisi desteği yayınını listeler. Bir uygulamanın otomatik istemci yeniden bağlantısını kullanacak bir uygulama için istemci iş istasyonlarını bu düzeylerden birine büyütmeniz gerekir. Çizelge 29 sayfa 305 , otomatik istemci yeniden bağlanmasını etkinleştirmek için diğer gereksinimleri listeler.

Yeniden bağlantı seçeneklerine program erişimi olan bir istemci uygulaması, yeniden bağlantı seçeneklerini ayarlayabilir. JMS ve XMS istemcileri dışında, bir istemci uygulamasının yeniden bağlantı oluşturma seçeneklerine erişimi varsa, yeniden bağlantı olaylarını işlemek için bir olay işleyicisi de yaratabilir.

Yeniden derleme ve bağlantı oluşturulmadan, var olan bir istemci uygulaması yeniden bağlantı desteğinden yararlanabiliyor:

- For a non-JMS client, set the `mqclient.ini` environment variable `DefRecon` to set reconnection options. Kuyruk yöneticisine bağlanmak için CCDT kullanın. İstemci çok eşgözümlü bir kuyruk yöneticisine bağlanacaksa, CCDT ' deki etkin ve beklemedeki kuyruk yöneticisi eşgözümlülerinin ağ adreslerini sağlayın.
- Bir JMS istemcisi için, bağlantı üreticisi yapısınıdaki yeniden bağlanma seçeneklerini belirleyin. When you use the WebSphere MQ Resource adapter or a JMS Client that is integrated in a Java EE Environment, automatic client reconnection might not be available. Bazı yönetilen ortamlarda bazı kısıtlamalar vardır; daha fazla bilgi için Java SE ve Java EE ortamlarındaki otomatik istemci yeniden bağlantısının kullanılması başlıklı konuya bakın.

Not: Otomatik istemci yeniden bağlantısı, Java için WebSphere MQ sınıfları tarafından desteklenmez.

Çizelge 28. Desteklenen istemciler

İstemci arabirimi	İstemci	Yeniden başlant seçeneklerine program eriYim	Yeniden bağlantı desteği
İleti alışverişi API	C, C + +, COBOL, Yönetilmeyen Visual Basic, XMS (Windowsüzerinde yönetilmeyen XMS)	7.0.1	7.0.1
	JMS (JSE ve Java EE istemci taşıyıcısı ve yönetilen taşıyıcılar)	7.0.1.3	7.0.1.3
	Java içinWebSphere MQ sınıfları	Desteklenmiyor	Desteklenmiyor
	Yönetilen XMS ve yönetilen .NET istemcileri: C#, Visual Basic	7.1	7.1
Diğer API 'ler	Windows Communication Foundation (Yönetilmeyen ¹)	Desteklenmiyor	7.0.1
	Windows Communication Foundation (Yönetilen ¹)	Desteklenmiyor	Desteklenmiyor
	Eksen 1	Desteklenmiyor	Desteklenmiyor
	Eksen 2	Desteklenmiyor	7.0.1.3
	HTTP (web 2.0)	Desteklenmiyor	7.0.1.3

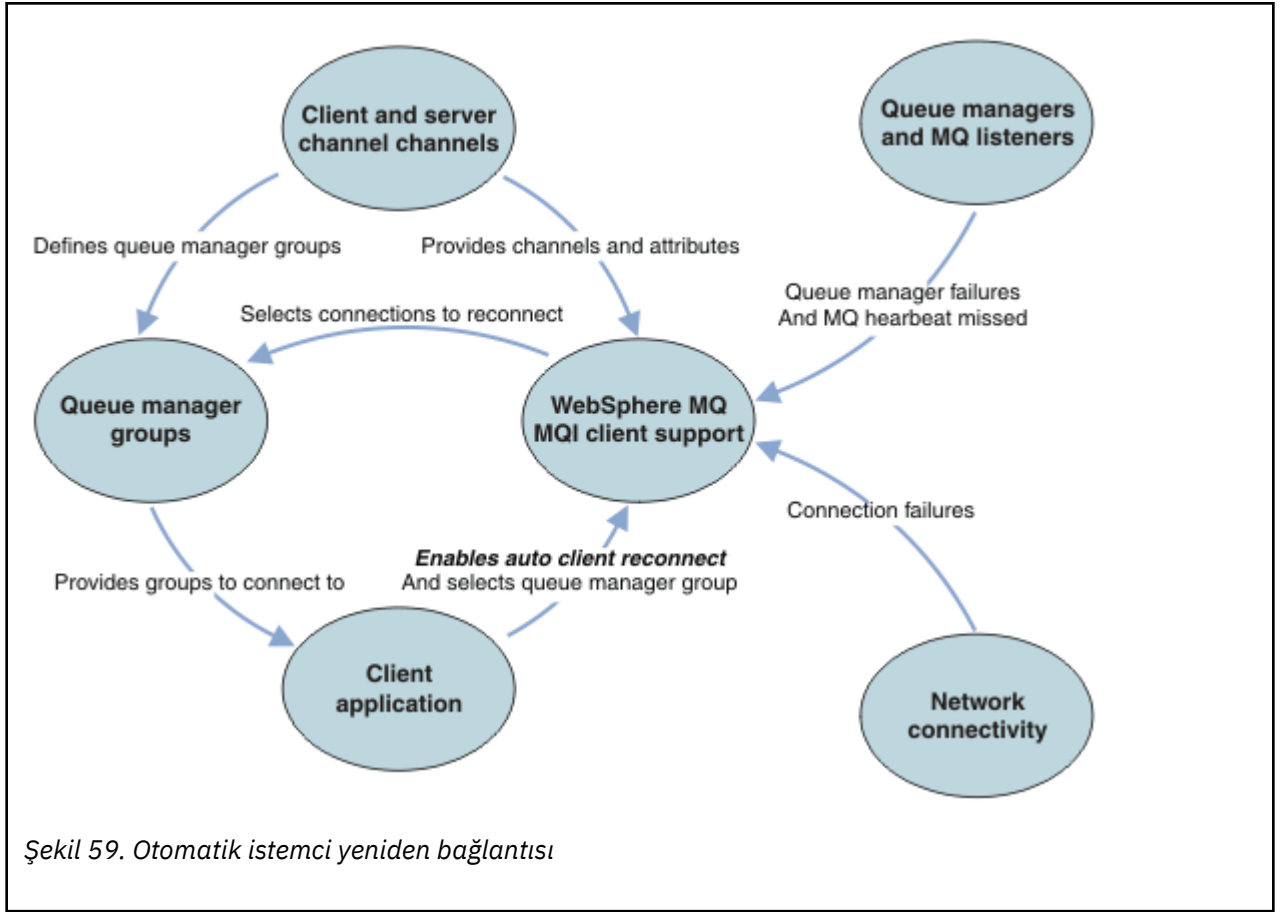
1. WCF bağ tanımı yapılanışındaki yönetilen ya da yönetilmeyen kipi ayarlayın.

Otomatik yeniden bağlanma, aşağıdaki yapılandırma gereksinimlerini içerir:

Çizelge 29. Otomatik yeniden bağlantı yapılandırma gereksinimleri

Bileşen	Gereksinim	Toplantı zorunluluğunun etkisi
WebSphere MQ MQI istemcisi kuruluşu	Bkz. Çizelge 28 sayfa 305	MQRC_OPTIONS_ERROR
WebSphere MQ Server kuruluşu	Düzy 7.0.1	MQRC_OPTIONS_ERROR
Kanal	SHARECNV > 0	MQRC_ENVIRONMENT_ERROR
Uygulama ortamı	Yivli olmalı	MQRC_ENVIRONMENT_ERROR
MQI	Aşağıdakilerden biri: <ul style="list-style-type: none"> MQCNO ileMQCONNX Seçenekler, MQCNO_RECONNECT veya MQCNO_RECONNECT_Q_MGRolarak ayarlanır. Defrecon=YES QMGR içinde mqclient.ini JMS ' de, bağlantı üreticisinin CLIENTRECONNECTORTIONS özelliğini ayarlayın. 	Bir bağlantı bozulduğunda ya da kuyruk yöneticisi sona erdiğinde ya da başarısız olduğundaMQCC_FAILED .

Şekil 59 sayfa 306 , istemci yeniden bağlantısında yer alan bileşenler arasındaki ana etkileşimleri gösterir.



İstemci uygulaması

İstemci uygulaması bir IBM WebSphere MQ MQI istemcidir.

- Varsayılan olarak, istemciler otomatik olarak yeniden bağlanmaz. MQCONNX MQCNO Option MQCNO_RECONNECT ya da MQCNO_RECONNECT_Q_MGR ayarını ayarlayarak, otomatik istemci yeniden bağlanma olanağını etkinleştirin.
- Birçok uygulama, ek kodlama olmadan otomatik yeniden bağlantıdan yararlanabilmekte olduğu şekilde yazılmıştır. mqclient.ini yapılandırma dosyasının kanallarında DefRecon özneliğini ayarlayarak, herhangi bir kodlama değişikliği yapmadan, var olan programlar için otomatik yeniden bağlantı özelliğini etkinleştirin.
- Şu üç seçenektten birini kullanın:
 - Programı, mantığın yeniden bağlantıdan etkilenmesi için değiştirin. Örneğin, eşitleme noktası içinde MQI çağrılarını yayınlayabilir ve gerçek olarak gerçek hareketleri yeniden g"nderebilirsiniz.
 - Yeniden bağlantıyı saptamak için bir olay işleyicisi ekleyin ve bağlantı yeniden kurulduğunda istemci uygulamasının durumunu geri yükleyin.
 - Otomatik yeniden bağlantı özelliğini etkinleştirmeyin: Bunun yerine, istemcinin bağlantısını keser ve aynı kuyruk yöneticisi grubunda çalışmakta olan başka bir kuyruk yöneticisi yönetim ortamını bulmak için yeni bir MQCONN ya da MQCONNX MQI çağrısını yayınlayın.

Bu üç seçenekte ilgili daha ayrıntılı bilgi için bkz. "Uygulama kurtarma" sayfa 384.

- Aynı addaki bir kuyruk yöneticisine yeniden bağlanmak, aynı kuyruk yöneticisinin eşgörünmesine yeniden bağlanmanız gerektiğini garanti etmez.

Aynı kuyruk yöneticisinin bir örneğine yeniden bağlanmak için bir MQCNO seçeneği MQCNO_RECONNECT_Q_MGR kullanın.

- Bir istemci, bir olay işleyicisini yeniden bağlantı durumunu bildirecek şekilde kaydettirebilir. Olay işleyicide geçirilen MQHCONN kullanılamıyor. Aşağıdaki neden kodları verilmiştir:

MQRC_RECONNECING

Bağlantı başarısız oldu ve sistem yeniden bağlanmayı deniyor. Birden çok yeniden bağlanma denemesi yapıldıysa, birden çok MQRC_RECONNECTING olayı alırsınız.

MQRC_REBAĞTED

Yeniden bağlanma yapıldı ve tüm tutamaçlar başarıyla yeniden kuruldu.

MQRC_RECONNECT_FAILED

Yeniden bağlantı başarısız oldu.

MQRC_RECONNECT_QMID_MISSATCH

Yeniden bağlanabilir bir bağlantı belirtildi MQCNO_RECONNECT_Q_MGR ve bağlantı, farklı bir kuyruk yöneticisine yeniden bağlanmayı denedi.

MQRC_RECONNECT_Q_MGR_REQD

An option, such MQMO_MATCH_MSG_TOKEN in an MQGET call, was specified in the client program that requires reconnection to the same queue manager.

- Yeniden bağlanabilir bir istemci, otomatik olarak yalnızca *bundan sonra* bağlanmasını yeniden bağlamayı başarabilir. Yani, MQCONNX ' un kendisi başarısız olursa yeniden denenmez. For example, if you receive the return code 2543 - MQRC_STANDBY_Q_MGR from MQCONNX, reissue the call after a short delay.

MQRC_RECONNECT_UYUMSUZ

Yeniden bağlanma seçenekleri ayarlandığında uygulama MQPMO_LOGICAL_ORDER (MQPUT ve MQPUT1ile) ya da MQGMO_LOGICAL_ORDER (MQGETile birlikte) kullanmayı denediğinde bu neden kodu döndürülür. Neden kodunun döndürülmesinin nedeni, uygulamaların bu tür durumlarda hiçbir zaman yeniden bağlantı kullanmayacağından emin olmaktır.

MQRC_CALL_INTERRUPT

Bu neden kodu, Kesinleştirme çağrısının yürütülmesi sırasında bağlantı kesildiğinde ve istemci yeniden bağlandığında döndürülür. Bir kalıcı iletinin, eşitleme noktasının dışında bir MQPUT değeri de, aynı neden kodunun uygulamaya geri döndürülmesine neden olur.

Çok eşgörünümlü kuyruk yöneticileri

Çok eşgörünümlü bir kuyruk yöneticisi otomatik istemci yeniden bağlantısını kullanarak, çok eşgörünümlü bir kuyruk yöneticisi yedek yönetim ortamını etkinleştirdikten sonra, WebSphere MQ MQI istemci uygulamalarını yeniden başlatmayı basitleştirin.

Çok eşgörünümlü bir kuyruk yöneticisinin yedek yönetim ortamı tipik olarak, etkin yönetim ortamına farklı bir ağ adresinde yer alıyor. İstemci bağlantı tanımı çizelgesine (CCDT) her iki yönetim ortamının ağ adreslerini de ekleyin. **CONNAME** parametresine ilişkin ağ adreslerinin bir listesini sağlayın ya da CCDT ' de kuyruk yöneticisi için birden çok satır tanımlayın.

Genel olarak, WebSphere MQ MQI istemcileri, kuyruk yöneticisi grubundaki herhangi bir kuyruk yöneticisine yeniden bağlanır. Bazen bir WebSphere MQ MQI istemcisinin yalnızca aynı kuyruk yöneticisine yeniden bağlanmasını istersiniz. Kuyruk yöneticisine benzeşimi olabilir. İstemcinin farklı bir kuyruk yöneticisine yeniden bağlanmasını önleyebilirsiniz. MQCNO seçeneğini ayarlayın, MQCNO_RECONNECT_Q_MGR. The WebSphere MQ MQI client fails if it reconnects to a different queue manager. MQCNO seçeneğini ayarladıysanız, MQCNO_RECONNECT_Q_MGR, aynı kuyruk yöneticisi grubuna başka kuyruk yöneticilerini eklemeyin. Bağlantı yeniden bağlandığı kuyruk yöneticisi, bağlı olduğu kuyruk yöneticisi değilse, istemci bir hata döndürür.

Kuyruk yöneticisi grupları

İstemci uygulamasının her zaman aynı adı taşıyan bir kuyruk yöneticisine, aynı kuyruk yöneticisine ya da istemci bağlantı çizelgesindeki aynı QMNAME değeriyle tanımlanmış kuyruk yöneticilerine bağlanıp bağlanmayacağını belirleyebilirsiniz.

- İstemci kanalı tanımlamasındaki kuyruk yöneticisi *ad* özniteliği QMNAME , bir kuyruk yöneticisi grubunun adıdır.

- İstemci uygulamanızda, MQCONN ya da MQCONNX QmgıName parametresinin değerini bir kuyruk yöneticisi adına ayarladıysanız, istemci yalnızca o adı taşıyan kuyruk yöneticilerine bağlanır. Kuyruk yöneticisi adını bir yıldız işareti (*) ile önlediğinizde, istemci kuyruk yöneticisi grubundaki herhangi bir kuyruk yöneticisine aynı QMNAME değerini içeren bir kuyruk yöneticisine bağlanır. Tam açıklama için bkz. [CCDT](#) 'deki kuyruk yöneticisi grupları.

Kuyruk paylaşım grupları

z/OS kuyruk paylaşım gruplarıyla otomatik istemci yeniden bağlantısı, başka bir ortam olarak yeniden bağlantı sağlamak için aynı mekanizmaları kullanır. İstemci, özgün bağlantı için yapılandırılmış olduğu kuyruk yöneticisi seçimine yeniden bağlanacak. Örneğin, istemci kanal tanımlama çizelgesini kullanırken, denetimcinin çizelgedeki tüm girişlerin aynı z/OS kuyruk paylaşım grubuna çözülmesini sağlaması gerekir.

İstemci ve sunucu kanalı tanımlamaları

İstemci ve sunucu kanalı tanımları, bir istemci uygulamasının yeniden bağlanabileceği kuyruk yöneticisi gruplarını tanımlar. Tanımlamalar, yeniden bağlantıların seçimini ve zamanlamasını ve güvenlik gibi diğer etkenleri yönetir; ilgili konulara bakın. Yeniden bağlantı için göz önünde bulundurulacak en ilgili kanal öznitelikleri iki grupta listelenir:

İstemci bağlantısı öznitelikleri

Bağlantı benzerliği (BENZEŞİMİ)BENZEŞİMİ

Bağlantı benzerliği.

İstemci kanal ağırlığı (CLNTWGHT)CLNTWGHT

Müşteri kanal ağırlığı.

Bağlantı adı (CONNNAME)CONNNAME

Bağlantı bilgileri.

Heartbeat interval (HBINT)HBINT

Kalp atışı aralığı. Sunucu bağlantı kanalındaki sağlıklı işletim bildirim aralığını ayarlayın.

Canlı Tutma Aralığı (KAINT)KAINT

Canlı tutma aralığı. Sunucu bağlantı kanalındaki canlı tutma aralığını ayarlayın.

KAINT ' in yalnızca z/OS için geçerli olduğunu unutmayın.

Kuyruk yöneticisi adı (QMNAME)QMNAME

Kuyruk yöneticisi adı.

Sunucu bağlantısı öznitelikleri

Heartbeat interval (HBINT)HBINT

Kalp atışı aralığı. İstemci bağlantı kanalındaki sağlıklı işletim bildirim aralığını ayarlayın.

Canlı Tutma Aralığı (KAINT)KAINT

Canlı tutma aralığı. İstemci bağlantı kanalındaki canlı tutma aralığını ayarlayın.

KAINT ' in yalnızca z/OS için geçerli olduğunu unutmayın.

KAINT , bir ağ katmanı sağlıklı işletim bildirim ve HBINT , istemci ile kuyruk yöneticisi arasındaki WebSphere MQ sağlıklı işletim bildirim. Bu kalp atışlarının daha kısa bir zamana ayarlanması iki amaca hizmet eder:

1. Bağlantıdaki etkinliği benzeterek, etkin olmayan bağlantıları kapatmaktan sorumlu olan ağ katmanı yazılımı bağlantınızı sona erdirmek için daha az olası olur.
2. Bağlantı sona erdirilirse, bozuk bağlantı algılanmadan önce gecikme süresi kısaltılır.

Varsayılan TCP/IP canlı tutma aralığı iki saattir. KAINT ve HBINT özniteliklerini daha kısa bir zamana ayarlamayı düşünün. Bir ağın normal davranışının, otomatik yeniden bağlantı gereksinmesine uygun olduğunu varsaymayın. Örneğin, bazı güvenlik duvarları etkin olmayan bir TCP/IP bağlantısını 10 dakikadan kısa bir süre sonra kapatabilir.

Ağ bağlantırlığı

Yalnızca, ağ tarafından WebSphere MQ MQI istemcisine geçirilen ağ hataları, istemcinin otomatik yeniden bağlanma yeteneği tarafından işlenir.

- İletim tarafından otomatik olarak gerçekleştirilen yeniden bağlantılar IBM WebSphere MQ' a görünmezdir.
- HBINT uyarı, WebSphere MQ' da görünmez olan ağ hatalarıyla başa çıkabilmenize yardımcı olur.

Kuyruk yöneticileri ve WebSphere MQ dinleyicileri

İstemci yeniden bağlantısı sunucu hatası, kuyruk yöneticisi hatası, ağ bağlantırlığı hatası ve bir denetimci tarafından başka bir kuyruk yöneticisi yönetim ortamına geçilerek tetiklenir.

- Çok eşgörünümlü bir kuyruk yöneticisi kullanıyorsanız, etkin kuyruk yöneticisi yönetim ortamından bir yedek yönetim ortamına denetim değiştirdiğinizde, istemci yeniden bağlanma işlemi için ek bir neden oluşur.
- Bir kuyruk yöneticisini varsayılan **endmqm** komutu kullanılarak sona erdirmek, otomatik istemci yeniden bağlantısını tetiklemez. Otomatik istemci yeniden bağlantısı istemek için **endmqm** komutuna **-r** seçeneğini ya da sona erdirildikten sonra beklemedeki bir kuyruk yöneticisi örneğine aktarmak için **-s** seçeneğini ekleyin.

WebSphere MQ MQI istemcisi otomatik yeniden bağlantı desteği

WebSphere MQ MQI istemcisinde otomatik istemci yeniden bağlantı desteğini kullanırsanız, istemci uygulaması otomatik olarak yeniden bağlanır ve kuyruk yöneticisine yeniden bağlanmak için bir MQCONN ya da MQCONNX MQI çağrısı yayınlamadan işlemeye devam eder.

- Otomatik istemci yeniden bağlantısı şu oluştumlardan biri tarafından tetiklenir:
 - kuyruk yöneticisi hatası
 - bir kuyruk yöneticisini sonlandırmak ve **-r**, yeniden bağlanma, **endmqm** komutunda yeniden bağlanma seçeneği
- The MQCONNX MQCNO options control whether you have enabled the automatic client reconnection. Seçenekler, [Yeniden yapılandırma seçenekleri](#) içinde açıklanır.
- Otomatik istemci yeniden bağlantı sorunları, uygulamanızın bağlantı tanıtıcısını ve diğer açık nesnelere geri yüklenmesi için uygulamanızı çağırır; böylece, programınız bozuk bağlantıyla sonuçlanan herhangi bir MQI hatalarını işledikten sonra olağan işlemeye devam edebilir. Bkz. [“Otomatik olarak yeniden bağlanan bir istemcinin kurtarılması”](#) sayfa 386.
- Bağlantı için bir kanal çıkış programı yazdıysanız, çıkış bu ek MQI çağrılarını alır.
- Yeniden bağlantı başladığında ve bittiğinde tetiklenen bir yeniden bağlantı olay işleyiciye kayıt yapabilirsiniz.

Yeniden bağlantı bir dakikadan fazla sürmese de, kuyruk yöneticisinin yönetmek için çok sayıda kaynağı olabileceği için yeniden bağlanma işlemi daha uzun sürebilir. Bu süre içinde, bir istemci uygulaması, WebSphere MQ kaynaklarına ait olmayan kilitleri tutuyor olabilir. İstemcinin yeniden bağlantı için bekleyeceği süreyi sınırlandırmak için yapılandırabileceğiniz bir zaman aşımı değeri vardır. Değer (saniye olarak) `mqclient.ini` dosyasında ayarlanır.

```
Channels:  
MQReconnectTimeout = 1800
```

Zaman aşımı süresi sona erdikten sonra yeniden bağlantı kurma girişiminde bulunmuyorsunuz. Sistem, zaman aşımı süresinin dolduğunu saptadığında bir MQRC_RECONNECT_FAILED hatası döndürür.

Konsol iletisi izleme

Özellikle, kuyruk yöneticisi ya da kanal başlatıcısı tarafından, özellikle önemli bir şekilde dikkate alınması gereken bir dizi bilgi iletisi yayınlandı. Bu iletiler, kendilerinde bir sorun olduğunu göstermez, ancak bu iletiler, potansiyel olarak ele alınması gereken olası bir sorunu belirttikleri için izlenmede yararlı olabilir.

Bu iletinin varlığı, bir kullanıcı uygulamasının sayfa kümesine çok sayıda ileti koyduğunu da gösterebilir; bu da daha büyük bir sorunun belirtisi olabilir:

- Denetimli olmayan döngü gibi iletileri PUT ' nin (PUT) içeren kullanıcı uygulamalarıyla ilgili bir sorun.
- Kuyruktan ileti alan bir kullanıcı uygulaması artık işlev görmeyecek.

Yüksek kullanılabilirlik yapılandırmalarıyla WebSphere MQ ' un kullanılması

If you want to operate your WebSphere MQ queue managers in a high availability (HA) configuration, you can set up your queue managers to work either with a high availability manager, such as PowerHA for AIX (formerly HACMP) or the Microsoft Cluster Service (MSCS), or with WebSphere MQ multi-instance queue managers.

Aşağıdaki yapılandırma tanımlamalarından haberdar olmanız gerekir:

Kuyruk yöneticisi kümeleri

Bir ya da daha çok bilgisayarda iki ya da daha çok kuyruk yöneticisi grubu, otomatik ara bağlantı sağlar ve yük dengeleme ve yedeklilik için kuyrukların bunlar arasında paylaşılmasına olanak sağlar.

HA kümeleri

HA kümeleri, iki ya da daha fazla bilgisayar ve disk gibi kaynak gruplarıdır; bunlardan biri başarısız olursa, yüksek kullanılabilirlikli bir yönetici (HACMP (UNIX) ya da MSCS (Pencereler), bir *yedek sisteme geçiş* işlemi gerçekleştirir), bu şekilde yapılandırılmış ve yapılandırılmış olan, disk ve ağ gibi kaynaklar gruplarıdır. Hata durumunda yedek sisteme geçiş işlemi, uygulamaların durum verilerini kümedeki başka bir bilgisayara aktarır ve bu uygulamaların çalışmasını yeniden başlatır. Bu, HA kümesi içinde çalışan hizmetlerin yüksek düzeyde kullanılabilirliğini sağlar. IBM WebSphere MQ kümeleri ile HA kümeleri arasındaki ilişki, [“HA kümelerinin kuyruk yöneticisi kümeleri ile ilişkisi” sayfa 311](#) içinde açıklanmıştır.

Çok eşgörünümlü kuyruk yöneticileri

Aynı kuyruk yöneticisinin iki ya da daha çok bilgisayarda yapılandırılmış eşgörünümleri. Birden çok yönetim ortamı başlatılarak, bir eşgörünüm etkin yönetim ortamı olur ve diğer yönetim ortamları standart olarak olur. Etkin yönetim ortamı başarısız olursa, farklı bir bilgisayarda çalışan bir yedek yönetim ortamı otomatik olarak sona erir. You can use multi-instance queue managers to configure your own highly available messaging systems based on WebSphere MQ, without requiring a cluster technology such as HACMP or MSCS. HA kümeleri ve çok eşgörünümlü kuyruk yöneticileri, kuyruk yöneticilerini yüksek kullanılabilirlikli yapmak için alternatif yollardır. Bir HA kümesine çok eşgörünümlü bir kuyruk yöneticisi koyarak bunları birleştirmeyin.

Çok eşgörünümlü kuyruk yöneticileri ve HA kümeleri arasındaki farklar

Çok eşgörünümlü kuyruk yöneticileri ve HA kümeleri, kuyruk yöneticilerinize yüksek düzeyde kullanılabilirlik elde etmek için alternatif yollardır. burada iki yaklaşım arasındaki farkları vurgulayan bazı noktalar yer alıyor.

Çok eşgörünümlü kuyruk yöneticileri aşağıdaki özellikleri içerir:

- WebSphere MQ ile bütünleştirilmiş temel hata durumunda yedek sisteme geçiş desteği
- HA kümesinden daha hızlı geçiş
- Basit yapılandırma ve işlem
- WebSphere MQ Explorer ile bütünleştirme

Çok eşgörünümlü kuyruk yöneticilerine ilişkin sınırlamalar şunlardır:

- Yüksek düzeyde kullanılabilir, yüksek performanslı ağ üzerinde çalışılan depolama
- Kuyruk yöneticisi IP adresini değiştiremediğinde IP adresini değiştirdiğinden, daha karmaşık ağ yapılandırması

HA kümeleri aşağıdaki özellikleri içerir:

- Uygulama sunucusu ya da veritabanı gibi birden çok kaynağı koordine etme yeteneği

- İki düğümden daha fazla düğüm oluşturan kümeler dahil olmak üzere daha esnek yapılandırma seçenekleri
- İşletmen müdahalesi olmadan birden çok kez yedek sisteme geçiş yapabilirsiniz
- Hata durumunda yedek sisteme geçişin bir parçası olarak kuyruk yöneticisinin IP adresinin devralınması

HA kümelerinin sınırlamaları şunlardır:

- Ek ürün satın alma ve becerileri gerekli
- Kümenin düğümleri arasında geçiş yapabilen diskler gereklidir
- HA kümelerinin yapılandırması görece olarak karmaşıktır
- Hata durumunda yedek sisteme geçiş, tarihsel olarak yavaş olmakla birlikte, yeni HA kümesi ürünleri bu durumu iyileştiriyor.
- Kuyruk yöneticileri gibi kaynakları izlemek için kullanılan komut dosyalarında eksiklikler varsa gereksiz sadakatler oluşabilir.

HA kümelerinin kuyruk yöneticisi kümeleri ile ilişkisi

Kuyruk yöneticisi kümeleri, denetimi azaltır ve kuyruk yöneticisi küme kuyrukları eşgörünümlerinde iletilerin yük dengelemesini sağlar. Ayrıca, kuyruk yöneticisinden oluşan bir hata nedeniyle, ileti alışverişi uygulamalarının bir kuyruk yöneticisi kümesinin yürürlükte kalan eşgörünümlerine erişmeye devam edebileceğinden, tek bir kuyruk yöneticisinden daha yüksek kullanılabilirlik sunarlar. Ancak, kuyruk yöneticisi kümeleri tek başına kuyruk yöneticisi hatasının otomatik olarak saptanması ve kuyruk yöneticisi yeniden başlatma ya da hata durumunda yedek sisteme geçiş özelliğinin otomatik olarak saptanması sağlamaz. HA kümeleri bu özellikleri sağlar. İki tip küme, iyi bir etki için birlikte kullanılabilir.

UNIX and Linux üzerinde yüksek kullanılabilirlikli küme içeren WebSphere MQ ' un kullanılması

You can use WebSphere MQ with a high availability (HA) cluster on UNIX and Linux platforms: for example, PowerHA for AIX (formerly HACMP), Veritas Cluster Server, HP Serviceguard, or a Red Hat Enterprise Linux cluster with Red Hat Cluster Suite.

HA kümelerini yapılandırmaya yardımcı olması için önce WebSphere MQ Version 7.0.1, SupportPac MC91 sağlanana kadar sağlanmıştır. WebSphere MQ Version 7.0.1 provided a greater degree of control than previous versions over where queue managers store their data. Bu, bir HA kümesinde kuyruk yöneticilerini yapılandırmanın daha kolay olmasını sağlar. SupportPac MC91 ile sağlanan komut dosyalarının çoğu artık gerekli değildir ve SupportPac geri çekilir.

Bu bölümde ["HA kümesi yapılandırmaları" sayfa 311](#), [HA kümelerinin kuyruk yöneticisi kümelerine ilişkin ilişkisi](#), ["WebSphere MQ istemcileri" sayfa 312](#) ve ["Bir HA kümesinde çalışan WebSphere MQ" sayfa 312](#) tanıtılır ve adımlar boyunca size yol gösterir ve HA kümesi ile kuyruk yöneticilerini yapılandırmak için uyarlayabileceğiniz örnek komut dosyaları sağlar.

Bu bölümde açıklanan yapılandırma adımlarıyla ilgili yardım almak için, ortamınıza özgü HA kümesi belgelerine bakın.

HA kümesi yapılandırmaları

In this section the term *düğüm* is used to refer to the entity that is running an operating system and the HA software; "computer", "system" or "machine" or "partition" or "blade" might be considered synonyms in this usage. Tüm küme düğümlerinin WebSphere MQ iş yükünü çalıştırdığı karşılıklı devralma da dahil olmak üzere yedek ya da devralma yapılandırmalarını ayarlamanıza yardımcı olması için WebSphere MQ olanağını kullanabilirsiniz.

Bekleme yapılandırması, en temel HA kümesi yapılandırmasıdır. Bu yapılandırma, diğer düğüm yalnızca yedekte bekleme olarak hareket ederken, bir düğümün iş yaptığı en temel HA kümesi yapılandırmasıdır. Bekleme düğümü iş gerçekleştiriyor ve boşta olarak adlandırılıyor; bu yapılandırma bazen *cold standby* olarak adlandırılır. Bu tür bir yapılandırma, yüksek düzeyde donanım yedekliliği gerektirir. Donanıma ilişkin tasarruf sağlamak için, bu yapılandırmanın tek bir yedek düğümle birden çok işçi

düğümüne sahip olmasını sağlamak mümkündür. Bunun amacı, beklemedeki düğümün diğer herhangi bir işçi düğümünün işini devralabileceğidir. Bu yapılandırma hala yedek yapılandırma olarak ve bazen de "N+1" yapılandırması olarak adlandırılıyor.

Bir *devralma* yapılandırması, düğüm hatası durumunda tüm düğümlerin bazı iş ve kritik iş işlemlerini gerçekleştirebileceği daha gelişmiş bir yapılandırmadır.

tek taraflı devralma yapılandırması, beklemedeki bir düğümün bazı ek, kritik olmayan ve taşınmaz işleri gerçekleştireceği bir yapılandırmadır. Bu yapılandırma, beklemedeki bir yapılandırmaya benzer, ancak beklemedeki düğüm tarafından gerçekleştirilen iş (kritik olmayan) çalışmalarıyla aynıdır.

karşılıklı devralma yapılandırması, tüm düğümlerin yüksek düzeyde kullanılabilir (taşınabilir) bir çalışma gerçekleştirmekte olduğu bir yapılandırmadır. Bu tip bir HA kümesi yapılandırması, tüm düğümlerin kritik iş yükünü etkin bir şekilde işlediğini belirtmek için bazen "Etkin/Etkin" olarak da adlandırılır.

Genişletilmiş yedek yapılandırma ya da devralma yapılandırmalarında, diğer düğümlerin çalışmasını devralabilen bir düğüme yerleştirilebilecek en yüksek yükü göz önünde bulundurmanız önemlidir. Böyle bir düğümün, kabul edilebilir bir performans düzeyini korumak için yeterli kapasiteye sahip olması gerekir.

HA kümelerinin kuyruk yöneticisi kümeleri ile ilişkisi

Kuyruk yöneticisi kümeleri, denetimi azaltır ve kuyruk yöneticisi küme kuyrukları eşgörünümlerinde iletilerin yük dengelemesini sağlar. Ayrıca, kuyruk yöneticisinden oluşan bir hata nedeniyle, ileti alışverişi uygulamalarının bir kuyruk yöneticisi kümesinin yürürlükte kalan eşgörünümlerine erişmeye devam edebileceğinden, tek bir kuyruk yöneticisinden daha yüksek kullanılabilirlik sunarlar. Ancak, kuyruk yöneticisi kümeleri tek başına kuyruk yöneticisi hatasının otomatik olarak saptanması ve kuyruk yöneticisi yeniden başlatma ya da hata durumunda yedek sisteme geçiş özelliğinin otomatik olarak saptanması sağlamaz. HA kümeleri bu özellikleri sağlar. İki tip küme, iyi bir etki için birlikte kullanılabilir.

WebSphere MQ istemcileri

WebSphere MQ istemcilerinin yeniden başlatma ya da devralma işlemi söz konusu olabilen bir kuyruk yöneticisiyle iletişim kuran istemcilerinin, bozuk bir bağlantıyı tolere etmek için yazılması ve yeniden sürekli yeniden bağlanma girişiminde bulunması gerekir. WebSphere MQ Sürüm 7, bağlantı kullanılabilirliği ve iş yükü dengelemesine yardımcı olan Client Channel Definition Table (CCDT) çizelgesinin işlenmesinde yer alan özellikleri kullanıma sunar; ancak bunlar, yedek sisteme geçiş sistemiyle çalışırken doğrudan ilgili değildir.

Bir WebSphere MQ MQI istemcisinin iki aşamalı hareketlere katılmasına olanak veren Extended Transactional Client (ETC) (Extended Transactional Client; her zaman aynı kuyruk yöneticisine bağlanmalıdır). ETC, bir IP yükleme dengeleyici gibi, kuyruk yöneticisi listesinden seçim yapmak için kullanamaz. Bir HA ürünü kullanırken, kuyruk yöneticisi kimliğini (ad ve adres) çalıştırdığı düğümü korur; böylece ETC, HA denetimi altındaki kuyruk yöneticileriyle kullanılabilir.

Bir HA kümesinde çalışan WebSphere MQ

Tüm HA kümeleri, hata durumunda yedek sisteme geçiş birimi kavramına sahiptir. Bu, yüksek kullanılabilirlikli hizmeti oluşturan tüm kaynakları içeren bir tanımlamalar kümesidir. Hata durumunda yedek sisteme geçiş birimi, hizmetin kendisine ve bağlı olduğu diğer tüm kaynaklara da dahildir.

HA çözümleri, bir hata durumunda yedek sisteme geçiş için farklı terimleri kullanır:

- On PowerHA for AIX the unit of failover is called a *kaynak grubu*.
- Veritas Cluster Server üzerinde *hizmet grubu* olarak bilinmektedir.
- Serviceguard 'da buna *paket* adı verilir.

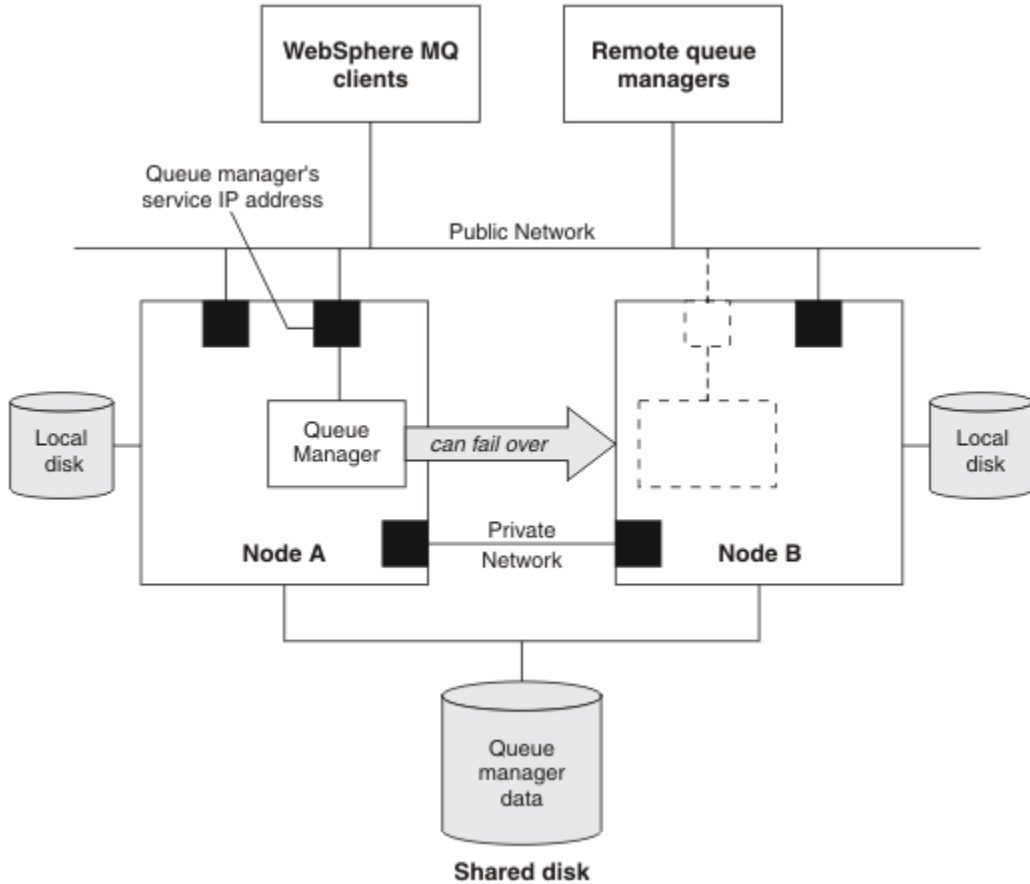
Bu konu, hata durumunda yedek sisteme geçiş birimini ifade etmek için *kaynak grubu* terimini kullanır.

WebSphere MQ için en küçük hata durumunda yedek sisteme geçiş birimi bir kuyruk yöneticidir. Tipik olarak, kuyruk yöneticisini içeren kaynak grubu, bir birim grubunda ya da disk grubunda, yalnızca kaynak grubu tarafından kullanılmak üzere ayrılmış paylaşılan diskleri ve kuyruk yöneticisine bağlanmak için

kullanılan IP adresini de içerir. Ayrıca, dinleyici ya da tetikleyici izleme programı gibi diğer WebSphere MQ kaynaklarının aynı kaynak grubunda, ayrı kaynaklar olarak ya da kuyruk yöneticisinin kendisinin denetimi altında olması da olanaklıdır.

Bir HA kümesinde kullanılacak bir kuyruk yöneticisinin verilerine sahip olması gerekir ve kümedeki düğümler arasında paylaşılan disklerde oturum açmalıdır. HA kümesi, kümedeki bir defada yalnızca bir düğümün disklere yazabilmesini sağlar. HA kümesi, kuyruk yöneticisinin durumunu izlemek için bir izleme komut dosyasını kullanabilir.

Kuyruk yöneticisiyle ilgili veriler ve günlükler için tek bir paylaşılan disk kullanmak mümkündür. Ancak, ayrı olarak boyutlandırılıp ayarlanabilmesi için ayrı paylaşılan dosya sistemlerini kullanmak normal bir uygulamadır.



Şekil 60. HA kümesi

Şekil 1, iki düğüm içeren bir HA kümesi gösterir. HA kümesi, bir kaynak grubunda tanımlı olan bir kuyruk yöneticisinin kullanılabilirliğini yönetmektedir. Bu etkin/pasif ya da soğuk bir bekleme yapılandırmasıdır; yalnızca bir düğüm, A düğümü şu anda bir kuyruk yöneticisi çalıştırıyor olabilir. Kuyruk yöneticisi, verileri ve paylaşılan bir diskteki günlük dosyalarıyla yaratıldı. Kuyruk yöneticisi, HA kümesi tarafından da yönetilen bir hizmet IP adresine sahiptir. Kuyruk yöneticisi, paylaşılan diske ve hizmet IP adresine bağlıdır. HA kümesi, kuyruk yöneticisini düğüm A 'dan düğüm B' ye kadar geçemezse, önce kuyruk yöneticisinin bağımlı kaynaklarını düğüm B ' ye taşır ve kuyruk yöneticisini başlatır.

HA kümesi birden çok kuyruk yöneticisi içeriyorsa, HA kümesi yapılandırmanız, hata durumunda yedek sisteme geçiş işleminden sonra aynı düğümde çalışan iki ya da daha çok kuyruk yöneticisine neden olabilir. HA kümesindeki her kuyruk yöneticisinin, belirli bir zamanda etkin olması için sahip olduğu küme düğümünde kullandığı kendi kapı numarasına atanmış olması gerekir.

Genel olarak, HA kümesi kök kullanıcı olarak çalışır. WebSphere MQ , mqm kullanıcısı olarak çalışır. WebSphere MQ yönetim denetimi mqm grubunun üyelerine verilir. Mqm kullanıcısı ve grubunun tüm HA kümesi düğümlerinde var olduğunu doğrulayın. Kullanıcı kimliği ve grup kimliği, küme boyunca tutarlı

olmalıdır. Kök kullanıcıya göre WebSphere MQ yönetimine izin verilmez; komut dosyalarının başlatılması, durdurulması ya da izleme komut dosyalarının mqm kullanıcılarına geçmeleri gerekir.

Not: WebSphere MQ , tüm düğümlerde doğru olarak kurulmalı; ürün yürütülebilir dosyalarını paylaşamazsınız.

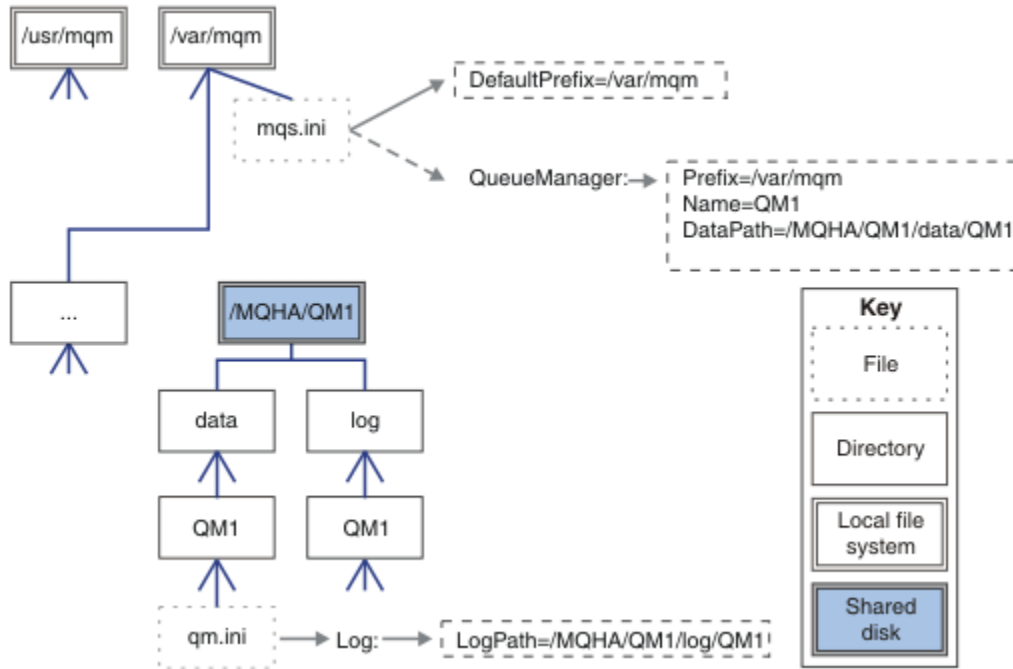
Paylaşılan disklerin yapılandırılması

HA kümesindeki bir WebSphere MQ kuyruk yöneticisi, veri dosyalarının ve günlük kütüklerinin paylaşılan bir disk üzerindeki ortak adlandırılmış uzak dosya sistemleri içinde olmasını gerektirir.

Paylaşılan diskleri yapılandırmak için aşağıdaki adımları tamamlayın:

1. Kuyruk yöneticisinin dosya sistemleri için bağlama noktalarının adlarına karar verin. Örneğin, kuyruk yöneticisinin veri dosyaları için /MQHA/qmgrname/data ve günlük dosyaları için /MQHA/qmgrname/log .
2. Kuyruk yöneticisinin veri ve günlük dosyalarını içerecek bir birim grubu (ya da disk grubu) yaratın. Bu birim grubu, kuyruk yöneticisiyle aynı kaynak grubunda bulunan yüksek kullanılabilirlik (HA) kümesi tarafından yönetilir.
3. Kuyruk yöneticisinin verileri ve birim grubu içindeki günlük dosyaları için dosya sistemlerini oluşturun.
4. Sırayla her bir düğüm için, dosya sistemlerine ilişkin bağlama noktalarını oluşturun ve dosya sistemlerinin bağlanabildiğinden emin olun. mqm kullanıcısının bağlama noktalarının sahibi olması gerekir.

Şekil 1 , HA kümesinde kuyruk yöneticisi için olası bir yerleşim düzenini gösterir. Kuyruk yöneticisinin verileri ve günlük dizinleri, her ikisi de /MQHA/QM1' ta bağlanan paylaşılan diskte yer alıyor. Hata durumunda yedek sisteme geçiş işlemi gerçekleştiğinde, bu disk HA kümesinin düğümleri arasında geçiş yapılır. Böylece, kuyruk yöneticisi yeniden başlatıldığında veriler kullanılabilir olur. mqm . ini dosyasında QM1 kuyruk yöneticisi için bir stanza vardır. qm . ini dosyasındaki Log stanza, LogPath için bir değere sahiptir.



Şekil 61. Paylaşılan adlandırılmış data ve log dizinleri

Yüksek kullanılabilirlik (HA) kümesinde kullanılmak üzere kuyruk yöneticisi yaratılması

Yüksek kullanılabilirlikli bir kümede kuyruk yöneticisi kullanmaya yönelik ilk adım, düğümlerin birinde kuyruk yöneticisi yaratılmasına neden olur.

Bir HA kümesinde kullanılmak üzere bir kuyruk yöneticisi yaratmak için, üzerinde kuyruk yöneticisinin yaratılacağı düğümlerden birini seçin. Bu düğümden aşağıdaki adımları tamamlayın:

1. Düğüm üzerindeki kuyruk yöneticisinin kütük sistemlerini sisteme bağlayın.

2. Create the queue manager by using the **crtmqm** command. Örneğin:

```
crtmqm -md /MQHA/qmgrname/data -ld /MQHA/qmgrname/log qmgrname
```

3. Start the queue manager manually by using the **strmqm** command.

4. Kuyruk yöneticisi başlatıldığında kuyruklar ve kanallar yaratılması ve kuyruk yöneticisinin otomatik olarak başlatılması için kuyruk yöneticisinin otomatik olarak başlatılacağı gibi, kuyruk yöneticisinin ilk yapılandırmasını tamamlayın.

5. **endmqm** komutunu kullanarak kuyruk yöneticisini durdurun.

6. “Yüksek kullanılabilirlikli (HA) kümedeki diğer düğümlere kuyruk yöneticisi yapılış bilgilerinin eklenmesi” sayfa 315’inde belgelenmiş olan daha sonraki bir görevde kullanabileceğiniz **addmqinf** komutunu görüntülemek için **dspmqinf** komutunu kullanın:

```
dspmqinf -o command qmgrname
```

Burada qmgrname , kuyruk yöneticisinin adıdır.

7. Görüntülenen **addmqinf** komutu aşağıdaki örneğe benzer olacaktır:

```
addmqinf -sQueueManager -vName=qmgrname -vDirectory=qmgrname \  
-vPrefix=/var/mqm -vDataPath=/MQHA/qmgrname/data/qmgrname
```

Görüntülenen komutta dikkatli bir not alın.

8. Kuyruk yöneticisinin dosya sistemlerini kaldırın.

Şimdi, “Yüksek kullanılabilirlikli (HA) kümedeki diğer düğümlere kuyruk yöneticisi yapılış bilgilerinin eklenmesi” sayfa 315’inde açıklanan adımları tamamlamaya hazırsınız.

Yüksek kullanılabilirlikli (HA) kümedeki diğer düğümlere kuyruk yöneticisi yapılış bilgilerinin eklenmesi

Kuyruk yöneticisi yapılışını HA kümesindeki diğer düğümlere eklemelisiniz.

Before you complete this task, you must have completed the steps in “Yüksek kullanılabilirlik (HA) kümesinde kullanılmak üzere kuyruk yöneticisi yaratılması” sayfa 314.

Kuyruk yöneticisine ilişkin yapılandırma bilgilerini HA kümesindeki diğer düğümlerin her birine eklemek için, diğer düğümlerin her birine ilişkin aşağıdaki adımları tamamlayın:

1. Kuyruk yöneticisi kütük sistemlerini sisteme bağlayın.

2. Add the queue manager configuration information to the node, either by editing `/var/mqm/mqs.ini` directly, or by issuing the **addmqinf** command that was displayed by the **dspmqinf** command in steps 6 and 7 in “Yüksek kullanılabilirlik (HA) kümesinde kullanılmak üzere kuyruk yöneticisi yaratılması” sayfa 314.

3. Yapılandırmayı doğrulamak için kuyruk yöneticisini başlatın ve durdurun.

Kuyruk yöneticisini başlatmak ve durdurmak için kullanılan komutlar, **addmqinf** komutuyla aynı IBM WebSphere MQ kurulumundan yayınlanmalıdır. Kuyruk yöneticisini farklı bir kurulumdan başlatmak ve durdurmak için, önce **setmqm** komutunu kullanarak kuyruk yöneticisiyle ilişkili kurulumu ayarlamanız gerekir. Ek bilgi için bkz. [setmqm](#).

4. Kuyruk yöneticisi kütük sistemlerini kaldırın.

Yüksek kullanılabilirlik (HA) kümesinin denetimi altında kuyruk yöneticisi başlatma

Kuyruk yöneticisi bir kaynak olarak HA kümesinde temsil edilir. HA kümesi, kuyruk yöneticisini başlatabilecek ve durdurabilmelidir. Çoğu durumda, kuyruk yöneticisini başlatmak için bir kabuk komut dosyası kullanabilirsiniz. Bu komut dosyalarını, bir ağ dosya sistemi kullanarak ya da yerel disklerin her birine kopyalayarak, kümedeki tüm düğümlerde aynı yerde kullanılabilir duruma getirmeniz gerekir.

Not: Başarısız bir kuyruk yöneticisini yeniden başlatmadan önce, kuyruk yöneticisinin bu yönetim ortamından uygulamalarının bağlantısını kesmeniz gerekir. Bunu yapmazsanız, kuyruk yöneticisi doğru olarak yeniden başlatılamayabilir.

Burada uygun kabuk komut dosyalarına örnekler verilir. Bunları gereksinimlerinize göre uyarlayabilir ve HA kümenizi denetleyebilmek için kuyruk yöneticisini başlatmak için kullanabilirsiniz.

Aşağıdaki kabuk komut dosyası, kuyruk yöneticisinin başarıyla başlatılabilmesi için, HA kümesi kullanıcısından mqm kullanıcısına nasıl geçiş yapacağına ilişkin bir örnektir:

```
#!/bin/ksh
# A simple wrapper script to switch to the mqm user.
su mqm -c name_of_your_script $*
```

Aşağıdaki kabuk komut dosyası, kuyruk yöneticisinin yürürlükteki durumuyla ilgili herhangi bir varsayım yapmadan, kuyruk yöneticisinin nasıl başlatılacağına ilişkin bir örnektir. Kuyruk yöneticisine ait tüm işlemleri sonlandırmak için son derece ani bir yöntem kullandığına dikkat edin:

```
#!/bin/ksh
#
# This script robustly starts the queue manager.
#
# The script must be run by the mqm user.

# The only argument is the queue manager name. Save it as QM variable
QM=$1

if [ -z "$QM" ]
then
    echo "ERROR! No queue manager name supplied"
    exit 1
fi

# End any queue manager processes which might be running.

srchstr="(|-m)$QM *.*$"
for process in amqzmuc0 amqzma0 amqfcxba amqfcpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqlsr amqcrsta amqirmfa amqrmppa \
               amqzfuma amqzmaa amqzmuf0 amqzmur0 amqzmgr0
do
    ps -ef | tr "\t" " " | grep $process | grep -v grep | \
    egrep "$srchstr" | awk '{print $2}' | \
    xargs kill -9 > /dev/null 2>&1
done

# It is now safe to start the queue manager.
# The stmqm command does not use the -x flag.
stmqm ${QM}
```

Diğer ilgili programları başlatmak için komut dosyasını değiştirebilirsiniz.

Yüksek kullanılabilirlik (HA) kümesi denetimi altında kuyruk yöneticisinin durdurulması

Çoğu durumda, bir kuyruk yöneticisini durdurmak için bir kabuk komut dosyası kullanabilirsiniz. Burada uygun kabuk komut dosyalarına örnekler verilir. Bunları gereksinimlerinize göre uyarlayabilir ve HA kümenizi kontrol altında kuyruk yöneticisini durdurmak için kullanabilirsiniz.

Aşağıdaki komut dosyası, kuyruk yöneticisinin yürürlükteki durumu hakkında varsayımlar gerçekleştirmeden hemen durdurulacağı bir örnektir. Komut dosyası mqm kullanıcısı tarafından çalıştırılmalıdır; bu nedenle, kullanıcıyı HA kümesi kullanıcısından mqm 'ye (örneğin, “Yüksek kullanılabilirlik (HA) kümesinin denetimi altında kuyruk yöneticisi başlatma” sayfa 315’inde bir örnek kabuk komut dosyası sağlanır) değiştirmek için bu komut dosyasının bir kabuk komut dosyasına sarılması gerekli olabilir:

```
#!/bin/ksh
#
# The script ends the QM by using two phases, initially trying an immediate
# end with a time-out and escalating to a forced stop of remaining
```

```

# processes.
#
# The script must be run by the mqm user.
#
# There are two arguments: the queue manager name and a timeout value.
QM=$1
TIMEOUT=$2

if [ -z "$QM" ]
then
    echo "ERROR! No queue manager name supplied"
    exit 1
fi

if [ -z "$TIMEOUT" ]
then
    echo "ERROR! No timeout specified"
    exit 1
fi

for severity in immediate brutal
do
    # End the queue manager in the background to avoid
    # it blocking indefinitely. Run the TIMEOUT timer
    # at the same time to interrupt the attempt, and try a
    # more forceful version. If the brutal version fails,
    # nothing more can be done here.

    echo "Attempting ${severity} end of queue manager '${QM}'"
    case $severity in

    immediate)
        # Minimum severity of endmqm is immediate which severs connections.
        # HA cluster should not be delayed by clients
        endmqm -i ${QM} &
        ;;

    brutal)
        # This is a forced means of stopping queue manager processes.

        srchstr="( |-m)$QM *.*$"
        for process in amqzmuc0 amqzma0 amqfcxba amqfcpub amqpcsea amqzlaa0 \
            amqzlsa0 runmqchi runmqslr amqcrista amqrrmfa amqrmppa \
            amqzfuma amqzdmaa amqzmuf0 amqzmur0 amqzmgr0
        do
            ps -ef | tr "\t" " " | grep $process | grep -v grep | \
                egrep "$srchstr" | awk '{print $2}' | \
                    xargs kill -9 > /dev/null 2>&1
        done

        esac

        TIMED_OUT=yes
        SECONDS=0
        while (( $SECONDS < $TIMEOUT ))
        do
            TIMED_OUT=yes
            i=0
            while [ $i -lt 5 ]
            do
                # Check for execution controller termination
                srchstr="( |-m)$QM *.*$"
                cnt=`ps -ef | tr "\t" " " | grep amqzma0 | grep -v grep | \
                    egrep "$srchstr" | awk '{print $2}' | wc -l`
                i=`expr $i + 1`
                sleep 1
                if [ $cnt -eq 0 ]
                then
                    TIMED_OUT=no
                    break
                fi
            done

            if [ ${TIMED_OUT} = "no" ]
            then
                break
            fi

            echo "Waiting for ${severity} end of queue manager '${QM}'"
            sleep 1
        done # timeout loop
    
```

```
if [ ${TIMED_OUT} = "yes" ]
then
  continue          # to next level of urgency
else
  break             # queue manager is ended, job is done
fi

done # next phase
```

Kuyruk yöneticisinin izlenmesi

Kuyruk yöneticisinin durumunu düzenli olarak izlemek için yüksek kullanılabilirlik (HA) kümesinin bir yolunu sağlamak olağan bir yöntemdir. Çoğu durumda, bunun için bir kabuk komut dosyası kullanabilirsiniz. Burada uygun kabuk komut dosyalarına örnekler verilir. Bu komut dosyalarını gereksinimlerinize göre uyarlayabilir ve ortamınıza özgü ek izleme denetimleri yapmak için bunları kullanabilirsiniz.

WebSphere MQ sürüm 7.1' den, bir sistemde birden çok WebSphere MQ ortak kuruluşu olması mümkündür. Birden çok kurulumla ilişkin ek bilgi için [Birden çok kuruluşbaşıllıklı konuya bakın](#). If you intend to use the monitoring script across multiple installations, including installations at version 7.1, or higher, you might need to perform some additional steps. Birincil kuruluşunuz varsa ya da komut dosyasını 7.1 sürümünden önceki sürümlerle kullanıyorsanız, komut dosyasını kullanmak için `MQ_INSTALLATION_PATH` belirtmenize gerek yoktur. Ters durumda, aşağıdaki adımlar `MQ_INSTALLATION_PATH` olanağının doğru olarak tanımlandığından emin olun:

1. Bir kuyruk yöneticisi için doğru `MQ_INSTALLATION_PATH` 'yi tanımlamak için sürüm 7.1 kuruluşundan **crtmqenv** komutunu kullanın:

```
crtmqenv -m qmname
```

Bu komut, `qmname` ile belirtilen kuyruk yöneticisi için doğru `MQ_INSTALLATION_PATH` değerini döndürür.

2. Uygun `qmname` ve `MQ_INSTALLATION_PATH` değıştirmeleriyle izleme komut dosyasını çalıştırın.

Not: AIX için PowerHA , kuyruk yöneticisine ilişkin izleme programına bir parametre sağlamanın bir yolunu sağlamaz. Kuyruk yöneticisi adını sarmalayan her kuyruk yöneticisi için ayrı bir izleme programı yaratmalısınız. Aşağıda, kuyruk yöneticisi adını kapsüllemek için AIX üzerinde kullanılan bir komut dosyası örneğı gösterilmektedir:

```
#!/bin/ksh
su mqm -c name_of_monitoring_script qmname MQ_INSTALLATION_PATH
```

where `MQ_INSTALLATION_PATH` is an optional parameter that specifies the path to the installation of IBM WebSphere MQ that the queue manager `qmname` is associated with.

Aşağıdaki komut dosyası, **runmqsc** ' in askıda kalması olasılığına karşı sağlam değildir. Genellikle HA kümeleri, asılı bir izleme komut dosyasını bir hata olarak kabul eder ve bu olasılığa kendileri de güçlü davranır.

Ancak komut dosyası, kuyruk yöneticisinin başlangıç durumunda olmasına göz yummaktadır. Bunun nedeni, HA kümesinin kuyruk yöneticisini başlatma işlemini başlatmış olduğu anda izlemeyi başlatması yaygın bir durum olduğundan kaynaklanır. Bazı HA kümeleri, kaynaklar için bir başlangıç aşamasını ve çalışan bir aşamayı ayırt eder, ancak başlangıç aşamanın süresini yapılandırmak için gereklidir. Kuyruk yöneticisi başlatmak için gereken süre, yapması gereken iş miktarına bağlı olduğundan, kuyruk yöneticisini başlatma süresi üst sınırını seçmek çok zor olur. Çok düşük bir değer seçerseniz, HA kümesi yanlış bir şekilde, başlatma işlemi tamamlanmadığında kuyruk yöneticisinin başarısız olduğunu varsayar. Bu sonsuz bir sadakatle sonuçlanabilir.

Bu komut dosyasının mqm kullanıcısı tarafından çalıştırılması gerekir; bu nedenle, kullanıcıyı HA kümesi kullanıcısından mqm 'ye (örneğin, "Yüksek kullanılabilirlik (HA) kümesinin denetimi altında kuyruk yöneticisi başlatma" sayfa 315 içinde bir örnek kabuk komut dosyası sağlanmıştır) değıştirmek için bu komut dosyasının bir kabuk komut dosyasına sarılması gerekli olabilir:

```
#!/bin/ksh
#
```

```

# This script tests the operation of the queue manager.
#
# An exit code is generated by the runmqsc command:
# 0 => Either the queue manager is starting or the queue manager is running and responds.
#     Either is OK.
# >0 => The queue manager is not responding and not starting.
#
# This script must be run by the mqm user.
QM=$1
MQ_INSTALLATION_PATH=$2

if [ -z "$QM" ]
then
    echo "ERROR! No queue manager name supplied"
    exit 1
fi

if [ -z "$MQ_INSTALLATION_PATH" ]
then
    # No path specified, assume system primary install or MQ level < 7.1.0.0
    echo "INFO: Using shell default value for MQ_INSTALLATION_PATH"
else
    echo "INFO: Prefixing shell PATH variable with $MQ_INSTALLATION_PATH/bin"
    PATH=$MQ_INSTALLATION_PATH/bin:$PATH
fi

# Test the operation of the queue manager. Result is 0 on success, non-zero on error.
echo "ping qmgr" | runmqsc ${QM} > /dev/null 2>&1
pingresult=$?

if [ $pingresult -eq 0 ]
then # ping succeeded

    echo "Queue manager '${QM}' is responsive"
    result=0

else # ping failed

    # Don't condemn the queue manager immediately, it might be starting.
    srchstr="(|-m)$QM *.*$"
    cnt=`ps -ef | tr "\t" " " | grep strmqm | grep "$srchstr" | grep -v grep \
        | awk '{print $2}' | wc -l`
    if [ $cnt -gt 0 ]
    then
        # It appears that the queue manager is still starting up, tolerate
        echo "Queue manager '${QM}' is starting"
        result=0
    else
        # There is no sign of the queue manager starting
        echo "Queue manager '${QM}' is not responsive"
        result=$pingresult
    fi

fi

exit $result

```

Kuyruk yöneticisini yüksek kullanılabilirlik (HA) kümesinin denetimi altına koyma

Kuyruk yöneticisini, kuyruk yöneticisinin IP adresi ve paylaşılan diskleriyle birlikte HA kümesinin denetimi altında yapılandırmalısınız.

Bir kaynak grubunu kuyruk yöneticisini ve ilişkili tüm kaynaklarını içerecek şekilde tanımlamak için aşağıdaki adımları izleyin:

1. Kuyruk yöneticisini, kuyruk yöneticisinin birim ya da disk grubunu ve kuyruk yöneticisinin IP adresini içeren kaynak grubunu yaratın. IP adresi, bilgisayarın IP adresi değil, sanal bir IP adresidir.
2. HA kümesinin, küme düğümleri arasındaki kaynakları doğru şekilde değiştirdiğini ve kuyruk yöneticisini denetmeye hazır olduğunu doğrulayın.

Yüksek kullanılabilirlikli (HA) küme düğümünden kuyruk yöneticisinin silinmesi

Kuyruk yöneticisini artık kuyruk yöneticisini çalıştırmak için gerekmeyecek bir düğümden kaldırmak isteyebilirsiniz.

Bir HA kümesindeki bir düğümden kuyruk yöneticisini kaldırmak için aşağıdaki adımları tamamlayın:

1. HA kümesinden düğümü kaldırın; böylece, HA kümesi bu düğümdeki kuyruk yöneticisini etkinleştirme girişiminde bulunmayacak.
2. Kuyruk yöneticisinin yapılanış bilgilerini kaldırmak için aşağıdaki **rmvmqinf** komutunu kullanın:

`rmvmqinf qmgrname`

Kuyruk yöneticisini tam olarak silmek için **dlmqm** komutunu kullanın. Ancak, bu işin kuyruk yöneticisinin veri ve günlük kütüklerini tümüyle sildiğini unutmayın. Kuyruk yöneticisini sildiğiniz zaman, diğer düğümlerden kalan yapılandırma bilgilerini kaldırmak için **rmvmqinf** komutunu kullanabilirsiniz.

Microsoft Cluster Service (MSCS) olanağını destekleme

Sanal sunucuların hata durumunda yedek sisteme geçiş özelliğini desteklemek için MSCS ' nin tanıtımı ve ayarlanması.

This information applies to WebSphere MQ for Pencereler only.

Microsoft Cluster Service (MSCS), sunucuları bir *küme*ye bağlamanızı, daha yüksek veri ve uygulama kullanılabilirliği vermenizi ve sistemin yönetilmesini kolaylaştırmanızı sağlar. MSCS, sunucu ya da uygulama hatalarını otomatik olarak algılayabilir ve bu hataları kurtarabilir.

MSCS, uygulamalara, Web sitelerine, yazdırma kuyruklarına ya da dosya paylaşımlarına (örneğin, disk yazım işlemleri, dosyalar ve IP adresleri gibi) karşılık gelen *sanal sunucular*' ın *yedek sisteme geçiş* özelliğini destekler.

Hata Durumunda Yedek Sisteme Geçiş , MSCS ' nin kümedeki bir bilgisayarda bir uygulamada hata saptadığı ve bozulan uygulamayı düzenli bir şekilde sona erdirdiği, durum verilerini diğer bilgisayara aktardığı ve uygulamanın yeniden başlatıldığı işlemidir.

Bu bölümde MSCS kümeleri tanıtılır ve aşağıdaki bölümlerde MSCS desteğini ayarlama açıklanır:

- [“MSCS kümelerini tanıma” sayfa 320](#)
- [“MSCS kümelemesi için IBM WebSphere MQ ' nin ayarlanması” sayfa 321](#)

Daha sonra, MSCS kümelemesi için WebSphere MQ ' un aşağıdaki kısımlarda nasıl yapılandırılacağı hakkında bilgi verir:

- [“MSCS ile kullanılmak üzere kuyruk yöneticisi yaratılması” sayfa 323](#)
- [“Kuyruk Yöneticisinin MSCS Depolaması için taşınması” sayfa 324](#)
- [“Kuyruk yöneticisini MSCS denetimin altına koyma” sayfa 325](#)
- [“Kuyruk yöneticisinin MSCS denetiminden kaldırılması” sayfa 332](#)

Daha sonra, MSCS ' yi WebSphere MQ ile birlikte kullanmaya ilişkin yararlı ipuçları verir ve aşağıdaki bölümlerde WebSphere MQ MSCS destek yardımcı programları hakkında ayrıntılı bilgiler verir:

- [“MSCS kullanımıyla ilgili ipuçları ve öneriler” sayfa 333](#)
- [“IBM WebSphere MQ MSCS desteği yardımcı programları” sayfa 336](#)

MSCS kümelerini tanıma

MSCS kümeleri iki ya da daha çok bilgisayarın gruplarıdır, birbirine bağlanır ve bu şekilde yapılandırılır; bir hata oluşursa, MSCS bir *hata durumunda yedek sisteme geçiş* gerçekleştirir, uygulamaların durum verilerini kümedeki başka bir bilgisayara kümedeki başka bir bilgisayara aktarmakta ve bu işlem burada yeniden başlatılabilmektedir.

[“Yüksek kullanılabilirlik yapılandırmalarıyla WebSphere MQ ' un kullanılması” sayfa 310](#) , MSCS kümeleri, çok eşgönlü kuyruk yöneticileri ve WebSphere MQ kümeleri arasında bir karşılaştırma içerir.

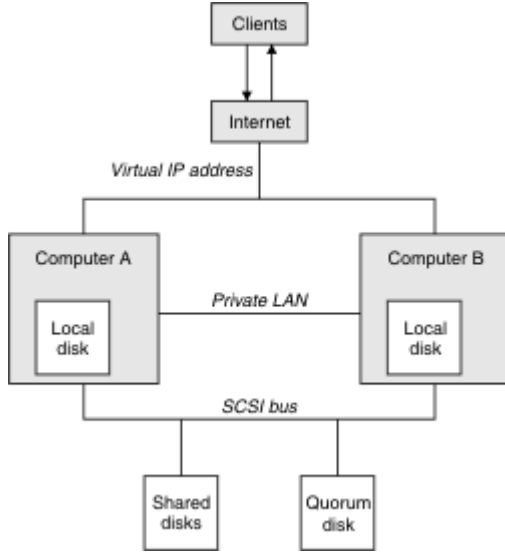
Bu bölümde ve alt konularında, *küme* terimi kendi başına kullanıldığında, **her zaman** bir MSCS kümesi anlamına gelir. Bu, bu kılavuzda başka bir yerde tanımlanan bir WebSphere MQ kümesinden farklıdır.

A two-machine cluster comprises two computers (for example, A and B) that are jointly connected to a network for client access using a *sanal IP adresi*. Ayrıca, bir ya da daha çok özel ağ tarafından birbirleriyle de bağlantı kurulabilirler. A ve B, her bir kullanım için sunucu uygulamaları için en az bir diskte paylaşım

sağlar. Ayrıca, MSCS 'nin dışlayıcı kullanımı için yedek bağımsız disk dizisi (RAID) Düzey 1 olması gereken başka bir paylaşılan disk de vardır; bu, *yetersayı* diski olarak bilinir. MSCS her iki bilgisayarı da, donanımın ve yazılımların doğru biçimde çalıştığını denetler.

Bunun gibi basit bir kurulumda, her iki bilgisayarın da üzerinde kurulu tüm uygulamaları vardır, ancak yalnızca A bilgisayarı canlı uygulamalar ile çalışır; B bilgisayarı yalnızca çalışır ve bekler. Bilgisayar A, herhangi bir sorun aralığı ile karşılaşır, MSCS, bozulan uygulamayı düzenli bir şekilde sona erdirir, durum verilerini diğer bilgisayara aktarır ve uygulamayı yeniden başlatır. Bu, *yedek sisteme geçiş* olarak bilinir. Uygulamalar, MSCS ve hata durumunda yedek sisteme geçiş ile tam olarak etkileşimde bulunmaları için *küme farkındalar* yapılabilir.

İki bilgisayarlı bir küme için tipik bir ayar, Şekil 62 sayfa 321 içinde gösterildiği gibidir.



Şekil 62. İki bilgisayarlı MSCS kümesi

Her bilgisayar, MSCS denetimi altında, aynı anda yalnızca biri olan paylaşılan diske erişebilirler. Hata durumunda yedek sisteme geçiş durumunda MSCS, diğer bilgisayara erişimi değiştirir. Paylaşılan diskin kendisi genellikle bir RAID 'dir, ancak gerekmez.

Her bilgisayar, istemci erişimi için dış ağa bağlıdır ve her birinin bir IP adresi vardır. Ancak, bu kümeyle iletişim kuran bir dış istemci, yalnızca bir *sanal IP adresi* nin farkındadır ve MSCS, küme içindeki IP trafiğini uygun şekilde yönlendirir.

MSCS, bir ya da daha fazla özel bağlantı ya da genel ağ üzerinden (örneğin, sağlıklı işletim bildirimi) durumlarını izlemek ve veritabanlarını uyumlulaştırmak için, iki bilgisayar arasında kendi iletişimini de gerçekleştirir.

MSCS kümelemesi için IBM WebSphere MQ 'nin ayarlanması

Kuyruk yöneticisini, MSCS 'ye yedek sisteme geçiş birimi yaparak kümeleme için IBM WebSphere MQ 'u yapılandırabilirsiniz. Bir kuyruk yöneticisini kaynak olarak MSCS 'ye tanımlayıp, daha sonra bunu izleyebilir ve bir sorun varsa, kümedeki başka bir bilgisayara aktarabilirsiniz.

Sisteminizi bu işlem için ayarlamak üzere kümedeki her bir bilgisayara IBM WebSphere MQ 'i kurarak başlamanız gerekir.

Kuyruk yöneticisi IBM WebSphere MQ kuruluş adıyla ilişkilendirildiği için, kümedeki tüm bilgisayarlardaki IBM WebSphere MQ kuruluş adı aynı olmalıdır. Bkz. [Kurma ve Kaldırma](#).

Kuyruk yöneticilerinin yalnızca, sizin yarattığınız bilgisayarda var olması gerekir. Hata durumunda yedek sisteme geçiş durumunda, MSCS diğer bilgisayardaki kuyruk yöneticilerini başlatır. Ancak kuyruk yöneticileri, yerel bir sürücüde değil, bir küme paylaşılan diskinde günlük ve veri dosyalarına sahip olmalıdır. Yerel bir sürücüde önceden kurulmuş bir kuyruk yöneticiniz varsa, bunu IBM WebSphere MQ ile verilen bir aracı kullanarak geçirebilirsiniz; bkz. ["Kuyruk Yöneticisinin MSCS Depolaması için taşınması"](#)

sayfa 324. MSCS ile kullanmak üzere yeni kuyruk yöneticileri yaratmak istiyorsanız, bkz. [“MSCS ile kullanılmak üzere kuyruk yöneticisi yaratılması” sayfa 323.](#)

Kuruluştan ve geçişten sonra, MSCS ' yi kuyruk yöneticilerinizden haberdar etmek için MSCS Cluster Administrator olanağını kullanın; bkz. [“Kuyruk yöneticisini MSCS denetimin altına koyma” sayfa 325.](#)

Bir kuyruk yöneticisini MSCS denetiminden kaldırmaya karar verirsiniz, [“Kuyruk yöneticisinin MSCS denetiminden kaldırılması” sayfa 332](#) içinde açıklanan yordamı kullanın.

Simetri ayarı

Bir uygulama bir düğümden diğerine geçtiğinde, düğüm ne olursa olsun aynı şekilde hareket etmelidir. Bunu sağlamanın en iyi yolu, ortamları aynı hale getirmemektir.

Bunu yapabiliyorsa, her bilgisayarda aynı donanım, işletim sistemi yazılımı, ürün yazılımı ve yapılandırma ile bir küme ayarlayabilirsiniz. In particular, ensure that all the required software installed on the two computers is identical in terms of version, maintenance level, SupportPacs, paths and exits, and that there is a common namespace (security environment) as described in [“MSCS güvenliği” sayfa 322.](#)

MSCS güvenliği

Başarılı MSCS güvenliği için bu yönergeleri izleyin.

Yönergeler aşağıdaki gibidir:

- Kümedeki her bir bilgisayarda aynı yazılım kuruluşlarına sahip olduğundan emin olun.
- Küme genelinde ortak bir ad alanı (güvenlik ortamı) yaratın.
- Make the nodes of the MSCS cluster members of a domain, within which the user account that is the *küme sahibi* is a domain account.
- Diğer kullanıcı hesaplarını da, her iki düğümde de kullanılabilir duruma getirmek için, küme üzerindeki diğer kullanıcı hesaplarını da etki alanı hesaplarından yapın. Zaten bir etki alanınız varsa ve WebSphere MQ ile ilgili hesaplar etki alanı hesaplarıyla otomatik olarak bu durumda olur. Şu anda bir etki alanınız yoksa, küme düğümlerine ve ilgili hesaplara yönelik olarak bir *mini etki alanı* ayarlamayı düşünebilirsiniz. Amacınız, iki bilgisayarın küğünüzü tek bir bilgi işlem kaynağı gibi görünmesini sağlamak.

Bir bilgisayar için yerel olan bir hesabın diğerinde var olmadığını unutmayın. Diğer bilgisayarda aynı adı taşıyan bir hesap oluşturursanız bile, onun güvenlik tanıtıcısı (SID) farklıdır; bu nedenle, uygulamanız başka bir düğümde taşındığında, ilgili düğümde izinler yok.

Hata durumunda yedek sisteme geçiş ya da taşıma işlemi sırasında, WebSphere MQ MSCS desteği, kuyruk yöneticisi nesnelerini içeren tüm dosyaların hedef düğümde eşdeğer izinlere sahip olmasını sağlar. Açık olarak, kod denetimcileri ve mqm gruplarının ve SYSTEM hesabının tam denetime sahip olduğunu ve Everyone ' in eski düğümde okuma erişimi varsa, hedef düğümde bu iznin eklendiğini denetler.

WebSphere MQ Hizmetinizi çalıştırmak için bir etki alanı hesabı kullanabilirsiniz. Bunun, kümedeki her bir bilgisayardaki yerel mqm grubunda var olduğundan emin olun.

MSCS ile birden çok kuyruk yöneticisinin kullanılması

Bir bilgisayarda birden çok kuyruk yöneticisi çalıştırıyorsanız, bu ayarlardan birini seçebilirsiniz.

Kurulumlar aşağıdaki gibidir:

- Tek bir gruptaki tüm kuyruk yöneticileri. Bu yapılandırmada, herhangi bir kuyruk yöneticisiyle ilgili bir sorun oluşursa, gruptaki tüm kuyruk yöneticileri bir grup olarak diğer bilgisayara geçiş durumunda yedek sisteme geçiş sağlar.
- Her grupta tek bir kuyruk yöneticisi. Bu yapılandırmada, kuyruk yöneticisiyle ilgili bir sorun oluşursa, diğer kuyruk yöneticilerini etkilemeden tek başına diğer bilgisayar üzerinde başarısız olur.
- İlk iki kurulumla ait bir karışım.

Küme kipleri

There are two modes in which you might run a cluster system with WebSphere MQ: Active/Passive or Active/Active.

Not: MSCS 'yi Microsoft Transaction Server (COM+) ile birlikte kullanıyorsanız, Etkin/Etkin kipi kullanamazsınız.

Etkin/Edilgen kipi

Etkin/Edilgen kipinde, A bilgisayarı üzerinde çalışan uygulama bulunur ve B bilgisayar yedektir, yalnızca MSCS bir sorun saptadığında kullanılır.

You can use this mode with only one shared disk, but, if any application causes a failover, **Tümü** the applications must be transferred as a group (because only one computer can access the shared disk at a time).

MSCS 'yi A' yı *tercih edilen* bilgisayarlı olarak yapılandırabilirsiniz. Daha sonra, A bilgisayarı onarıldığında ya da değiştirildikten sonra yeniden düzgün bir şekilde çalışıldığında, MSCS bunu algılar ve uygulamayı otomatik olarak A bilgisayarına geçirir.

Birden fazla kuyruk yöneticisi çalıştırırsanız, her biri için ayrı bir paylaşılan disk olduğunu düşünün. Daha sonra, her kuyruk yöneticisini MSCS 'de ayrı bir gruba yerleştirin. Bu şekilde, herhangi bir kuyruk yöneticisi diğer kuyruk yöneticilerini etkilemeden diğer bilgisayara hata durumunda yedek sisteme geçiş yapabilir.

Etkin/Etkin kip

Etkin/Etkin kipte, A ve B bilgisayarlarında çalışan uygulamalar vardır ve her bilgisayardaki gruplar, diğer bilgisayarı yedek olarak kullanacak şekilde ayarlanırlar. Bilgisayar A 'da bir hata saptanırsa, MSCS durum verilerini B bilgisayarına aktarır ve uygulamayı yeniden başlatır. Bilgisayar B, kendi uygulamasını ve A 'yı çalıştırır.

Bu kurulum için en az iki paylaşılan diske gereksinim duyarsınız. MSCS 'yi A uygulamaları için tercih edilen bilgisayar olarak ve B uygulamaları için tercih edilen bilgisayar olarak B' yi yapılandırabilirsiniz. Hata durumunda yedek sisteme geçiş ve onarım işleminden sonra, her uygulama otomatik olarak kendi bilgisayarına geri döner.

WebSphere MQ için bu, örneğin, A ve B 'nin her birinde birer tane olmak üzere iki kuyruk yöneticisini kendi bilgisayarının tam gücünü kullanarak çalıştırabilirsiniz anlamına gelir. Bilgisayar A 'daki bir hatadan sonra, her iki kuyruk yöneticisi B bilgisayarında çalışır. Bu, bir bilgisayarın gücünü paylaşarak, yüksek miktardaki verileri hızla işleme yeteneğine sahip olacak şekilde paylaşmayı anlamına gelir. Ancak, hatalı A 'daki hatayı bulup onardığınızda, kritik öneme sahip uygulamalarınız yine de kullanılabilir olacaktır.

MSCS ile kullanılmak üzere kuyruk yöneticisi yaratılması

Bu yordam, MSCS denetimi altına hazırlama ve yerleştirme için uygun bir şekilde yeni bir kuyruk yöneticisinin yaratılmasına olanak sağlar.

Kuyruk yöneticisini yerel bir sürücüdeki tüm kaynaklarıyla yaratıp, günlük dosyalarını ve veri dosyalarını paylaşılan bir diske geçirmenizi sağlar. (Bu işlemi tersine çevirebilirsiniz.) **değil**, paylaşılan bir sürücüdeki kaynakları ile bir kuyruk yöneticisi yaratma girişiminde bulunmayı dener.

MSCS ile bir komut isteminden ya da WebSphere MQ Explorer 'da iki şekilde kullanmak üzere bir kuyruk yöneticisi yaratabilirsiniz. Bir komut istemi kullanmanın yararı, kuyruk yöneticisinin yaratıldığını *durdurulmuş* ve MSCS için hazır olan *el ile başlatma* olarak ayarlanır. (IBM WebSphere MQ Gezgini otomatik olarak yeni bir kuyruk yöneticisini başlatır ve yaratıldıktan sonra otomatik başlatmaya ayarlar. Bunu değiştirmeniz lazımdır.)

Komut isteminden kuyruk yöneticisi yaratılması

MSCS ile kullanmak üzere bir komut isteminden kuyruk yöneticisi yaratmak için aşağıdaki adımları izleyin:

1. Ensure that you have the environment variable MQSPREFIX set to refer to a local drive, for example C:\WebSphere MQ. Bunu değiştirirseniz, sistem hesabı değişikliği seçmesi için makineyi yeniden başlatın. Değişkeni ayarlamadıysanız, kuyruk yöneticisi kuyruk yöneticilerine ilişkin WebSphere MQ varsayılan dizininde yaratılır.

2. Create the queue manager using the **crtmqm** command. Örneğin, varsayılan dizinde `mcs_test` adlı bir kuyruk yöneticisi yaratmak için şunu kullanın:

```
crtmqm mcs_test
```

3. [“Kuyruk Yöneticisinin MSCS Depolaması için taşınması” sayfa 324' a geçin.](#)

WebSphere MQ Gezgini 'ni kullanarak kuyruk yöneticisi yaratılması

MSCS ile kullanmak üzere IBM WebSphere MQ Gezgini 'ni kullanarak kuyruk yöneticisi yaratmak için aşağıdaki adımları izleyin:

1. Start (Başlat) menüsünden IBM WebSphere MQ Explorer (Windows Gezgini) aracını başlatın.
2. Navigator Görünümü 'nde, Queue Managers ağaç düğümünü bulmak için ağaç düğümlerini genişletin.
3. Queue Managers ağaç düğümünü sağ tıklayın ve New->Queue Manager seçeneğini belirleyin. Create Queue Manager (Kuyruk Yöneticisi Oluştur) panosu görüntülenir.
4. İletişim kutusunu tamamlayın (Adım 1) ve Next>seçeneğini tıklayın.
5. İletişim kutusunu tamamlayın (Adım 2) ve Next>seçeneğini tıklayın.
6. Complete the dialog (Step 3), ensuring that Start Queue Manager and Create Server Connection Channel are not selected, then click Next>.
7. İletişim kutusunu tamamlayın (Adım 4) ve Finish seçeneğini tıklayın.
8. [“Kuyruk Yöneticisinin MSCS Depolaması için taşınması” sayfa 324' a geçin.](#)

Kuyruk Yöneticisinin MSCS Depolaması için taşınması

Bu yordam, var olan bir kuyruk yöneticisini MSCS denetimi altına yerleştirmeye uygun hale getirmek için yapılandırır.

Bunu başarmak için, günlük dosyalarını ve veri dosyalarını, hata durumunda diğer bilgisayar tarafından kullanılabilir duruma getirmek için paylaşılan disklere taşınmasını sağlar. Örneğin, varolan kuyruk yöneticisi `C:\WebSphere MQ\log\<QMname>` ve `C:\WebSphere MQ\qmgrs\<QMname>` gibi yollara sahip olabilir. Dosyaları elle taşımak için **not** seçeneğini tıklayın; bu konuda açıklandığı gibi, WebSphere MQ MSCS Desteği 'nin bir parçası olarak sağlanan yardımcı programı kullanın.

Taşınmakta olan kuyruk yöneticisi SSL bağlantıları kullanıyorsa ve SSL anahtar havuzu yerel makineden kuyruk yöneticisi veri dizininde yer aldıysa, anahtar havuzu kuyruk yöneticisinin geri kalanıyla paylaşılan diske taşınır. Varsayılan olarak, SSL anahtar havuzu yerini (SSLKEYR) belirten kuyruk yöneticisi özniteliği, kuyruk yöneticisi veri dizini altında bulunan `MQ_INSTALLATION_PATH\qmgrs\QMGRNAME\ssl\key` olarak ayarlanır. `MQ_INSTALLATION_PATH` WebSphere MQ ' un kurulu olduğu üst düzey dizini temsil eder. `hamvmqm` komutu, bu kuyruk yöneticisi özniteliğini değiştirmiyor. In this situation you must modify the queue manager attribute, SSLKEYR, using the IBM WebSphere MQ Explorer or the MQSC command ALTER QMGR, to point to the new SSL key repository file.

Yordam aşağıdaki gibidir:

1. Kuyruk yöneticisini sona erdirin ve hata olmadığını doğrulayın.
2. Kuyruk yöneticisinin günlük dosyaları ya da kuyruk dosyaları önceden paylaşılan bir disk üzerinde saklanıyorsa, bu yordamın geri kalan kısmını atlayın ve doğrudan [“Kuyruk yöneticisini MSCS denetimin altına koyma” sayfa 325' e geçin.](#)
3. Kuyruk dosyalarının ve günlük kütüklerinin tam bir ortam yedeklemesini yapın ve yedeklemeyi güvenli bir yerde saklayın (bunun neden önemli olduğunu görmek için bkz. [“Kuyruk yöneticisi günlük dosyaları” sayfa 334](#)).
4. Önceden uygun bir paylaşılan disk kaynağınız varsa, adım 6 'ya devam edin. Ters durumda, MSCS Cluster Administrator ürününü kullanarak, kuyruk yöneticisi günlük dosyalarını ve veri (kuyruk) dosyalarını saklamak için yeterli kapasiteye sahip *paylaşılan disk* tipinde bir kaynak yaratın.

5. Paylaşılan disk, MSCS Cluster Administrator olanağını kullanarak tek bir küme düğümünden diğerine taşımak için sınavarak sınavarak yeniden deneyin.
6. Paylaşılan diskin, kuyruk yöneticisi günlüğünün ve veri dosyalarının yerel olarak saklandığı küme düğümündeki çevrimiçi olduğundan emin olun.
7. Kuyruk yöneticisini aşağıdaki gibi taşımak için yardımcı programı çalıştırın:

```
hamvmqm /m qmname /dd "e:\
WebSphere MQ" /ld "e:\
WebSphere MQ\log"
```

substituting your queue manager name for *qmname*, your shared disk drive letter for *E*, and your chosen directory for *WebSphere MQ*. Dizin önceden yoksa, yaratılırlar.

8. Test the queue manager to ensure that it works, using the IBM WebSphere MQ Explorer. Örneğin:
 - a. Kuyruk yöneticisi ağaç düğümünü farenin sağ düğmesiyle tıklatın ve Start öğesini seçin. Kuyruk yöneticisi başlatılır.
 - b. Queues ağaç düğümünü sağ tıklatın, ardından New->Local Queue . . . seçeneğini belirleyin ve kuyruğa bir ad verin.
 - c. Finish öğesini tıklatın.
 - d. Kuyruğu sağ tıklatın ve Put Test Message . . . seçeneğini belirleyin. Put Test Message (Sınama Sınaması İletisi) panosu görüntülenir.
 - e. Bir ileti metni yazın, ardından Put Test Message seçeneğini tıklatın ve panoyu kapatın.
 - f. Kuyruğu sağ tıklatın ve Browse Messages . . . seçeneğini belirleyin. İleti Tarayıcısı panosu görüntülenir.
 - g. İletinizin kuyruğun üzerinde olduğundan emin olun ve Close seçeneğini tıklatın. İleti Tarayıcısı panosu kapanır.
 - h. Kuyruğu sağ tıklatın ve Clear Messages . . . seçeneğini belirleyin. Kuyruktaki iletiler temizlenir.
 - i. Kuyruğu sağ tıklatın ve Delete . . . seçeneğini belirleyin. Bir onay panosu görüntülenir, OK seçeneğini tıklatın. Kuyruk silindi.
 - j. Kuyruk yöneticisi ağaç düğümünü farenin sağ düğmesiyle tıklatın ve Stop . . . öğesini seçin. End Queue Manager panosu görüntülenir.
 - k. OK öğesini tıklatın. Kuyruk yöneticisi durur.
9. WebSphere MQ Denetimcisi olarak, kuyruk yöneticisinin başlatma özneliğinin el ile ayarlandığını doğrulayın. IBM WebSphere MQ Explorer 'da, Startup alanını kuyruk yöneticisi özellikleri panosunda manual olarak ayarlayın.
10. ["Kuyruk yöneticisini MSCS denetimin altına koyma" sayfa 325' a geçin.](#)

Kuyruk yöneticisini MSCS denetimin altına koyma

Önkoşul görevleri de dahil olmak üzere, MSCS denetimi altına kuyruk yöneticisi yerleştirmede yer alan görevler.

Bir kuyruk yöneticisini MSCS denetimi altına sokmadan önce

Bir kuyruk yöneticisini MSCS denetimin altına yerleştirmeden önce aşağıdaki görevleri gerçekleştirin:

1. Ensure that IBM WebSphere MQ and its MSCS Support are installed on both machines in the cluster and that the software on each computer is identical, as described in ["MSCS kümelemesi için IBM WebSphere MQ ' nin ayarlanması" sayfa 321.](#)
2. Use the **haregyp** utility program to register WebSphere MQ as an MSCS resource type on all the cluster nodes. Ek bilgi için bkz. ["IBM WebSphere MQ MSCS desteği yardımcı programları" sayfa 336 .](#)
3. Kuyruk yöneticisini henüz yaratmadıysanız, bkz. ["MSCS ile kullanılmak üzere kuyruk yöneticisi yaratılması" sayfa 323.](#)

4. Kuyruk yöneticisini yarattığınız ya da önceden varsa, yordamı "Kuyruk Yöneticisinin MSCS Depolaması için taşınması" sayfa 324' ta gerçekleştirdiğinizden emin olun.
5. Çalışmakta olan kuyruk yöneticisini bir komut istemi ya da IBM WebSphere MQ Explorer kullanarak durdurun.
6. Bu konudaki aşağıdaki Windows yordamlarından birine geçmeden önce, paylaşılan sürücülerin MSCS işlemini test edin.

Windows Server 2012



Uyarı: MSCS desteği, 32 bit DLL kullanan WebSphere MQ 7.5 ' te teslim edilir. Due to a restriction in Windows 2012, the IBM WebSphere MQ queue manager does not fail over following a restart.

Microsoft, Pencereleer 2012 ile 32 bit DLL ' lerin kullanılmasından kullanımdan kaldırılmıştır ve bu nedenle, şu anda bu sorun için işletim sistemi düzeltmesi yoktur. IBM , IBM WebSphere MQ 7.5 için 64 bit kitaplık sağlamıyor.

From IBM MQ 8.0 a 64-bit library is available, so you must use this version of the product for full MSCS functionality with Pencereleer 2012, and later.

To place a queue manager under MSCS control on Pencereleer Server 2012, use the following procedure:

1. Kuyruk yöneticisini bulunduran küme düğümü bilgisayarında oturum açın ya da küme denetimi izinlerine sahip bir kullanıcı olarak uzak bir iş istasyonunda oturum açın ve kuyruk yöneticisini bulunduran küme düğümüne bağlanın.
2. Hata Durumunda Yedek Sisteme Geçiş Kümesi Yönetimi aracını başlatın.
3. **Failover Cluster Management > Connect Cluster ...**(Yedek Sisteme Geçiş Kümesi Yönetimi > (kümeyle bir bağlantı açmak için).
4. Önceki Windowssürümlerinde MSCS Cluster Administrator 'da kullanılan grup şemasının tersine, Yedek Sisteme Geçiş Kümesi Yönetimi aracı, hizmet ve uygulama kavramını kullanır. Yapılandırılan bir hizmet ya da uygulama, bir uygulamanın kümelenmesi için gereken tüm kaynakları içerir. Bir kuyruk yöneticisini MSCS altında aşağıdaki gibi yapılandırabilirsiniz:

- a. Kümeyi farenin sağ düğmesiyle tıklatın ve yapılanış sihirbazını başlatmak için **Rolü Yapılandır** ögesini seçin.
- b. "Hizmet ya da Uygulama Seç" panosunda **Diğer Sunucu** seçeneğini belirleyin.
- c. İstemci erişim noktası olarak uygun bir IP adresi seçin.

Bu adresin, istemcilerin ve diğer kuyruk yöneticilerinin *sanal* kuyruk yöneticisine bağlanmaları için kullanılmayan bir IP adresi olması gerekir. Bu IP adresi, her iki düğümün olağan (statik) adresi değildir; bu adresler arasında *döşemeler* yer alan ek bir adrestir. Bu adresin yönlendirilmesi MSCS tarafından işlenmesine rağmen, **değil** adresinin adresin ulaşılabildiğine emin olun.

- d. Kuyruk yöneticisi tarafından özel kullanım için bir depolama aygıtı atayın. Bu aygıt atanmadan önce bir kaynak eşgörünümü olarak yaratılmalıdır.

Hem günlükleri, hem de kuyruk dosyalarını depolamak için bir sürücü kullanabilir ya da bunları sürücülerde bölebilirsiniz. Her iki durumda da, her kuyruk yöneticisinin kendi paylaşılan diski varsa, bu kuyruk yöneticisi tarafından kullanılan tüm sürücülerin bu kuyruk yöneticisine dışlayıcı olduğundan emin olun; başka bir şey, başka hiçbir şeyin sürücülere güvenmemesine neden olur. Ayrıca, kuyruk yöneticisinin kullandığı her sürücü için bir kaynak somut örneği yaratıp yaratmadığınızı da denetleyin.

Bir sürücüye ilişkin kaynak tipi, kullandığınız SCSI desteğine bağlıdır; SCSI bağdaştırıcınızla ilgili yönergelere bakın. Paylaşılan sürücülerin her biri için önceden gruplar ve kaynaklar olabilir. Böyle bir durumda, her bir sürücü için kaynak somut örneği yaratmanız gerekmez. Bunu, kuyruk yöneticisi için yaratılacak yürürlükteki grubundan bir gruba taşıyın.

Her bir sürücü kaynağı için, olası sahipleri her iki düğüme de ayarlayın. Bağımlı kaynakları yok olarak ayarlayın.

- e. "Kaynak Tipi Seç" panelinde **IBM MQSeries MSCS** kaynağını seçin.

- f. Sihirbazda kalan adımları tamamlayın.
5. Kaynağı çevrimiçi hale getirmeden önce, IBM MQSeries MSCS kaynağının ek yapılandırlara gerek vardır:
- 'Yeni IBM MQSeries MSCS' adı verilen bir kaynağı içeren yeni tanımlanan hizmeti seçin.
 - MQ kaynağında **Özellikler** seçeneğini sağ tıklayın.
 - Kaynağı yapılandır:
 - Name; hangi kuyruk yöneticisinin hangi kuyruk için olduğunu belirlemesini kolaylaştıran bir ad seçin.
 - Run in a separate Resource Monitor; daha iyi yalıtım için
 - Possible owners; her iki düğümü de ayarlayın
 - Dependencies; bu kuyruk yöneticisine ilişkin sürücüyü ve IP adresini ekleyin.

Uyarı: Bu bağımlılıklar eklenememesi, IBM WebSphere MQ ' un kuyruk yöneticisi durumunu faikseverler sırasında yanlış küme diskine yazmayı denediği anlamına gelir. Birçok işlem bu diske aynı anda yazma girişiminde bulunulabileceğinden, bazı IBM WebSphere MQ işlemlerinin çalışması engellenebilir.

 - Parameters; aşağıdaki gibi:
 - QueueManagerName (gerekli); bu kaynağın denetime konacağı kuyruk yöneticisinin adı. Bu kuyruk yöneticisi yerel bilgisayarda var olmalıdır.
 - PostOnlineCommand (isteğe bağlı); kuyruk yöneticisi kaynağı durumunu çevrimdışıdan çevrimiçiye değiştirdiğinde çalıştırılacak bir program belirleyebilirsiniz. Daha fazla ayrıntı için bkz. [“PostOnlineKomutu ve PreOfflineKomutu” sayfa 335.](#)
 - PreOfflineCommand (isteğe bağlı); kuyruk yöneticisi kaynağı durumunu çevrimiçi olarak çevrimdışıya değiştirdiğinde çalıştırılacak bir program belirleyebilirsiniz. Daha fazla ayrıntı için bkz. [“PostOnlineKomutu ve PreOfflineKomutu” sayfa 335.](#)

Not: *looksAlive* yoklama aralığı, varsayılan değer olan 5000 ms olarak ayarlanır. *isAlive* yoklama aralığı varsayılan değer olarak 60000 ms olarak ayarlanır. Bu varsayılan değerler yalnızca, kaynak tanımlaması tamamlandıktan sonra değiştirilebilir. Daha fazla ayrıntı için bkz. [“looksAlive ve isAlive yoklama özeti” sayfa 331.](#) - Optionally, set a preferred node (but note the comments in [“Yeğlenen düğümlerin kullanılması” sayfa 335](#))
 - Yedek Sisteme Geçiş* varsayılan olarak duyarlı değerlere ayarlanır, ancak kuyruk yöneticisine yerleştirilen yüklerle eşleşmesi için *Kaynak Yedek Kaynak* ve *Hata Durumunda Grupla* ' i denetleyen eşik ve dönemleri ayarlayabilirsiniz.
6. Kuyruk yöneticisini, MSCS Cluster Administrator 'da çevrimiçi hale getirerek ve bir sınama iş yüküne çıkartarak sınavın. Test kuyruğu yöneticisiyle deniyorsanız, IBM WebSphere MQ Gezgini 'ni kullanın. Örneğin:
- Queues ağaç düğümünü sağ tıklayın, ardından New->Local Queue . . .seçeneğini belirleyin ve kuyruğa bir ad verin.
 - Finishögesini tıklayın. Kuyruk yaratılır ve içerik görünümünde görüntülenir.
 - Kuyruğu sağ tıklayın ve Put Test Message . . .seçeneğini belirleyin. Put Test Message (Sınama Sınaması İletisi) panosu görüntülenir.
 - Bir ileti metni yazın, ardından Put Test Messageseçeneğini tıklayın ve panoyu kapatın.
 - Kuyruğu sağ tıklayın ve Browse Messages . . .seçeneğini belirleyin. İleti Tarayıcısı panosu görüntülenir.
 - İletinizin kuyruğun üzerinde olduğundan emin olun ve Close ögesini tıklayın. İleti Tarayıcısı panosu kapanır.
 - Kuyruğu sağ tıklayın ve Clear Messages . . .seçeneğini belirleyin. Kuyruktaki iletiler temizlenir.
 - Kuyruğu sağ tıklayın ve Delete . . .seçeneğini belirleyin. Bir onay panosu görüntülenir, OKseçeneğini tıklayın. Kuyruk silindi.

7. Kuyruk yöneticisinin MSCS Cluster Administrator programını kullanarak çevrimdışı ve yeniden çevrimiçi duruma alınabileceğini test edin.
8. Yedeği devreye sokma benzetimi

MSCS kümesi denetimcisinde, kuyruk yöneticisini içeren grubu farenin sağ düğmesiyle tıklatın ve Move Group öğesini seçin. Bu işlem birkaç dakika sürebilir. (Bir kuyruk yöneticisini başka bir düğüme hızlı bir şekilde taşımak istiyorsanız, "[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)" sayfa 324 içindeki yordamı izleyin.) You can also right-click and select **Initiate Failure**; the action (local restart or failover) depends on the current state and the configuration settings.

Windows Server 2008

To place a queue manager under MSCS control on Pencereler Server 2008, use the following procedure:

1. Kuyruk yöneticisini bulunduran küme düğümü bilgisayarında oturum açın ya da küme denetimi izinlerine sahip bir kullanıcı olarak uzak bir iş istasyonunda oturum açın ve kuyruk yöneticisini bulunduran küme düğümüne bağlanın.
2. Hata Durumunda Yedek Sisteme Geçiş Kümesi Yönetimi aracını başlatın.
3. **Failover Cluster Management > Manage a Cluster ...**(Hata Durumunda Yedek Sisteme Geçiş Yönetimi (kümeye bir bağlantı açmak için).
4. Önceki Windowssürümlerinde MSCS Cluster Administrator 'da kullanılan grup şemasının tersine, Yedek Sisteme Geçiş Kümesi Yönetimi aracı, hizmet ve uygulama kavramını kullanır. Yapılandırılan bir hizmet ya da uygulama, bir uygulamanın kümelenmesi için gereken tüm kaynakları içerir. Bir kuyruk yöneticisini MSCS altında aşağıdaki gibi yapılandırabilirsiniz:
 - a. **Hizmetler ve Uygulamalar > Bir Hizmet ya da Uygulama Yapılandır ...** öğelerini sağ tıklatın. Konfigürasyon sihirbazını başlatmak için.
 - b. "Hizmet ya da Uygulama Seç" panosunda **Diğer Sunucu** seçeneğini belirleyin.
 - c. İstemci erişim noktası olarak uygun bir IP adresi seçin.

Bu adresin, istemcilerin ve diğer kuyruk yöneticilerinin *sanal* kuyruk yöneticisine bağlanmaları için kullanılmayan bir IP adresi olması gerekir. Bu IP adresi, her iki düğümün olağan (statik) adresi değildir; bu adresler arasında *döşemeler* yer alan ek bir adrestir. Bu adresin yönlendirilmesi MSCS tarafından işlenmesine rağmen, **değil** adresinin adresin ulaşılabildiğine emin olun.
 - d. Kuyruk yöneticisi tarafından özel kullanım için bir depolama aygıtı atayın. Bu aygıt atanmadan önce bir kaynak eşgörünümü olarak yaratılmalıdır.

Hem günlükleri, hem de kuyruk dosyalarını depolamak için bir sürücü kullanabilir ya da bunları sürücülerde bölebilirsiniz. Her iki durumda da, her kuyruk yöneticisinin kendi paylaşılan disk varsa, bu kuyruk yöneticisi tarafından kullanılan tüm sürücülerin bu kuyruk yöneticisine dışlayıcı olduğundan emin olun; başka bir şey, başka hiçbir şeyin sürücülere güvenmemesine neden olur. Ayrıca, kuyruk yöneticisinin kullandığı her sürücü için bir kaynak somut örneği yaratıp yaratmadığınızı da denetleyin.

Bir sürücüye ilişkin kaynak tipi, kullandığınız SCSI desteğine bağlıdır; SCSI bağdaştırıcınızla ilgili yönergelere bakın. Paylaşılan sürücülerin her biri için önceden gruplar ve kaynaklar olabilir. Böyle bir durumda, her bir sürücü için kaynak somut örneği yaratmanız gerekmez. Bunu, kuyruk yöneticisi için yaratılacak yürürlükteki grubundan bir gruba taşıyın.

Her bir sürücü kaynağı için, olası sahipleri her iki düğüme de ayarlayın. Bağımlı kaynakları yok olarak ayarlayın.
 - e. "Kaynak Tipi Seç" panelinde **IBM MQSeries MSCS** kaynağını seçin.
 - f. Sihirbazda kalan adımları tamamlayın.
5. Kaynağı çevrimiçi hale getirmeden önce, IBM MQSeries MSCS kaynağının ek yapılanışlara gerek vardır:
 - a. 'Yeni IBM MQSeries MSCS' adı verilen bir kaynağı içeren yeni tanımlanan hizmeti seçin.
 - b. MQ kaynağında **Özellikler** seçeneğini sağ tıklatın.
 - c. Kaynağı yapılandır:

- Name; hangi kuyruk yöneticisinin hangi kuyruk için olduğunu belirlemesini kolaylaştıran bir ad seçin.
- Run in a separate Resource Monitor; daha iyi yalıtım için
- Possible owners; her iki düğümü de ayarlayın
- Dependencies; bu kuyruk yöneticisine ilişkin sürücüyü ve IP adresini ekleyin.

Uyarı: Bu bağımlılıklar eklenememesi, WebSphere MQ ' un kuyruk yöneticisi durumunu faikseverler sırasında yanlış küme diskine yazmayı denediği anlamına gelir. Birçok işlem bu diske aynı anda yazma girişiminde bulunulabileceğinden, bazı IBM WebSphere MQ işlemlerinin çalışması engellenebilir.

- Parameters; aşağıdaki gibi:

- QueueManagerName (gerekli); bu kaynağın denetime konacağı kuyruk yöneticisinin adı. Bu kuyruk yöneticisi yerel bilgisayarda var olmalıdır.
- PostOnlineCommand (isteğe bağlı); kuyruk yöneticisi kaynağı durumunu çevrimdışıdan çevrimiçiye değiştirdiğinde çalıştırılacak bir program belirleyebilirsiniz. Daha fazla ayrıntı için bkz. [“PostOnlineKomutu ve PreOfflineKomutu” sayfa 335.](#)
- PreOfflineCommand (isteğe bağlı); kuyruk yöneticisi kaynağı durumunu çevrimiçi olarak çevrimdışıya değiştirdiğinde çalıştırılacak bir program belirleyebilirsiniz. Daha fazla ayrıntı için bkz. [“PostOnlineKomutu ve PreOfflineKomutu” sayfa 335.](#)

Not: *looksAlive* yoklama aralığı, varsayılan değer olan 5000 ms olarak ayarlanır. *isAlive* yoklama aralığı varsayılan değer olarak 60000 ms olarak ayarlanır. Bu varsayılan değerler yalnızca, kaynak tanımlaması tamamlandıktan sonra değiştirilebilir. Daha fazla ayrıntı için bkz. [“looksAlive ve isAlive yoklama özeti” sayfa 331.](#)

- Optionally, set a preferred node (but note the comments in [“Yeğlenen düğümlerin kullanılması” sayfa 335](#))
 - Yedek Sisteme Geçiş* varsayılan olarak duyarlı değerlere ayarlanır, ancak kuyruk yöneticisine yerleştirilen yüklerle eşleşmesi için *Kaynak Yedek Kaynak* ve *Hata Durumunda Grupla* ' i denetleyen eşik ve dönemleri ayarlayabilirsiniz.
- Kuyruk yöneticisini, MSCS Cluster Administrator 'da çevrimiçi hale getirerek ve bir sınama iş yüküne çıkartarak sınavın. Test kuyruğu yöneticisiyle deniyorsanız, IBM WebSphere MQ Gezginini 'ni kullanın. Örneğin:
 - Queues ağaç düğümünü sağ tıklayın, ardından New->Local Queue . . . seçeneğini belirleyin ve kuyruğa bir ad verin.
 - Finish ögesini tıklayın. Kuyruk yaratılır ve içerik görünümünde görüntülenir.
 - Kuyruğu sağ tıklayın ve Put Test Message . . . seçeneğini belirleyin. Put Test Message (Sınama Sınaması İletisi) panosu görüntülenir.
 - Bir ileti metni yazın, ardından Put Test Message seçeneğini tıklayın ve panoyu kapatın.
 - Kuyruğu sağ tıklayın ve Browse Messages . . . seçeneğini belirleyin. İleti Tarayıcısı panosu görüntülenir.
 - İletinizin kuyruğun üzerinde olduğundan emin olun ve Close ögesini tıklayın. İleti Tarayıcısı panosu kapanır.
 - Kuyruğu sağ tıklayın ve Clear Messages . . . seçeneğini belirleyin. Kuyruktaki iletiler temizlenir.
 - Kuyruğu sağ tıklayın ve Delete . . . seçeneğini belirleyin. Bir onay panosu görüntülenir, OK seçeneğini tıklayın. Kuyruk silindi.
 - Kuyruk yöneticisinin MSCS Cluster Administrator programını kullanarak çevrimdışı ve yeniden çevrimiçi duruma alınabileceğini test edin.
 - Yedeği devreye sokma benzetimi

MSCS kümesi denetimcisinde, kuyruk yöneticisini içeren grubu farenin sağ düğmesiyle tıklayın ve Move Group ögesini seçin. Bu işlem birkaç dakika sürebilir. (Bir kuyruk yöneticisini başka bir düğüme hızlı bir şekilde taşımak istiyorsanız, [“Kuyruk Yöneticisinin MSCS Depolaması için taşınması” sayfa 324](#) içindeki

yordamı izleyin.) You can also right-click and select **Initiate Failure**; the action (local restart or failover) depends on the current state and the configuration settings.

Windows 2003

To place a queue manager under MSCS control on Pencereler 2003, use the following procedure:

1. Kuyruk yöneticisini bulunduran küme düğümü bilgisayarında oturum açın ya da küme denetimi izinlerine sahip bir kullanıcı olarak uzak bir iş istasyonunda oturum açın ve kuyruk yöneticisini bulunduran küme düğümüne bağlanın.
2. MSCS Cluster Administrator programını başlatın.
3. Kümeyle bir bağlantı açın.
4. Kuyruk yöneticisine ilişkin kaynakları içermek üzere kullanılacak bir MSCS grubu yaratın. Grubu, hangi kuyruk yöneticisinin ilişkilendireceği belli olduğu gibi bir şekilde adlayın. Her grup, "MSCS ile birden çok kuyruk yöneticisinin kullanılması" sayfa 322 içinde açıklandığı gibi birden çok kuyruk yöneticisi içerebilir.

Kalan tüm adımlar için grubu kullanın.

5. Kuyruk yöneticisinin kullandığı SCSI mantıksal sürücülerinin her biri için bir kaynak eşgörünümü yaratın.

Hem günlükleri, hem de kuyruk dosyalarını depolamak için bir sürücü kullanabilir ya da bunları sürücülerde bölebilirsiniz. Her iki durumda da, her kuyruk yöneticisinin kendi paylaşılan diski varsa, bu kuyruk yöneticisi tarafından kullanılan tüm sürücülerin bu kuyruk yöneticisine dışlayıcı olduğundan emin olun; başka bir şey, başka hiçbir şeyin sürücülere güvenmemesine neden olur. Ayrıca, kuyruk yöneticisinin kullandığı her sürücü için bir kaynak somut örneği yaratıp yaratmadığınızı da denetleyin.

Bir sürücüye ilişkin kaynak tipi, kullandığınız SCSI desteğine bağlıdır; SCSI bağdaştırıcınızla ilgili yönergelere bakın. Paylaşılan sürücülerin her biri için önceden gruplar ve kaynaklar olabilir. Böyle bir durumda, her bir sürücü için kaynak somut örneği yaratmanız gerekmez. Bunu, kuyruk yöneticisi için yaratılacak yürürlükteki grubundan bir gruba taşıyın.

Her bir sürücü kaynağı için, olası sahipleri her iki düğüme de ayarlayın. Bağımlı kaynakları yok olarak ayarlayın.

6. IP adresi için bir kaynak somut örneği yaratın.

Bir IP adresi kaynağı yaratın (kaynak tipi *IP adresi*). Bu adresin, istemcilerin ve diğer kuyruk yöneticilerinin *sanal* kuyruk yöneticisine bağlanmaları için kullanılmayan bir IP adresi olması gerekir. Bu IP adresi, her iki düğümün olağan (statik) adresi değildir; bu adresler arasında *döşemeler* yer alan ek bir adrestir. Bu adresin yöneltmesi MSCS tarafından işlenmesine rağmen, **değil** adresinin adresin ulaşılabilirliğine emin olun.

7. Kuyruk yöneticisi için bir kaynak somut örneği yaratın.

IBM WebSphere MQ MSCStipli bir kaynak yaratın. Sihirbazda aşağıdakiler de içinde olmak üzere çeşitli öğeler için bilgi isteminde bulunur:

- Name; hangi kuyruk yöneticisinin hangi kuyruk için olduğunu belirlemesini kolaylaştıran bir ad seçin.
- Add to group; oluşturduğunuz grubu kullanın
- Run in a separate Resource Monitor; daha iyi yalıtım için
- Possible owners; her iki düğümü de ayarlayın
- Dependencies; bu kuyruk yöneticisine ilişkin sürücüyü ve IP adresini ekleyin.

Uyarı: Bu bağımlılıklar eklenememesi, WebSphere MQ ' un kuyruk yöneticisi durumunu faikseverler sırasında yanlış küme diskine yazmayı denediği anlamına gelir. Birçok işlem bu diske aynı anda yazma girişiminde bulunulabileceğinden, bazı IBM WebSphere MQ işlemlerinin çalışması engellenebilir.

- Parameters; aşağıdaki gibi:

- `QueueManagerName` (gerekli); bu kaynağın denetime konacağı kuyruk yöneticisinin adı. Bu kuyruk yöneticisi yerel bilgisayarda var olmalıdır.
- `PostOnlineCommand` (isteğe bağlı); kuyruk yöneticisi kaynağı durumunu çevrimdışıdan çevrimiçiye değiştirdiğinde çalıştırılacak bir program belirleyebilirsiniz. Daha fazla ayrıntı için bkz. "[PostOnlineKomutu ve PreOfflineKomutu](#)" sayfa 335.
- `PreOfflineCommand` (isteğe bağlı); kuyruk yöneticisi kaynağı durumunu çevrimiçi olarak çevrimdışıya değiştirdiğinde çalıştırılacak bir program belirleyebilirsiniz. Daha fazla ayrıntı için bkz. "[PostOnlineKomutu ve PreOfflineKomutu](#)" sayfa 335.

Not: `looksAlive` yoklama aralığı, varsayılan değer olan 5000 ms olarak ayarlanır. `isAlive` yoklama aralığı, 30000 ms varsayılan değeri olarak ayarlanır. Bu varsayılan değerler yalnızca, kaynak tanımlaması tamamlandıktan sonra değiştirilebilir. Daha fazla ayrıntı için bkz. "[looksAlive ve isAlive yoklama özeti](#)" sayfa 331.

8. Optionally, set a preferred node (but note the comments in "[Yeğlenen düğümlerin kullanılması](#)" sayfa 335)
9. The *Yedek Sisteme Geçiş* (as defined in the properties for the group) is set by default to sensible values, but you can tune the thresholds and periods that control *Kaynak Yedek Kaynak* and *Hata Durumunda Grupla* to match the loads placed on the queue manager.
10. Kuyruk yöneticisini, MSCS Cluster Administrator 'da çevrimiçi hale getirerek ve bir sınama iş yüküne çıkartarak sınavın. Test kuyruğu yöneticisiyle deniyorsanız, IBM WebSphere MQ Gezgini 'ni kullanın. Örneğin:
 - a. Queues ağaç düğümünü sağ tıklayın, ardından New->Local Queue . . . seçeneğini belirleyin ve kuyruğa bir ad verin.
 - b. Finish ögesini tıklayın. Kuyruk yaratılır ve içerik görünümünde görüntülenir.
 - c. Kuyruğu sağ tıklayın ve Put Test Message . . . seçeneğini belirleyin. Put Test Message (Sınama Sınaması İletisi) panosu görüntülenir.
 - d. Bir ileti metni yazın, ardından Put Test Messageseçeneğini tıklayın ve panoyu kapatın.
 - e. Kuyruğu sağ tıklayın ve Browse Messages . . . seçeneğini belirleyin. İleti Tarayıcısı panosu görüntülenir.
 - f. İletinizin kuyruğun üzerinde olduğundan emin olun ve Close ögesini tıklayın. İleti Tarayıcısı panosu kapanır.
 - g. Kuyruğu sağ tıklayın ve Clear Messages . . . seçeneğini belirleyin. Kuyruktaki iletiler temizlenir.
 - h. Kuyruğu sağ tıklayın ve Delete . . . seçeneğini belirleyin. Bir onay panosu görüntülenir, OKseçeneğini tıklayın. Kuyruk silindi.
11. Kuyruk yöneticisinin MSCS Cluster Administrator programını kullanarak çevrimdışı ve yeniden çevrimiçi duruma alınabileceğini test edin.
12. Yedeği devreye sokma benzetimi

MSCS kümesi denetimcisinde, kuyruk yöneticisini içeren grubu farenin sağ düğmesiyle tıklayın ve Move Group ögesini seçin. Bu işlem birkaç dakika sürebilir. (Bir kuyruk yöneticisini başka bir düğüme hızlı bir şekilde taşımak istiyorsanız, "[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)" sayfa 324 başlıklı bölümdeki yordamı izleyin.) You can also right-click and select *Initiate Failure*; the action (local restart or failover) depends on the current state and the configuration settings.

looksAlive ve isAlive yoklama özeti

`looksAlive` ve `isAlive`, MSCS ' nin kaynak tiplerine verilen kitaplık kodunu geri çağırdığı aralıklardır ve kaynağın kendi durumunu belirlemek için denetler gerçekleştirilmesini ister. Bu, en sonunda MSCS ' nin kaynak üzerinde başarısız olup olmadığını belirler.

`looksAlive` aralığı (varsayılan 5000 ms) olduğunda, kuyruk yöneticisi kaynağının durumunun tatmin edici olup olmadığını saptamak için, kuyruk yöneticisi kaynağının kendi denetini gerçekleştirmek üzere çağırılması gerekir.

isAlive aralığının (varsayılan 30000 ms) her durumda, kaynağın doğru çalışıp çalışmadığını belirlemek için kuyruk yöneticisi kaynağının başka bir denetimi gerçekleştirmesi için başka bir çağrı yapılır. Bu, iki düzey kaynak tipi denetlemesini etkinleştirir.

1. Kaynağın çalışıp çalışmadığını belirlemek için bir *looksAlive* durum denetimi.
2. Kuyruk yöneticisi kaynağının etkin olup olmadığını belirlemek için daha önemli bir *isAlive* denetimi.

Kuyruk yöneticisi kaynağının etkin olmadığı saptanırsa, diğer gelişmiş MSCS seçeneklerini temel alan MSCS, kaynak ve ilişkili bağımlı kaynaklar için kümedeki başka bir düğüme ilişkin bir hata işlemi tetikler. Ek bilgi için [MSCS belgeleri](#) başlıklı konuya bakın.

Kuyruk yöneticisinin MSCS denetiminden kaldırılması

Kuyruk yöneticilerini MSCS denetiminden kaldırabilir ve bunları el ile yönetmeye geri getirebilirsiniz.

Bakım işlemleri için MSCS denetiminden kuyruk yöneticilerini kaldırmanıza gerek yoktur. Bunu, MSCS Cluster Administrator olanağını kullanarak, bir kuyruk yöneticisini geçici olarak çevrimdışı duruma getirebilirsiniz. Bir kuyruk yöneticisini MSCS denetiminden kaldırmak daha kalıcı bir değişiktir; yalnızca, MSCS 'nin kuyruk yöneticisini daha fazla denetime sahip olmasını istememeye karar verdiğinizde bunu yapın.

If the queue manager being removed uses SSL connections you must modify the queue manager attribute, SSLKEYR, using the WebSphere MQ Explorer or the MQSC command ALTER QMGR, to point to the SSL key repository file on the local directory.

Yordam şöyledir:

1. Take the queue manager resource offline using the MSCS Cluster Administrator, as described in [“MSCS 'den çevrimdışı kuyruk yöneticisi alma” sayfa 332](#)
2. Kaynak somut örneğini yok eder. Bu, kuyruk yöneticisini yok etmez.
3. İsteğe bağlı olarak, kuyruk yöneticisi dosyalarını paylaşılan sürücülerden yerel sürücülere geri geçirmeniz gerekir. Bunu yapmak için bkz. [“MSCS depolama alanından kuyruk yöneticisi döndürülüyor” sayfa 332.](#)
4. Kuyruk yöneticisini test edin.

MSCS ' den çevrimdışı kuyruk yöneticisi alma

MSCS ' den bir kuyruk yöneticisini çevrimdışı duruma getirmek için aşağıdaki adımları gerçekleştirin:

1. MSCS Cluster Administrator programını başlatın.
2. Kümeyle bir bağlantı açın.
3. Groupsseçeneğini belirleyin ve taşınmak üzere kuyruk yöneticisini içeren grubu açın.
4. Kuyruk yöneticisi kaynağını seçin.
5. Dosyayı sağ tıklayın ve Offlinesseçeneğini belirleyin.
6. Tamamlanmak için bekleyin.

MSCS depolama alanından kuyruk yöneticisi döndürülüyor

Bu yordam, kuyruk yöneticisinin bilgisayarının yerel sürücüsünde geri olacak şekilde yapılandırıldığını, yani *normal* WebSphere MQ kuyruk yöneticisi olacağını tanımlar. Bunu gerçekleştirmek için, günlük dosyalarını ve veri dosyalarını paylaşılan disklerden taşınmayı sağlar. Örneğin, varolan kuyruk yöneticisi E:\WebSphere MQ\log\hamvmqm yardımcı programını kullanın:

1. Kuyruk yöneticisini sona erdirin ve hata olmadığını doğrulayın.
2. Kuyruk dosyalarının ve günlük kütüklerinin tam bir ortam yedeklemesini yapın ve yedeklemeyi güvenli bir yerde saklayın (bunun neden önemli olduğunu görmek için bkz. [“Kuyruk yöneticisi günlük dosyaları” sayfa 334](#)).

3. Hangi yerel sürücünün kullanılacağını belirleyin ve kuyruk yöneticisi günlük dosyalarını ve veri (kuyruk) dosyalarını saklamak için yeterli kapasiteye sahip olduğundan emin olun.
4. Kuyruk yöneticisi günlüğünün ve veri dosyalarının taşınacağı küme düğümünde, kütüklerin bulunduğu paylaşılan disklerin çevrimiçi durumda olduğundan emin olun.
5. Kuyruk yöneticisini aşağıdaki gibi taşımak için yardımcı programı çalıştırın:

```
hamvmqm /m qmname /dd "c:\
WebSphere MQ" /ld "c:\
WebSphere MQ\log"
```

substituting your queue manager name for *qmname*, your local disk drive letter for *C*, and your chosen directory for *WebSphere MQ* (the directories are created if they do not already exist).

6. Kuyruk yöneticisini test edin ("[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)" sayfa 324 içinde açıklandığı gibi) çalıştığından emin olun.

MSCS kullanımıyla ilgili ipuçları ve öneriler

Bu bölüm, MSCS ' ye ilişkin WebSphere MQ desteğini etkin bir şekilde kullanmanıza yardımcı olacak bazı genel bilgileri içerir.

Bu bölüm, MSCS ' ye ilişkin WebSphere MQ desteğini etkin bir şekilde kullanmanıza yardımcı olacak bazı genel bilgileri içerir.

Bir kuyruk yöneticisinin bir makineden diğerine geçememesi ne kadar zaman alır? Bu, kuyruk yöneticilikteki iş yükü miktarına ve trafik karışımını, örneğin, ne kadarının kalıcı olduğunu, eşitleme noktası içinde ve başarısızlığa ne kadar bağlı olduğunu gösteren iş yükü miktarına bağlıdır. IBM sınamaları, hata durumunda yedek sisteme geçiş ve onarım sonrası geri dönüş sürelerini yaklaşık bir dakika içinde sağlar. Bu, çok hafif yüklü bir kuyruk yöneticisiydi ve gerçek sürler, yüklemeye bağlı olarak önemli ölçüde farklılık gösterecektir.

MSCS ' nin çalışmakta olduğu doğrulanıyor

Çalışmakta olan bir MSCS kümeniz olduğundan emin olmak için aşağıdaki adımları izleyin.

"[MSCS ile kullanılmak üzere kuyruk yöneticisi yaratılması](#)" sayfa 323 ile başlayan görev tanımlamaları, içinde kaynakları yaratabileceğiniz, yeni düzeye geçirebileceğiniz ve yok edebileceğiniz bir MSCS kümesinin olduğunu varsaydığını varsayar. Böyle bir kümeniz olduğundan emin olmak istiyorsanız:

1. MSCS Cluster Administrator olanağını kullanarak bir grup oluşturun.
2. Within that group, create an instance of a generic application resource, specifying the system clock (path name C:\winnt\system32\clock.exe and working directory of C:\).
3. Kaynağı çevrimiçi duruma getirebileceğinizden, bunu içeren grubu diğer düğüme taşıyabileceğinizden ve kaynağı çevrimdışı duruma getirebileceğinizden emin olun.

El ile başlatma

For a queue manager managed by MSCS, you **gerekir** set the startup attribute to manual. Bu, WebSphere MQ MSCS desteğinin, kuyruk yöneticisini hemen başlatmadan IBM MQSeries Service 'i yeniden başlatabilmesini sağlar.

WebSphere MQ MSCS desteğinin, izleme ve denetim gerçekleştirebilmesi için hizmeti yeniden başlatabilmesi, ancak kendisinin de hangi kuyruk yöneticilerinin çalıştırıldığı ve hangi makinelerde olduğu denetimde kalması gerekir. Ek bilgi için "[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)" sayfa 324 başlıklı konuya bakın.

MSCS ve kuyruk yöneticileri

MSCS kullanırken kuyruk yöneticilerine ilişkin dikkat edilecek noktalar.

Diğer düğümde eşleşen bir kuyruk yöneticisi yaratılması

WebSphere MQ ile çalışmak üzere kümeleme için, düğüm A 'da her biri için düğüm B' de aynı kuyruk yöneticisine gerek duyarsınız. Ancak, ikincisini belirttik olarak yaratmanıza gerek yoktur. Bir düğümde bir

kuyruk yöneticisi oluşturabilir ya da hazırlayabilir, “[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)” sayfa 324' de açıklandığı gibi diğer düğüme taşıyabilir ve o düğümden tam olarak yinelenir.

Varsayılan kuyruk yöneticileri

MSCS denetimi altında varsayılan kuyruk yöneticisi kullanmayın. Bir kuyruk yöneticisinin özelliği, onu varsayılan değer kılan bir özelliğe sahip değil; WebSphere MQ kendi ayrı kaydını tutar. Bir kuyruk yöneticisini varsayılan değer olarak yedek sisteme geçiş sırasında diğer bilgisayara taşırsanız, varsayılan değer olarak bu değer varsayılan değer olarak geçerli olmaz. Tüm uygulamalarınızın ada göre belirli kuyruk yöneticilerine başvurmasını sağlar.

Kuyruk Yöneticisinin Silinmesi

Bir kuyruk yöneticisi düğümü taşıdıktan sonra, ayrıntıları her iki bilgisayarda da kayıt defterinde bulunur. When you want to delete it, do so as normal on one computer, and then run the utility described in “[IBM WebSphere MQ MSCS desteği yardımcı programları](#)” sayfa 336 to clean up the registry on the other computer.

Var olan kuyruk yöneticileri için destek

Kuyruk yöneticisi günlük dosyalarını ve kuyruk dosyalarını, iki makine arasındaki paylaşılan SCSI veriyolunda bulunan bir diske koyabilmeniz koşuluyla, MSCS denetimin altına var olan bir kuyruk yöneticisi yerleştirebilirsiniz (bkz. [Şekil 62 sayfa 321](#)). MSCS kaynağı yaratıldığında kuyruk yöneticisini kısa bir süreliğine çevrimdışı duruma getirmeniz gerekir.

Yeni bir kuyruk yöneticisi yaratmak istiyorsanız, bunu MSCS ' den bağımsız olarak yaratın, test edin ve MSCS denetimin altına koyun. Bakınız:

- “[MSCS ile kullanılmak üzere kuyruk yöneticisi yaratılması](#)” sayfa 323
- “[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)” sayfa 324
- “[Kuyruk yöneticisini MSCS denetimin altına koyma](#)” sayfa 325

MSCS ' ye hangi kuyruk yöneticilerinin yönetmeye ilişkin olduğunu

Bu tip her kuyruk yöneticisi için bir kaynak somut örneği yaratmak üzere MSCS kümesi denetimcisi kullanarak MSCS denetimin altına hangi kuyruk yöneticilerinin yerleştirileceğini seçmenizdir. Bu süreç, yönetim ortamının yönetmesini istediğiniz kuyruk yöneticisini seçecek kaynakların listesini içerir.

Kuyruk yöneticisi günlük dosyaları

Bir kuyruk yöneticisini MSCS depolamaya taşıdığınızda, günlük ve veri dosyalarını paylaşılan bir diske taşıyorsunuz (örneğin, “[Kuyruk Yöneticisinin MSCS Depolaması için taşınması](#)” sayfa 324 başlıklı konuya bakın).

Taşınmadan önce, kuyruk yöneticisini temizleme işlemini sona erdirmek ve veri dosyalarının ve günlük dosyalarının tam yedeğini almak için bu öneriyi önerilir.

Birden çok kuyruk yöneticisi

WebSphere MQ MSCS desteği, her makinede birden çok kuyruk yöneticisi çalıştırmanızı ve tek tek kuyruk yöneticilerini MSCS denetimin altına yerleştirmenizi sağlar.

Kümelere yönetmek için her zaman MSCS kullan

Denetim komutlarını ya da IBM WebSphere MQ Explorer ' ı kullanarak, MSCS ' nin denetimi altındaki kuyruk yöneticisinde başlatma ve durdurma işlemlerini doğrudan gerçekleştirmeye çalışmayın. Bunun yerine, kuyruk yöneticisini çevrimiçi duruma getirmek için MSCS Cluster Administrator olanağını kullanın ya da çevrimdışı duruma getirin.

MSCS Cluster Administrator olanağının kullanılması, MSCS ' nin kuyruk yöneticisinin çevrimdışı olduğunu bildirmesinin neden olduğu karışıklığı önler. Bu durum, bunu MSCS denetimi dışında başlatmış olduğunuz bir durumda başlatmış olabilir. Daha ciddi olarak, MSCS ' yi kullanmadan kuyruk yöneticisini durdurma işlemi MSCS tarafından bir hata olarak algılanır ve diğer düğümde hata durumunda yedek sisteme geçiş işlemini başlatmaktadır.

Etkin/Etkin kipte çalışma

MSCS kümesindeki her iki bilgisayar da, kuyruk yöneticilerini Etkin/Etkin kipinde çalıştırabilir. Tam olarak boşta duran bir makinenin yedek olarak hareket etmesine gerek yoktur (ancak, isterseniz, Etkin/Edilgen Kipi 'nde yapabilirsiniz).

İş yükünü çalıştırmak için her iki makineyi de kullanmayı planlıyorsanız, küme iş yükünü tatmin edici düzeyde bir başarımlı düzeyinde çalıştırmak için yeterli kapasiteye sahip (işlemci, bellek, ikincil depolama) her birine yeterli düzeyde destek sağlayın.

Not: MSCS ' yi Microsoft Transaction Server (COM +) ile birlikte kullanıyorsanız, Etkin/Etkin kipi **kullanamazsınız** . Bunun nedeni, MSCS ve COM + ile WebSphere MQ ' yi kullanmak içindir:

- WebSphere MQ'un COM + desteğini kullanan uygulama bileşenleri, COM +' ın bir parçası olan Distributed Transaction Coordinator (DTC) ile aynı bilgisayar üzerinde çalışmalıdır.
- Kuyruk yöneticisi aynı bilgisayar üzerinde de çalışmalıdır.
- DTC, MSCS kaynağı olarak yapılandırılmalı ve bu nedenle, herhangi bir zamanda kümedeki yalnızca bilgisayarlardan yalnızca biri üzerinde çalıştırılabilir.

PostOnlineKomutu ve PreOfflineKomutu

WebSphere MQ MSCS desteğini diğer sistemlerle bütünleştirmek için bu komutları kullanın. Bunları WebSphere MQ komutlarını vermek için kullanabilirsiniz, bazı kısıtlamalar da kullanabilirsiniz.

Specify these commands in the Parameters to a resource of type IBM WebSphere MQ MSCS. Bunları, WebSphere MQ MSCS desteğini diğer sistemlerle ya da yordamlarla tümleştirmek için kullanabilirsiniz. Örneğin, posta iletisi gönderen bir programın adını belirtebilir, bir çağrı aygıtı etkinleştirebilir ya da başka bir izleme sistemi tarafından yakalanacak başka bir uyarı formu oluşturur.

PostOnline, kaynak çevrimdışıdan çevrimiçiye geçtiğinde çağrılır; PreOfflineKomutu çevrimdışıdan çevrimdışına bir değişiklik içininvokek komutu çağrılır. Bu komutlar çağrıldığında, varsayılan olarak Windows sistem dizininden çalıştırılır. WebSphere MQ , Windows 64 bit sistemlerde 32 bitlik bir kaynak izleme işlemi kullandığından, \Windows\system32 dizini değil, \Windows\SysWOW64 dizinidir. Daha fazla bilgi için, Windows x64 ortamında dosya yeniden yönlendirmesiyle ilgili Microsoft belgelerine bakın. Her iki komut da MSCS Küme Hizmeti 'nin çalıştırılması için kullanılan kullanıcı hesabı altında çalışır ve zamanuyumsuz olarak çağrılır; WebSphere MQ MSCS desteği devam etmeden önce bunların tamamlanmasını beklemeyin. Bu, kümeleme işlemlerini engelleyebilir ya da daha fazla küme işlemlerini geciktirme riskini ortadan kaldırır.

Bu komutları, WebSphere MQ komutlarını vermek için de kullanabilirsiniz; örneğin, Requester kanallarını yeniden başlatmanızı sağlar. Ancak, komutlar, kuyruk yöneticisinin durumu geçtiğinde, zaman noktasında çalıştırılır; bu nedenle, kuyruk yöneticisinin yürürlükteki durumu hakkında varsayımlar yapmamalıdır; kuyruk yöneticisi çevrimiçi duruma getirildikten hemen sonra, bir yönetici çevrimdışı bir komut yayınladıktan hemen sonra, komutlar, kuyruk yöneticisinin yürürlükteki durumuyla ilgili olarak varsayımlar yapmamalıdır.

Kuyruk yöneticisinin durumuna bağlı olan programları çalıştırmak istiyorsanız, MSCS Generic Application kaynak tipinin eşgörünümlerini yaratmayı, bunları kuyruk yöneticisi kaynağı olarak aynı MSCS grubuna yerleştirerek ve kuyruk yöneticisi kaynağına bağımlı hale getirmenizi sağlar.

Yeğlenen düğümlerin kullanılması

Her kuyruk yöneticisi için *tercih edilen düğüm* yapılandırmak için Active/Active kipi kullanılırken yararlı olabilir. Ancak genel olarak, tercih edilen bir düğüm ayarlamamak, ancak el ile onarım sonrası geri dönüş yapmak daha iyi olur.

Diğer bazı görelili durum bilgisi olmayan kaynakların tersine, bir kuyruk yöneticisi bir düğümden diğerine (ya da bir düğümden diğerine) geçemez (ya da geri). Gereksiz kesintileri önlemek için, kurtarılan düğümü,

kuyruk yöneticisini geri vermeden önce sınavın. Bu, immediate onarım sonrası geri dönüş ayarının kullanılmasını engeller. Onarım sonrası geri dönüşü, günün belirli saatleri arasında gerçekleşecek şekilde yapılandırabilirsiniz.

Düğümün tam olarak kurtarılacağına eminseniz, en güvenli yol kuyruk yöneticisini el ile gerekli düğüme geri hareket ettirir. Bu, preferred node seçeneğinin kullanılmasından önce gelir.

Uygulama Olay günlüğünde COM + hataları oluşursa

Yeni kurulan bir MSCS kümesine WebSphere MQ ' u kurduğunuzda, Application Event (Uygulama Olay) günlüğünde Source COM + ve Event ID 4691 ile bildirilen bir hata bulabilirsiniz.

Bu, Microsoft Distributed Transaction Coordinator (MSDTC), böyle bir ortamda çalışacak şekilde yapılandırılmadığında, Microsoft Cluster Server (MSCS) ortamında WebSphere MQ ' yı çalıştırmaya çalıştığınızda demektir. MSDTC ' nin kümelenmiş bir ortamda yapılandırılmasına ilişkin bilgi edinmek için Microsoft belgelerine bakın.

IBM WebSphere MQ MSCS desteği yardımcı programları

Bir komut isteminde çalıştırabileceğiniz MSCS yardımcı programları için IBM WebSphere MQ desteğinin bir listesi.

MSCS için IBM WebSphere MQ desteği, aşağıdaki yardımcı programları içerir:

Kaynak tipini kaydettirin ya da kaydını kaldır

haregtyp.exe

After you *kayıt kaldırma* the IBM WebSphere MQ MSCS resource type you can no longer create any resources of that type. MSCS, küme içinde hala bir tip eşgörünümlerinizi varsa, kaynak tipini kaydetmenize izin vermiyor:

1. Using the MSCS Cluster Administrator, stop any queue managers that are running under MSCS control, by taking them offline as described in [“MSCS ' den çevrimdışı kuyruk yöneticisi alma” sayfa 332.](#)
2. MSCS Cluster Administrator olanağını kullanarak, kaynak somut örneklerini silin.
3. Bir komut isteminde, aşağıdaki komutu girerek kaynak tipinin kaydını kaldırın:

```
haregtyp /u
```

Tipi *kaydettirmek* istiyorsanız (ya da daha sonra yeniden kaydettirmek istiyorsanız), komut isteminde aşağıdaki komutu girin:

```
haregtyp /r
```

MSCS kitaplıklarını başarıyla kaydettikten sonra, IBM WebSphere MQ kuruluşundan bu yana bu işlemi yapmadıysanız, sistemi yeniden başlatmanız gerekir.

Kuyruk yöneticisini MSCS depolamaya taşı

hamvmqm.exe

Bkz. [“Kuyruk Yöneticisinin MSCS Depolaması için taşınması” sayfa 324.](#)

Bir düğümden kuyruk yöneticisini silme

had1tmqm.exe

Kümenizde bir kuyruk yöneticinizin olduğu durumu göz önünde bulundurun, bir düğümden diğerine taşınmıştır ve şimdi onu yok etmek istersiniz. Use the IBM WebSphere MQ Explorer to delete it on the node where it currently is. Bunun için kayıt girişleri diğer bilgisayarda var olmaya devam eder. Bunları silmek için, o bilgisayardaki bir komut isteminde aşağıdaki komutu girin:

```
had1tmqm /m qmname
```

Burada qmname , kaldırılacak kuyruk yöneticisinin adıdır.

Kurulum ayrıntılarını denetleyin ve kaydedin

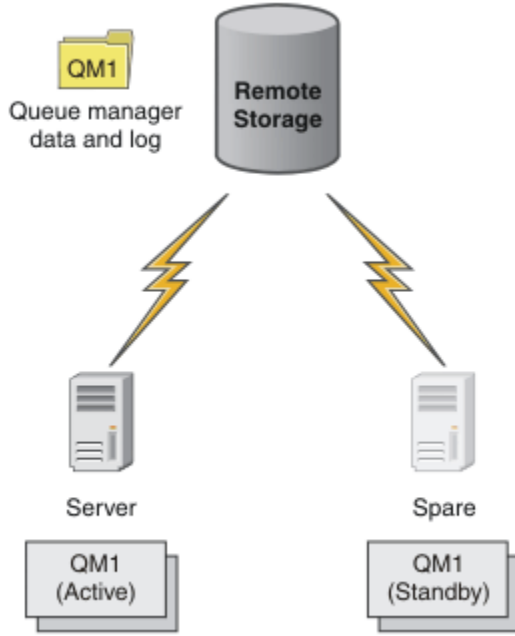
amqmsysn.exe

Bu yardımcı program, IBM WebSphere MQ MSCS Support (Destek) kurulumunuzun tam ayrıntılarını gösteren bir iletişim kutusu sunar; örneğin, IBM desteği için istekte bulunsanız da istenebilir. Ayrıntıları bir dosyaya saklamanız için bir seçenek vardır.

Çok eşgörunümlü kuyruk yöneticileri

Çok eşgörunümlü kuyruk yöneticileri, farklı sunucularda yapılandırılmış aynı kuyruk yöneticisinin eşgörunümleridir. Kuyruk yöneticisinin bir eşgörunümü etkin yönetim ortamı olarak tanımlanır ve başka bir yönetim ortamı yedek yönetim ortamı olarak tanımlanır. Etkin yönetim ortamı başarısız olursa, çoklu yönetim ortamı kuyruk yöneticisi otomatik olarak yedek sunucuda yeniden başlatılır.

Şekil 63 sayfa 337 , QM1 için çok eşgörunümlü bir yapılandırma göstermektedir. IBM WebSphere MQ , biri yedek olmak üzere iki sunucuya kurulur. Bir kuyruk yöneticisi (QM1) yaratıldı. Bir QM1 eşgörunümü etkindir ve bir sunucuda çalışıyor. The other instance of QM1 is running in standby on the other server, doing no active processing, but ready to take over from the active instance of QM1, if the active instance fails.



Şekil 63. Çok eşgörunümlü kuyruk yöneticisi

Bir kuyruk yöneticisini çok eşgörunümlü bir kuyruk yöneticisi olarak kullanmayı amaçladığınızda, **crtmqm** komutunu kullanarak sunuculardan birinde tek bir kuyruk yöneticisi yaratın, kuyruk yöneticisi verilerini ve günlüklerini paylaşılan ağ depolamasındaki günlükler oluşturun. On the other server, rather than create the queue manager again, use the **admqinf** command to create a reference to the queue manager data and logs on the network storage.

Kuyruk yöneticisini şimdi sunuculardan birinden çalıştırabilirsiniz. Her bir sunucu aynı kuyruk yöneticisi verilerine ve günlüklerine gönderme yapar; yalnızca bir kuyruk yöneticisi vardır ve bu, aynı anda yalnızca bir sunucuda etkindir.

Kuyruk yöneticisi tek bir eşgörunüm kuyruk yöneticisi olarak ya da çok eşgörunümlü bir kuyruk yöneticisi olarak çalışabilir. Her iki durumda da, kuyruk yöneticisinin yalnızca bir eşgörunümü çalışıyor, istekleri işliyor. Bu fark, çok eşgörunümlü bir kuyruk yöneticisi olarak çalışırken, kuyruk yöneticisinin etkin yönetim ortamını çalıştırmayan sunucu, etkin sunucu başarısız olursa, etkin yönetim ortamından otomatik olarak devralmaya hazır bir yedek eşgörunüm olarak çalışır.

Önce etkin olan yönetim ortamının etkin olduğu tek denetim ögesi, iki sunucuda kuyruk yöneticisini başlatma sırasındır. Kuyruk yöneticisi verilerine okuma/yazma kilitleri elde etmek için kullanılan ilk yönetim ortamı etkin yönetim ortamı olur.

Etkin yönetim ortamını, denetimi yedek sisteme aktarmak için yedek sisteme geçiş seçeneğini kullanarak, etkin eşgörünümü durdurarak diğer sunucuya takabilirsiniz.

Etkin olan QM1 yönetim ortamı, paylaşılan kuyruk yöneticisi verilerine dışlayıcı erişim ve dosya çalışırken günlük klasörlerini günlüğe kaydeder. The standby instance of QM1 detects when the active instance has failed, and becomes the active instance. It takes over the QM1 data and logs in the state they were left by the active instance, and accepts reconnections from clients and channels.

Etkin yönetim ortamı, beklemedeki yedekleme işleminin sona ermesine neden olan çeşitli nedenlerden dolayı başarısız olabilir:

- Etkin kuyruk yöneticisi yönetim ortamını bulandıran sunucu başarısız oldu.
- Etkin kuyruk yöneticisi yönetim ortamını ve dosya sistemini barındıran sunucu arasında bağlantı hatası.
- Kuyruk yöneticisi WebSphere MQ tarafından algılanan, kuyruk yöneticisi işlemlerinin yanıt verebilirliğini kaldırın.

Kuyruk yöneticisi yapılandırma bilgilerini birden çok sunucuya ekleyebilir ve etkin/yedek çift olarak çalışacak iki sunucunun da seçilmesini seçebilirsiniz. Toplam iki eşgörünümle ilgili bir sınır vardır. İki beklemedeki eşgörünümlere ve bir etkin yönetim ortamına sahip olamazsınız.

Çok eşgörünümlü kuyruk yöneticisi, yüksek kullanılabilirlik çözümünün bir bölümüdür. Yararlı bir yüksek kullanılabilirlik çözümü oluşturmak için bazı ek bileşenlere gereksinim duyarsınız.

- Client and channel reconnection to transfer WebSphere MQ connections to the computer that takes over running the active queue manager instance.
- Kilitleri doğru bir şekilde yöneten ve ortam ve dosya sunucusu hatasına karşı koruma sağlayan yüksek performanslı paylaşılan ağ dosya sistemi (NFS).

Önemli: NFS sürücüsünde bakım gerçekleştirebilmek için önce ortamınızda çalışan çok eşgörünümlü tüm kuyruk yöneticisi eşgörünümlerini durdurmalısınız. NFS hatası durumunda, kurtarılacak kuyruk yöneticisi yapılandırma yedeklemeniz olduğundan emin olun.

- Temel altyapıda tek hata noktalarını ortadan kaldırmak için esnek ağlar ve güç kaynakları.
- Yedek sisteme geçiş işlemini tolere eden uygulamalar Özellikle, işlemsel uygulamaların davranışlarına ve WebSphere MQ kuyruklarına göz atmanın uygulamalarıyla yakından ilgilenmeniz gerekir.
- Çalışmakta olduğundan emin olmak için etkin ve beklemedeki eşgörünümlerin izlenmesi ve yönetilmesi ve başarısız olan etkin eşgörünümleri yeniden başlatmaları. Çok eşgörünümlü kuyruk yöneticileri otomatik olarak yeniden başlatılmasına rağmen, yedekleme eşgörünümlerinizin çalıştığından, devralmaya hazır olduğundan ve başarısız olan eşgörünümlerin yeni yedek yönetim ortamları olarak yeniden çevrimiçi olarak geri getirildiğinden emin olmanız gerekir.

WebSphere MQ MQI istemcileri ve kanalları, etkinleştiginde otomatik olarak yedek kuyruk yöneticisine yeniden bağlanır. Yeniden bağlantı hakkında daha fazla bilgi ve yüksek kullanılabilirlikli bir çözümdeki diğer bileşenler ilgili konularda bulunabilir. Otomatik istemci yeniden bağlanma, Java için IBM WebSphere MQ sınıfları tarafından desteklenmez.

Desteklenen platformlar

You can create a multi-instance queue manager on any of the non-z/OS platforms from version 7.0.1.

Automatic client reconnection is supported for MQI clients from version 7.0.1 onwards.

Çok eşgörünümlü bir kuyruk yöneticisi yarat

Çok eşgörünümlü bir kuyruk yöneticisi yaratın, bir sunucuda kuyruk yöneticisi yaratılıyor ve başka bir sunucuda IBM WebSphere MQ yapılandırılıyor. Çok eşgörünümlü kuyruk yöneticileri, kuyruk yöneticisi verilerini ve günlüklerini paylaştı.

Çok eşgörünümli bir kuyruk yöneticisi yaratımında yer alan tüm çabalar, paylaşılan kuyruk yöneticisi verilerini ve günlük dosyalarını ayarlama görevidir. Ağ depolaması üzerinde paylaşılan dizinler oluşturulmalı ve bu dizinleri ağ paylaşımlarını kullanarak diğer sunucular için kullanılabilir duruma getirmelisiniz. Bu görevlerin, UNIX and Linux sistemlerinde *root* gibi bir yönetici yetkisi olan biri tarafından gerçekleştirilmesi gerekir. Adımlar aşağıdaki gibidir:

1. Veriler ve günlük dosyaları için paylaşımları oluşturun.
2. Bir sunucuda kuyruk yöneticisini yaratın.
3. Kuyruk yöneticisi yapılanış verilerini toplamak ve panoya kopyalamak için, ilk sunucuda **dspmqlnf** komutunu çalıştırın.
4. İkinci sunucuda kuyruk yöneticisi yapılanışını yaratmak için, kopyalanan verileri kullanarak **addmqinf** komutunu çalıştırın.

İkinci sunucuda kuyruk yöneticisini yeniden yaratmak için **crtmqm** komutunu çalıştırmayın.

Dosya erişimi denetimi

Diğer tüm sunucularda mqm adlı kullanıcının ve grubun paylaşımlara erişme iznine sahip olması dikkat çekmeniz gerekir.

On UNIX and Linux, you need to make the uid and gid of mqm the same on all the systems. You might need to edit /etc/passwd on each system to set a common uid and gid for mqm, and then reboot your system.

Microsoft Windows üzerinde, kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliği, kuyruk yöneticisi verilerini ve günlük dosyalarını içeren dizinlere tam denetim iznine sahip olmalıdır. İzni iki şekilde yapılandırabilirsiniz:

1. Alternatif güvenlik birincil kullanıcısı olarak bir genel gruba birlikte bir kuyruk yöneticisi yaratın. Genel grubu, kuyruk yöneticisi verilerini ve günlük dosyalarını içeren dizinlere tam denetim erişimine sahip olacak şekilde yetkilendirin; bkz. [“Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli” sayfa 365](#). Kuyruk yöneticisini çalıştıran kullanıcı kimliğini, genel grubun bir üyesi yapın. Yerel bir kullanıcıyı bir genel grubun üyesi yapamıyorsanız, kuyruk yöneticisi işlemlerinin bir etki alanı kullanıcı kimliği altında çalışması gerekir. The domain user ID must be a member of the local group mqm. [“Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümli kuyruk yöneticisi yarat” sayfa 341](#) görevi, bu şekilde güvenli kılınan dosyaları kullanarak çok eşgörünümli bir kuyruk yöneticisinin nasıl ayarlanmasını gösterir.
2. Etki alanı denetleyicide bir kuyruk yöneticisi yaratın; böylece, yerel mqm grubunun etki alanı kapsamı ("domain local"). Dosya paylaşımı etki alanı yerel mqm ile güvenli hale getirin ve aynı etki alanı yerel mqm grubu altındaki bir kuyruk yöneticisinin tüm eşgörünümünde kuyruk yöneticisi işlemlerini çalıştırın. The task, [“Etki alanı denetleyicilerinde çok eşgörünümli kuyruk yöneticisi yarat” sayfa 355](#), demonstrates how to set up a multi-instance queue manager using files secured in this way.

Yapılandırma bilgileri

Her bir sunucuyla ilgili IBM WebSphere MQ kuyruk yöneticisi yapılanış bilgilerini değiştirerek, gerek olduğu kadar kuyruk yöneticisi eşgörünümü yapılandırın. Her sunucu, uyumlu bir düzeltme düzeyinde kurulu IBM WebSphere MQ ürününün aynı sürümüne sahip olmalıdır. **dspmqlnf** ve **addmqinf** komutları, ek kuyruk yöneticisi eşgörünümünü yapılandırmanıza yardımcı olur. Diğer bir seçenek olarak, **mqqs.ini** ve **qm.ini** dosyalarını doğrudan düzenleyebilirsiniz. The topics, [“Linux üzerinde çok eşgörünümli bir kuyruk yöneticisi yarat” sayfa 376](#), [“Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümli kuyruk yöneticisi yarat” sayfa 341](#), and [“Etki alanı denetleyicilerinde çok eşgörünümli kuyruk yöneticisi yarat” sayfa 355](#) are examples showing how to configure a multi-instance queue manager.

Windows, UNIX and Linux sistemlerinde, tek bir **mqqs.ini** dosyasını, ağ paylaşımına yerleştirerek ve **AMQ_MQS_INI_LOCATION** ortam değişkenini işaret edecek şekilde ayarlayarak paylaşabilirsiniz.

Kısıtlamalar

1. Aynı kuyruk yöneticisinin birden çok örneğini yalnızca aynı işletim sistemine, mimariye ve endikasyonlarına sahip sunucularda yapılandırın. Örneğin, her iki makine de 32 bitlik ya da 64 bitlik olmalıdır.
2. All IBM WebSphere MQ installations must be at release level 7.0.1 or higher.
3. Genellikle, etkin ve beklemedeki kuruluşlar aynı bakım düzeyinde tutulur. Tüm kuruluşları birlikte büyütmemeniz gerekip gerekmediğini denetlemek için her yükseltme için bakım yönergelerine başvurun.
Etkin ve pasif kuyruk yöneticilerine ilişkin bakım düzeylerinin aynı olması gerektiğini unutmayın.
4. Kuyruk yöneticisi verilerini paylaşın ve yalnızca aynı IBM WebSphere MQ kullanıcı, grup ve erişim denetimi mekanizmasıyla yapılandırılan kuyruk yöneticileri arasında günlükler.
5. UNIX and Linux sistemlerinde, ağ üzerinde paylaşılan kütük sistemini `seft`, kesilebilir, bir yumuşak bağlama yerine, kesilebilir, sisteme bağlatılabilir bir sistemle yapılandırın. Bir sistem çağrısıyla kesintiye uğratılıncaya kadar, kuyruk yöneticisinin asılması zor bir kesilebilir bağlama işlemi. Yumuşak bağlamalar, sunucu hatasından sonra veri tutarlılığını garanti etmez.
6. Paylaşılan günlük ve veri dizinleri bir FAT ya da bir NFSv3 dosya sisteminde saklanamaz. Windows üzerinde çok eşgörünümlü kuyruk yöneticileri için, ağ üzerinde çalışan saklama alanına, Windows ağları tarafından kullanılan CIFS (Common Internet File System; Ortak İnternet Dosya Sistemi) protokolünün erişilmesi gerekir.

Windows etki alanları ve çok eşgörünümlü kuyruk yöneticileri

Windows üzerinde çok eşgörünümlü bir kuyruk yöneticisi, verilerinin ve günlüklerin paylaşılmasını gerektirir. Paylaşımın, kuyruk yöneticisinin farklı sunucularda ya da iş istasyonlarında çalışan tüm eşgörünümlerine erişebilmesi gerekir. Kuyruk yöneticilerini yapılandırın ve bir Windows etki alanının bir parçası olarak paylaşın. Kuyruk yöneticisi bir etki alanı iş istasyonunda ya da sunucuda ya da etki alanı denetleyicisinde çalışabilir.

Çok eşgörünümlü bir kuyruk yöneticisi yapılandırmadan önce, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimin nasıl denetleneceğini gözden geçirmek için “Secure unshared queue manager data and log directories and files on Windows” sayfa 368 ve “Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli” sayfa 365 bilgilerini okuyun. Konular eğitmandir; bir Windows etki alanında çok eşgörünümlü bir kuyruk yöneticisi için paylaşılan dizinleri doğrudan ayarlamaya gitmek isterseniz, bkz. “Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümlü kuyruk yöneticisi yarat” sayfa 341.

Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümlü kuyruk yöneticisi çalıştırma

Version 7.1' tan, çok eşgörünümlü kuyruk yöneticileri, bir etki alanının üyesi olan bir iş istasyonunda ya da sunucuda çalıştırılır. Version 7.1 öncesinde, çok eşgörünümlü kuyruk yöneticileri yalnızca etki alanı denetleyicilerinde çalıştırılır; bkz. “Etki alanı denetleyicilerinde çok eşgörünümlü kuyruk yöneticisi çalıştır” sayfa 341. Windows üzerinde çok eşgörünümlü bir kuyruk yöneticisini çalıştırmak için, aynı etki alanına bağlı aynı kuyruk yöneticisini çalıştıran bir etki alanı denetleyicisi, bir dosya sunucusu ve iki iş istasyonu ya da sunucu gereklidir.

Bir etki alanındaki herhangi bir sunucuda ya da iş istasyonunda çok eşgörünümlü kuyruk yöneticisi çalıştırılabilmesini sağlayan değişiklik, artık ek bir güvenlik grubuyla bir kuyruk yöneticisi oluşturabilmenizdir. Ek güvenlik grubu, -a parametresindeki `crtmqm` komutuna geçilir. Kuyruk yöneticisi verilerini içeren dizinleri güvenli bir şekilde ve grup ile birlikte günlüğe kaydetmenizi sağlar. Kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliği, bu grubun bir üyesi olmalıdır. Kuyruk yöneticisi dizinlere eriştiğinde, Windows , kullanıcı kimliğinin dizinlere erişmesi için gereken izinleri denetler. Hem grubu, hem de kullanıcı kimliği etki alanı kapsamını vererek, kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliğinin genel gruptan kimlik bilgileri vardır. Kuyruk yöneticisi farklı bir sunucuda çalışıyorsa, kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliği aynı kimlik bilgilerine sahip olabilir. Kullanıcı kimliği aynı olmak zorunda değil. Yerel `mqm` grubunun bir üyesi olarak, diğer güvenlik grubunun bir üyesi olması gerekir.

Çok eşgörünümli bir kuyruk yöneticisi yaratma görevi, bir değişiklikle birlikte Version 7.0.1 ' da olduğu gibi. Ek güvenlik grubu adını **crtmqm** komutunun değiştirgelerine eklemelisiniz. Görev, “Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümli kuyruk yöneticisi yarat” sayfa 341 içinde açıklanmaktadır.

Etki alanını ve etki alanı sunucularını ve iş istasyonlarını yapılandırmak için birden çok adım gereklidir. You must understand how Windows authorizes access by a queue manager to its data and log directories. Kuyruk yöneticisi işlemlerinin günlük ve veri dosyalarına erişme yetkisine sahip olduğundan emin değilseniz, “Secure unshared queue manager data and log directories and files on Windows” sayfa 368 konusunu okuyun. Bu konuda, gerekli adımları anlamanıza yardımcı olacak iki görev yer alır. Görevler şunlardır: “Yerel mqm grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması” sayfa 370 ve “Alternatif bir yerel güvenlik grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması” sayfa 373. Başka bir konu (“Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli” sayfa 365), kuyruk yöneticisi verileri içeren ve diğer güvenlik grubuyla birlikte günlük dosyaları içeren paylaşılan dizinlerin nasıl güvenli hale getirileceğini açıklar. Bu konuda dört görev yer alır; bir Windows etki alanı ayarlamak, dosya paylaşımı yaratmak, IBM WebSphere MQ for Windows' ı kurmak ve paylaşımı kullanmak için bir kuyruk yöneticisi yapılandırmak için. Görevler aşağıdaki gibidir:

1. “IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344.
2. “IBM WebSphere MQ ' un Windows etki alanındaki bir sunucuya ya da iş istasyonuna kurulması” sayfa 347.
3. “Kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan izin yaratılması” sayfa 350.
4. “Diğer bir genel güvenlik grubu tarafından yetkilendirilen paylaşılan verileri ve günlük dosyalarını okuma ve yazma” sayfa 353.

Daha sonra, etki alanını kullanarak görevi (“Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümli kuyruk yöneticisi yarat” sayfa 341) yapabilirsiniz. Bilginizi bir üretim etki alanına aktarmadan önce, çok eşgörünümli bir kuyruk yöneticisini ayarlamayı keşfetmek için bu görevleri gerçekleştirin.

Etki alanı denetleyicilerinde çok eşgörünümli kuyruk yöneticisi çalıştır

Version 7.0.1' ta, çok eşgörünümli kuyruk yöneticileri yalnızca etki alanı denetleyicilerinde çalıştırılırlar. Queue manager data could be secured with the domain mqm group. As the topic “Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli” sayfa 365 explains, you cannot share directories secured with the local mqm group on workstations or servers. Ancak, etki alanı denetleyicilerinde tüm grup ve birincil kullanıcıların etki alanı kapsamı vardır. IBM WebSphere MQ for Windows ürününü bir etki alanı denetleyicisine kursanız, kuyruk yöneticisi verileri ve günlük dosyaları, paylaşılabilen etki alanı mqm grubu ile güvenli kılınabilir. Follow the steps in the task, “Etki alanı denetleyicilerinde çok eşgörünümli kuyruk yöneticisi yarat” sayfa 355 to configure a multi-instance queue manager on domain controllers.

İlgili bilgiler

Windows 2000, Windows Server 2003 ve Windows Server 2008 küme düğümlerini etki alanı denetleyicileri olarak

Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümli kuyruk yöneticisi yarat

Örneğin, bir iş istasyonunda ya da Windows etki alanının bir parçası olan bir sunucuda Windows üzerinde çok eşgörünümli bir kuyruk yöneticisinin nasıl ayarlanacağı gösterilmektedir. Sunucu bir etki alanı denetleyicisi olmak zorunda değil. Kurulum, üretim ölçeği olmak yerine ilgili kavramları gösterir. Örnek, Windows Server 2008 'e dayalıdır. Adımlar, Windows Server 'ın diğer sürümlerinde değişiklik gösterebilir.

Bir üretim ölçeği yapılandırmasında, yapılandırmayı var olan bir etki alanına göre uyarlamak zorunda kalabilirsiniz. Örneğin, farklı paylaşımları yetkilendirmek için farklı etki alanı grupları tanımlayabilir ve kuyruk yöneticilerini çalıştıran kullanıcı kimliklerini gruplayabilirsiniz.

Örnek yapılandırma üç sunucudan oluşur:

sun

Bir Windows Server 2008 etki alanı denetleyicisi. *Sun, mars* ve *venus* içeren *wmq.example.com* etki alanına sahiptir. Şekil olarak, dosya sunucusu olarak da kullanılır.

mars

İlk IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörümlü kuyruk yöneticisinin bir eşgörümlüğünü içerir.

venus

İkinci IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörümlü kuyruk yöneticisinin ikinci örneğini içerir.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

Windows' ta, kuyruk yöneticisi verilerini depolamayı ve dosya günlük dosyalarını depolamayı planladığınız dosya sistemini doğrulamanız gerekmez. Denetleme yordamı, Paylaşılan dosya sistemi davranışı doğrulanıyor, UNIX and Linux için geçerlidir. Windows' ta denetimler her zaman başarılı olur.

Aşağıdaki görevlerdeki adımları gerçekleştirin. Görevler, etki alanı denetleyicisi ve etki alanını oluşturur, IBM WebSphere MQ for Windows ürününü bir sunucuya kurar ve veri ve günlük dosyaları için dosya paylaşımı oluşturur. Var olan bir etki alanı denetleyicisini yapılandırırıyorsanız, yeni bir Windows Server 2008 içindeki adımları denemek için yararlı olabilir. Adımlarınızı etki alanınıza uyarlayabilirsiniz.

1. [“IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344.](#)
2. [“IBM WebSphere MQ ' un Windows etki alanındaki bir sunucuya ya da iş istasyonuna kurulması” sayfa 347.](#)
3. [“Kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan izin yaratılması” sayfa 350.](#)
4. [“Diğer bir genel güvenlik grubu tarafından yetkilendirilen paylaşılan verileri ve günlük dosyalarını okuma ve yazma” sayfa 353.](#)

Bu görev hakkında

Bu görev, bir etki alanı denetleyicisini ve etki alanında iki sunucuyu bir kuyruk yöneticisinin eşgörümlüğünü çalıştırmak üzere yapılandırmak için kullanılan bir dizi görevlerden biridir. In this task you configure a second server, *venus*, to run another instance of the queue manager *QMGR*. Kuyruk yöneticisinin ikinci örneğini yaratmak için bu görevdeki adımları izleyin, *QMGR*' u ve çalıştığını sınavdım.

Bu görev, önceki bölümdeki dört görevden ayrıdır. Bu, tek bir eşgörümlü kuyruk yöneticisini çok eşgörümlü bir kuyruk yöneticisine dönüştüren adımları içerir. Diğer adımların tümü, tek ya da çok eşgörümlü kuyruk yöneticileri için ortaklardır.

Yordam

1. IBM WebSphere MQ for Windows programını çalıştırmak için ikinci bir sunucu yapılandırın.
 - a) Do the steps in the task [“IBM WebSphere MQ ' un Windows etki alanındaki bir sunucuya ya da iş istasyonuna kurulması” sayfa 347](#) to create a second domain server. Bu görev sırasında ikinci sunucunun adı *venus* olur.

İpucu: İki sunucunun her birindeki IBM WebSphere MQ için aynı kuruluş varsayılanlarını kullanarak ikinci kuruluşu yaratın. If the defaults differ, you might have to tailor the *Önek* and the *InstallationName* variables in the *QMGR QueueManager* stanza in the IBM WebSphere MQ configuration file *mq.s.ini*. Değişkenler, her bir sunucudaki her bir kuruluş ve kuyruk yöneticisi için farklılık gösterebilecek yollara başvurur. Yollar her sunucuda aynı kalırsa, çok eşgörümlü bir kuyruk yöneticisinin yapılandırılması daha basittir.

2. *venus* üzerinde ikinci bir *QMGR* yönetim ortamı yaratın.
 - a) *mars* üzerinde *QMGR* yoksa, bunu yaratmak için [“Diğer bir genel güvenlik grubu tarafından yetkilendirilen paylaşılan verileri ve günlük dosyalarını okuma ve yazma” sayfa 353](#) görevini yapın.

b) *venus* için Prefix ve InstallationName parametrelerinin doğru olup olmadığını denetleyin.

mars' ta **dspmqrinf** komutunu çalıştırın:

```
dspmqrinf QMGR
```

Sistem yanıtı:

```
QueueManager:  
  Name=QMGR  
  Directory=QMGR  
  Prefix=C:\Program Files\IBM\WebSphere MQ  
  DataPath=\\sun\wmq\data\QMGR  
  InstallationName=Installation1
```

c) **QueueManager** stanza 'nın makine tarafından okunabilir formunu panoya kopyalayın.

mars üzerinde, -o command parametresi ile **dspmqrinf** komutunu yeniden çalıştırın.

```
dspmqrinf -o command QMGR
```

Sistem yanıtı:

```
addmqrinf -s QueueManager -v Name=QMGR  
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"  
-v DataPath=\\sun\wmq\data\QMGR
```

d) On *venus* run the **addmqrinf** command from the clipboard to create an instance of the queue manager on *venus*.

Adjust the command, if necessary, to accommodate differences in the Önek or InstallationName parameters.

```
addmqrinf -s QueueManager -v Name=QMGR  
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"  
-v DataPath=\\sun\wmq\data\QMGR
```

WebSphere MQ configuration information added.

3. Start the queue manager *QMGR* on *venus*, permitting standby instances.

a) *mars* üzerindeki *QMGR* denetimi durdurulduğunu doğrulayın.

mars' ta **dspmqr** komutunu çalıştırın:

```
dspmqr -m QMGR
```

Sistem yanıtı, kuyruk yöneticisinin nasıl durdurulmasına bağlıdır; örneğin:

```
C:\Users\Administrator>dspmqr -m QMGR  
QMNAME(QMGR) STATUS(Ended immediately)
```

b) *venus* 'ta **strmqm** komutunu çalıştırmak için *QMGR* ' un izin verme sıralarını çalıştırın:

```
strmqm -x QMGR
```

Sistem yanıtı:

```
WebSphere MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

Sonuçlar

Çok eşgörünümlü kuyruk yöneticisi anahtarlarının üzerine geçiş yapmak için aşağıdaki adımları gerçekleştirin:

1. *mars*'ta **strmqm** komutunu çalıştırın; *QMGR* ' un izin verme sıralarını başlatın:

```
strmqm -x QMGR
```

Sistem yanıtı:

```
WebSphere MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
A standby instance of queue manager 'QMGR' has been started.  
The active instance is running elsewhere.
```

2. *venus* üzerinde **endmqm** komutunu çalıştırın:

```
endmqm -r -s -i QMGR
```

venus üzerindeki sistem yanıtı:

```
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ended, permitting switchover to  
a standby instance.
```

Ve *mars*' ta:

```
dspmq  
QMNAME(QMGR) STATUS(Running as standby)  
C:\Users\wmquser2>dspmq  
QMNAME(QMGR) STATUS(Running as standby)  
C:\Users\wmquser2>dspmq  
QMNAME(QMGR) STATUS(Running)
```

Sonraki adım

Örnek programları kullanarak çok eşgörünümlü bir kuyruk yöneticisini doğrulamak için; bkz. [“Pencereler üzerinde çok eşgörünümlü kuyruk yöneticisini doğrulayın” sayfa 363.](#)

IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması

This task creates the domain *wmq.example.com* on a Windows 2008 domain controller called *sun*. Etki alanındaki Domain *mqm* genel grubunu, doğru haklarla ve bir kullanıcıyla yapılandırır.

Bir üretim ölçeği yapılandırmasında, yapılandırmayı var olan bir etki alanına göre uyarlamak zorunda kalabilirsiniz. Örneğin, farklı paylaşımları yetkilendirmek için farklı etki alanı grupları tanımlayabilir ve kuyruk yöneticilerini çalıştıran kullanıcı kimliklerini gruplayabilirsiniz.

Örnek yapılandırma üç sunucudan oluşur:

sun

Bir Windows Server 2008 etki alanı denetleyicisi. *Sun*, *mars* ve *venus* üzerinden *wmq.example.com* etki alanına sahiptir. Şekil olarak, dosya sunucusu olarak da kullanılır.

mars

İlk IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörünümlü kuyruk yöneticisinin bir eşgörünümünü içerir.

venus

İkinci IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, QMGRadlı çok eşgörsümüli kuyruk yöneticisinin ikinci örneğini içerir.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

1. Görev adımları, kurulu olan ancak herhangi bir rolle yapılandırılmamış bir Windows Server 2008 ile tutarlıdır. Var olan bir etki alanı denetleyicisini yapılandırıyorsanız, yeni bir Windows Server 2008 içindeki adımları denemek için yararlı olabilir. Adımlarınızı etki alanınıza uyarlayabilirsiniz.

Bu görev hakkında

Bu görevde, yeni bir etki alanı denetleyicisinde bir Active Directory ve DNS etki alanı oluşturursun. You then configure it ready to install IBM WebSphere MQ on other servers and workstations that join the domain. Follow the task if you are unfamiliar with installing and configuring Active Directory to create a Windows domain. Çok eşgörsümüli bir kuyruk yöneticisi yapılışı yaratmak için bir Windows etki alanı yaratmalısınız. Bu görev, bir Windows etki alanını yapılandırmak için en iyi şekilde size yol gösterecek şekilde tasarlanmamaktadır. Bir üretim ortamında çok eşgörsümüli kuyruk yöneticilerini konuşlandırmak için Windows belgelerine bakmanız gerekir.

Görev sırasında aşağıdaki adımları gerçekleştirin:

1. Active Directory ürününü kurun.
2. Etki alanı ekle.
3. Etki alanını DNS ' ye ekleyin.
4. Create the global group Domain mqm and give it the correct rights.
5. Add a user and make it a member of the global group Domain mqm.

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, [“Windows etki alanları ve çok eşgörsümüli kuyruk yöneticileri”](#) sayfa 340.

Görev için etki alanı denetleyicisi anasistem adı *sun*, iki IBM WebSphere MQ sunucusu ise *mars* ve *venus* olarak adlandırılır. Etki alanının adı *wmq.example.com*. Görevdeki tüm italik adların yerine kendi seçtiğiniz adlara sahip olabilirsiniz.

Yordam

1. Yerel ya da Workgroup yöneticisi olarak, etki alanı denetleyicide *sun* oturum açın.
Sunucu etki alanı denetleyicisi olarak önceden yapılandırıldıysa, etki alanı yöneticisi olarak oturum açmanız gerekir.
2. Active Directory Etki Alanı Hizmetleri sihirbazını çalıştırın.
 - a) **Başlat > Çalıştır ...** seçeneklerini tıklatın. *dcprmo* yazın ve **Tamam'** i tıklatın.
If the Active Directory binary files are not already installed, Windows installs the files automatically.
3. Sihirbazın ilk penceresinde, **Gelişmiş kip kuruluşunu kullan** onay kutusunun işaretini kaldırın. **İleri > İleri** 'yi ve sonra **Yeni bir ormanda yeni bir etki alanı oluştur > İleri** 'yi tıklatın.
4. **Orman kök etki alanının FQDN değeri** alanına *wmq.example.com* yazın. **İleri** 'yi tıklatın.
5. Forest Functional Level (Ormanı İşlevsel Düzey Ayarla) penceresinde, **Forest işlevsel düzeyler > Sonrakilistesinden Windows Server 2003** ya da sonraki bir sürümü seçin.
IBM WebSphere MQ tarafından desteklenen en eski Windows Server düzeyi Windows Server 2003 'tür.
6. İsteğe bağlı: Etki Alanı İşlevsel Düzeyi Ayarla penceresinde, **Etki alanı işlevsel düzeyleri > İleri** listesinden **Windows Server 2003** ya da sonraki bir sürümü seçin.

Bu adım yalnızca, Forest Functional Level 'ı **Windows Server 2003** olarak ayarladığınızda gereklidir.

7. Ek Etki Alanı Denetleyicisi Seçenekleri penceresi açılır ve **DNS sunucusu** seçeneği ek bir seçenek olarak seçilir. Uyarı penceresini temizlemek için **Next** (İleri) ve **Yes** (Evet) seçeneklerini tıklatın.

İpucu: Bir DNS sunucusu önceden kurulduysa, bu seçenek size sunulmaz. Bu görevi tam olarak izlemek istiyorsanız, bu etki alanı denetleyicisinden tüm rolleri kaldırın ve yeniden başlatın.

8. Leave the Database, Log Files, and SYSVOL directories unchanged; click **Sonraki**.
9. Directory Services Restore Mode Administrator Password penceresindeki **Password** (Parola) ve **Confirm password** (Parolayı onaylayın) alanlarına bir parola yazın. **İleri** > **İleri** 'yi tıklatın. Son sihirbaz penceresinde **Tamamlamayı yeniden başlat** seçeneğini belirleyin.
10. When the domain controller reboots, log on as *wmq\Administrator*.

Sunucu yöneticisi otomatik olarak başlatılır.

11. *wmq.example.com*\Users klasörünü açın

- a) **Server Manager** > **Roller** > **Active Directory Etki Alanı Hizmetleri** > *wmq.example.com* > **Kullanıcılar** seçeneklerini açın.

12. **Kullanıcılar** > **Yeni** > **Grup** öğelerini sağ tıklatın.

- a) **Grup adı** alanına bir grup adı yazın.

Not: Tercih edilen grup adı Domain *mqm*. Tam olarak gösterildiği gibi yazın.

- Domain *mqm* adlı grubu çağırmak, "Hazırlama IBM WebSphere MQ" sihirbazının davranışını bir etki alanı iş istasyonuna ya da sunucuya göre değiştirmektedir. It causes the "Hazırlama IBM WebSphere MQ" wizard automatically to add the group Domain *mqm* to the local *mqm* group on each new installation of IBM WebSphere MQ in the domain.
- İş istasyonlarını ya da sunucuları, Domain *mqm* genel grubu olmayan bir etki alanına kurabilirsiniz. Bunu yapmazsanız, Domain *mqm* grubu ile aynı özellikleri taşıyan bir grup tanımlamanız gerekir. Bu grubu ya da üyesi olan kullanıcıları, yerel *mqm* grubunun üyeleri, IBM WebSphere MQ ' in bir etki alanında kurulu olduğu her yerde gerçekleştirmelisiniz. Etki alanı kullanıcılarını birden çok gruba yerleştirebilirsiniz. Ayrı olarak yönetmek istediğiniz bir kuruluş kümesine karşılık gelen her bir grup için birden çok etki alanı grubu oluşturun. Etki alanı kullanıcılarını, yönettikleri kuruluşlara göre, farklı etki alanı gruplarına göre bölün. Her etki alanı grubunu ya da gruplarını, farklı IBM WebSphere MQ kuruluşlarına ilişkin yerel *mqm* grubuna ekleyin. Yalnızca belirli bir yerel *mqm* grubunun üyesi olan etki alanı gruplarındaki etki alanı kullanıcıları, bu kuruluş için kuyruk yöneticilerini oluşturabilir, yönetebilir ve çalıştırabilir.
- Bir etki alanındaki bir iş istasyonuna ya da sunucuya IBM WebSphere MQ kurulurken aday gösterdiğiniz etki alanı kullanıcısı, Domain *mqm* grubunun bir üyesi ya da Domain *mqm* grubu ile aynı özelliklerle tanımladığınız bir alternatif grubun üyesi olmalıdır.

- b) Leave **Genel** clicked as the **Grup kapsamı**, or change it to **Evrensel**. Leave **Güvenlik** clicked as the **Grup tipi. Tamam** 'ı tıklatın.

13. Hakları, **İzin Ver Grup üyeliğini okuma** ve **İzin Ver Read groupMembershipSAM** haklarını Domain *mqm* genel grubunun haklarına ekleyin.

- a) Server Manager işlem çubuğunda **Görünüm** > **Gelişmiş özellikler** seçeneklerini tıklatın.

- b) Server Manager dolaşma ağacında **Users**(Kullanıcılar) öğesini tıklatın.

- c) Users (Kullanıcılar) penceresinde **Domain mqm** > **Properties**(Etki Alanı *mqm*) seçeneğini sağ tıklatın.

- d) **Güvenlik** > **Gelişmiş** > **Ekle ...** seçeneklerini tıklatın. Domain *mqm* yazın ve **Adları denetle** > **Tamam** düğmesini tıklatın.

The **Ad** field is prefilled with the string, Domain *mqm* (*domain name*\Domain *mqm*).

- e) **Özellikler** 'i tıklatın. **Uygula** listesinde, listenin alt kısmındaki **Alt Öğe Kullanıcı Nesneleri** ' yi seçin.

- f) **İzinler** listesinden, **Grup üyeliğini okuma** ve **Read groupMembershipSAM İzin Ver** onay kutularını seçin; **Tamam** > **Uygula** > **Tamam** > **Tamam** seçeneğini tıklatın.

14. Domain *mqm* genel grubuna iki ya da daha fazla kullanıcı ekleyin.

One user, *wmquser1* in the example, runs the IBM WebSphere MQ service, and the other user, *wmquser2*, is used interactively.

Etki alanı yapılışındaki alternatif güvenlik grubunu kullanan bir kuyruk yöneticisi yaratmak için etki alanı kullanıcısı gereklidir. Denetimci **crtmqm** komutunu çalıştırma yetkisine sahip olsa da, kullanıcı kimliğinin bir denetimci olması yeterli değildir. Yönetici olabilen etki alanı kullanıcısı, diğer güvenlik grubunun yanı sıra yerel **mqm** grubunun da bir üyesi olmalıdır.

Örnekte, Domain **mqm** genel grubunun *wmquser1* ve *wmquser2* üyelerini gerçekleştirmenizi sağlar. "Prepare IBM WebSphere MQ" sihirbazı, Domain **mqm** konfigürasyonunu, sihirbazın çalıştırıldığı yerel **mqm** grubunun bir üyesi olarak otomatik olarak yapılandırır.

Tek bir bilgisayarda her IBM WebSphere MQ kuruluşu için IBM WebSphere MQ hizmetini çalıştırmak için farklı bir kullanıcı sağlamanız gerekir. Aynı kullanıcıları farklı bilgisayarlardaki kullanıcıları yeniden kullanabilirsiniz.

- a) Server Manager dolaşma ağacında **Kullanıcılar > Yeni > Kullanıcı** öğelerini tıklayın.
 - b) Yeni Nesne-Kullanıcı penceresinde, **Kullanıcı oturum açma adı** alanına *wmquser1* yazın. Type *WebSphere* into the **Adı** field, and *MQ1* into the **Soyadı** field. **İleri**'yi tıklayın.
 - c) **Password** (Parola) ve **Confirm password** (Parolayı doğrulayın) alanlarına bir parola yazın ve **User musive change password at next logon** (Sonraki oturum açma sırasında parolayı değiştirmeli) onay **İleri > Sondüğmesini** tıklayın.
 - d) Users (Kullanıcılar) penceresinde, **WebSphere MQ > Add to a group ...**(Bir gruba ekle ...) seçeneğini sağ tıklayın. Domain **mqm** yazın ve **Adları Denetle > Tamam > Tamamdüğmesini** tıklayın.
 - e) Repeat steps a to g to add *WebSphere MQ2* as *wmquser2*.
15. IBM WebSphere MQ hizmeti hizmet olarak çalıştırılıyor.

If you need to run IBM WebSphere MQ as a service, and then give the domain user (that you obtained from your domain administrator) the right to run as a service, carry out the following procedure:

- a) **Başlat > Çalıştır ...**seçeneklerini tıklayın.
secpol.msc komutunu yazın ve **OK**(Tamam) düğmesini tıklayın.
- b) Open **Güvenlik Ayarları > Yerel İlkeler > Kullanıcı Hakları Atamaları**.
İlkeler listesinde, **Hizmet olarak oturum aç > Özellikler**seçeneğini sağ tıklayın.
- c) **Kullanıcı ya da Grup Ekle ...**düğmesini tıklayın.
Etki alanı denetimcinizden edindiğiniz kullanıcının adını yazın ve **Adları Denetle**düğmesini tıklayın.
- d) Bir Windows Güvenlik penceresi tarafından istenirse, yeterli yetkiye sahip bir kullanıcı ya da hesap kullanıcısının kullanıcı adını ve parolasını yazın ve **Tamam > Uygula > Tamam**seçeneğini tıklayın.
Yerel Güvenlik İlkesi penceresini kapatın.

Not: Windows Vista ve Windows Server 2008 'de User Account Control (UAC) varsayılan olarak etkindir.

UAC özelliği, Administrators (Yöneticiler) grubunun üyeleri olsalar da, kullanıcıların belirli işletim sistemi tesislerinde gerçekleştirebileceği işlemleri kısıtlar. Bu kısıtlamanın üstesinden gelmek için uygun adımları atmanız gerekir.

Sonraki adım

Sonraki göreve ("[IBM WebSphere MQ ' un Windows etki alanındaki bir sunucuya ya da iş istasyonuna kurulması](#)" sayfa 347) devam edin.

IBM WebSphere MQ ' un Windows etki alanındaki bir sunucuya ya da iş istasyonuna kurulması

Bu görevde, IBM WebSphere MQ ürününü bir sunucuya ya da iş istasyonuna *wmq.example.com* Windows etki alanında kurmanızı ve yapılandırmanızı sağlar.

Bir üretim ölçeği yapılandırmasında, yapılandırmayı var olan bir etki alanına göre uyarlamak zorunda kalabilirsiniz. Örneğin, farklı paylaşımları yetkilendirmek için farklı etki alanı grupları tanımlayabilir ve kuyruk yöneticilerini çalıştıran kullanıcı kimliklerini gruplayabilirsiniz.

Örnek yapılandırma üç sunucudan oluşur:

sun

Bir Windows Server 2008 etki alanı denetleyicisi. *Sun*, *mars* ve *venus* için *wmq.example.com* etki alanına sahiptir. Şekil olarak, dosya sunucusu olarak da kullanılır.

mars

İlk IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, QMGRadlı çok eşgörümlü kuyruk yöneticisinin bir eşgörümlü kuyruk yöneticisini içerir.

venus

İkinci IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, QMGRadlı çok eşgörümlü kuyruk yöneticisinin ikinci örneğini içerir.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

1. Do the steps in “IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344 to create a domain controller, *sun*, for the domain *wmq.example.com*. Yapılanışınıza uygun olarak, italik olarak adları değiştirin.
2. See [Windows sistemleri üzerinde donanım ve yazılım gereksinimleri](#) for other Windows versions you can run IBM WebSphere MQ on.

Bu görev hakkında

Bu görevde, *wmq.example.com* etki alanının bir üyesi olarak *mars* adlı bir Windows Server 2008 'i yapılandırmanız. IBM WebSphere MQ' u kurun ve kuruluşu, *wmq.example.com* etki alanının bir üyesi olarak çalışacak şekilde yapılandırın.

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, “[Windows etki alanları ve çok eşgörümlü kuyruk yöneticileri](#)” sayfa 340.

Görev için etki alanı denetleyicisi anasistem adı *sun*, iki IBM WebSphere MQ sunucusu ise *mars* ve *venus* olarak adlandırılır. Etki alanının adı *wmq.example.com*. Görevdeki tüm italik adların yerine kendi seçtiğiniz adlara sahip olabilirsiniz.

Yordam

1. Etki alanı denetleyicisini, *sun.wmq.example.com* 'yi bir DNS sunucusu olarak *mars* ' e ekleyin.
 - a) *mars* üzerinde, *mars\Administrator* olarak oturum açın ve **Start**(Başlat) ögesini tıklayın.
 - b) **Ağ > Özellikler > Ağ bağlantılarını yönet** seçeneğini sağ tıklayın.
 - c) Ağ bağdaştırıcısını farenin sağ düğmesiyle tıklayın ve **Properties**(Özellikler) seçeneğini belirleyin.

Sistem, bağlantı kullandığı öğeleri listeleyen Yerel Ağ Bağlantısı Özellikleri pencere ile yanıt verir.
 - d) Yerel Ağ Bağlantısı Özellikleri penceresindeki öğeler listesinden **Internet Protocol Sürüm 4** ya da **Internet Protocol Sürüm 6** ' yı seçin. **Özellikler > Gelişmiş ...** seçeneğini tıklayın. ve **DNS** sekmesini tıklayın.
 - e) DNS sunucusu adresleri altında, **Ekle ...** düğmesini tıklayın.
 - f) DNS sunucusu da olan etki alanı denetleyicisinin IP adresini yazın ve **Ekle** düğmesini tıklayın.
 - g) **Bu DNS soneklerini ekle > Ekle ...** düğmesini tıklayın.
 - h) *wmq.example.com* yazın ve **Ekle** düğmesini tıklayın.
 - i) **Bu bağlantı için DNS soneki** alanına *wmq.example.com* yazın.
 - j) **Bu bağlantının adresini DNS ' de kaydet** ve **DNS kaydındaki bu bağlantının sonekini kullan** seçeneğini belirleyin. **Tamam > Tamam > Kapat** seçeneğini tıklayın.

- k) Bir komut penceresi açın ve TCP/IP ayarlarını gözden geçirmek için **ipconfig /all** komutunu yazın.
2. *mars*' ta, bilgisayarı *wmq.example.com* etki alanına ekleyin.
- Başlat**düğmesini tıklayın.
 - Bilgisayar > Özellikler**seçeneğini sağ tıklayın. Bilgisayar adı, etki alanı ve çalışma grubu ayarları bölümünde **Ayarları değiştir**seçeneğini tıklayın.
 - Sistem Özellikleri pencerelerinde **Değiştir ...**düğmesini tıklayın.
 - Etki alanı öğesini tıklayın, *wmq.example.com* yazın ve **Tamam**' ı tıklayın.
 - Etki alanı denetleyicisi yöneticisinin **Kullanıcı adı** ve **Parola** alanlarını, bilgisayarın etki alanına katılmasına izin verme yetkisine sahip olan ve **Tamam**' ı tıklayın.
 - "*wmq.example.com* etki alanına hoş geldiniz" iletisine yanıt olarak **Tamam > Tamam > Kapat > Şimdi Yeniden Başlat** simgesini tıklayın.
3. Bilgisayarın *wmq.example.com* etki alanının bir üyesi olup olmadığını denetleyin.
- sun*' ta, etki alanı denetleyicide *wmq\Administrator* olarak oturum açın.
 - Open **Sunucu Yöneticisi > Active Directory Etki Alanı Hizmetleri > wmq.example.com > Bilgisayarlar** and check *mars* is listed correctly in the Computers window.
4. *mars* üzerine IBM WebSphere MQ for Windows kurun.

IBM WebSphere MQ for Windows kuruluş sihirbazını çalıştırma hakkında daha fazla bilgi için; bkz. [Installing IBM WebSphere MQ server on Windows](#) .

- mars*' ta yerel yönetici olarak oturum açın: *mars\Administrator*.
 - IBM WebSphere MQ for Windows kuruluş ortamında **Setup** komutunu çalıştırın.
IBM WebSphere MQ Launchpad uygulaması başlar.
 - Önkoşul yazılımların kurulu olup olmadığını denetlemek için **Yazılım Gereksinimleri** seçeneğini tıklayın.
 - Etki alanı kullanıcı kimliğini yapılandırmak için **Ağ Yapılandırması > Evet** seçeneklerini tıklayın.
The task, "IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması" sayfa 344, configures a domain user ID for this set of tasks.
 - Click **WebSphere MQ Kuruluşu**, select an installation language and click Launch IBM WebSphere MQ Installer.
 - Lisans sözleşmesini onaylayın ve varsayılan yapılandırmayı kabul etmek için **Next > Next > Install** (İleri-> Kur) seçeneklerini tıklayın. Kuruluşun tamamlanmasını bekleyin ve **Finish**(Son) düğmesini tıklayın.
Kuruluşun adını değiştirmek, farklı bileşenleri kurmak, kuyruk yöneticisi verileri ve günlükler için farklı bir dizin yapılandırmak ya da farklı bir dizine kurmak isteyebilirsiniz. Böyle bir durumda, **Tipik** yerine **Özel** öğesini tıklayın.
IBM WebSphere MQ kuruludur ve kuruluş programı "Hazırlama IBM WebSphere MQ" sihirbazını başlatır.
Önemli: Sihirbazı henüz çalıştırmayın.
5. Configure the user that is going to run the IBM WebSphere MQ service with the **Hizmet olarak çalıştır** right.
- Yerel *mqm* grubunun, *Domain mqm* grubunun ya da IBM WebSphere MQ hizmetini çalıştıracığı kullanıcının, doğru olarak yapılandırılıp yapılandırılmayacağını seçin. Örnekte, kullanıcıya doğru değeri verirsiniz.
- Başlat > Çalıştır ...** seçeneklerini tıklayın, **secpol.msc** komutunu girin ve **Tamam**' ı tıklayın.
 - Güvenlik Ayarları > Yerel İlkeler > Kullanıcı Hakları Atamaları**' ı açın. İlkeler listesinde, **Hizmet olarak oturum aç > Özellikler** seçeneğini sağ tıklayın.

- c) **Kullanıcı ya da Grup Ekle ...**düğmesini tıklatın. ve *wmquser1* yazın ve **Adları Denetle** seçeneğini tıklatın.
- d) Bir etki alanı yöneticisinin kullanıcı adını ve parolasını yazın (*wmq\Administrator*) ve **OK > Apply > OK**(Tamam-> Uygula-> Tamam) seçeneklerini tıklatın. Yerel Güvenlik İlkesi penceresini kapatın.
6. "PreparePrepare IBM WebSphere MQ" sihirbazını çalıştırın.

"Hazırlama IBM WebSphere MQ" sihirbazını çalıştırma hakkında daha fazla bilgi için; bkz. [Prepare WebSphere MQ sihirbazı ile WebSphere MQ uygulamasını yapılandırma](#) .

- a) IBM IBM WebSphere MQ Installer, "Prepare IBM WebSphere MQ" programını otomatik olarak çalıştırır.

To start the wizard manually, find the shortcut to the "Hazırlama IBM WebSphere MQ" in the **Başlat > Tüm programlar > IBM WebSphere MQ** folder. Çoklu kuruluş yapılandırmasındaki IBM WebSphere MQ kuruluşuna karşılık gelen kısayolu seçin.

- b) Click **Sonraki** and leave **Evet** clicked in response to the question "Ağda bir Windows 2000 ya da daha sonraki bir etki alanı denetleyicisi olup olmadığını belirleyin".
- c) Click **Evet > Sonraki** in the first Configuring IBM WebSphere MQ for Windows for Windows domain users window.
- d) In the second Configuring IBM WebSphere MQ for Windows for Windows domain users window, type *wmq* in the **Etki Alanı** field. **Kullanıcı adı** alanına *wmquser1* yazın ve **Parola** alanında bir değer ayarladıysanız, parolayı yazın. **İleri**'yi tıklatın.

Sihirbaz, IBM IBM WebSphere MQ ürününü *wmquser1* ile yapılandırır ve başlatır.

- e) Sihirbazın son sayfasında, istediğiniz şekilde onay kutularını işaretleyin ya da kutunun işaretini kaldırın ve **Son**düğmesini tıklatın.

Sonraki adım

1. Do the task, "[Yerel mqm grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması](#)" sayfa 370, to verify that the installation and configuration are working correctly.
2. Bir dosya paylaşımının konfigürasyonunu, çok eşgörümlü bir kuyruk yöneticisinin veri ve günlük kütüklerini saklamak için yapılandırmak için "[Kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan dizin yaratılması](#)" sayfa 350 görevini yapın.

İlgili kavramlar

[Bir WebSphere MQ Windows Hizmeti için gereken kullanıcı hakları](#)

Kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan dizin yaratılması

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir.

Bir üretim ölçeği yapılandırmasında, yapılandırmayı var olan bir etki alanına göre uyarlamak zorunda kalabilirsiniz. Örneğin, farklı paylaşımları yetkilendirmek için farklı etki alanı grupları tanımlayabilir ve kuyruk yöneticilerini çalıştıran kullanıcı kimliklerini gruplayabilirsiniz.

Örnek yapılandırma üç sunucudan oluşur:

sun

Bir Windows Server 2008 etki alanı denetleyicisi. *Sun, marsve venus* çeren *wmq.example.com* etki alanına sahiptir. Şekil olarak, dosya sunucusu olarak da kullanılır.

mars

İlk IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörümlü kuyruk yöneticisinin bir eşgörümlü içerir.

venus

İkinci IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörümlü kuyruk yöneticisinin ikinci örneğini içerir.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

1. To do this task exactly as documented, do the steps in the task, “[IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması](#)” sayfa 344, to create the domain *sun.wmq.example.com* on the domain controller *sun*. Yapılanışınıza uygun olarak, italik olarak adları değiştirin.

Bu görev hakkında

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, “[Windows etki alanları ve çok eşgözümlü kuyruk yöneticileri](#)” sayfa 340.

Bu görevde, bir veri ve günlük dizini içeren bir paylaşım ve paylaşım erişim yetkisi vermek için bir genel grup oluşturursun. You pass the name of the global group that authorizes the share to the **crtmqm** command in its -a parameter. Genel grup, bu paylaşımın kullanıcılarını diğer paylaşımların kullanıcılarından ayırmanın esnekliğini size sunar. Bu esnekliğe gereksinim duyarsanız, paylaşım yeni bir genel grup yaratmak yerine Domain **mqm** grubuyla yetki verir.

Bu görevde paylaşım için kullanılan genel grup *wmqha* olarak adlandırılır ve paylaşım *wmq* adı verilir. They are defined on the domain controller *sun* in the Windows domain *wmq.example.com*. Paylaşım, *wmqha* adlı genel grup için tam denetim izinleri var. Görevdeki italikleştirilmiş adları, seçtiğiniz adlarla değiştirin.

Bu görevin amaçlarına göre, etki alanı denetleyicisi, dosya sunucusuyla aynı sunucudur. Pratik uygulamalarda, izin ve dosya hizmetlerini, performans ve kullanılabilirlik için farklı sunucular arasında bölüşürün.

Kuyruk yöneticisinin çalışmakta olduğu kullanıcı kimliğini, iki grubun üyesi olacak şekilde yapılandırmanız. Bu, IBM WebSphere MQ sunucusundaki ve *wmqha* genel grubunun yerel **mqm** grubunun bir üyesi olmalıdır.

In this set of tasks, when the queue manager is running as a service, it runs under the user ID *wmquser1*, so *wmquser1* must be a member of *wmqha*. When the queue manager is running interactively, it runs under the user ID *wmquser2*, so *wmquser2* must be a member of *wmqha*. Both *wmquser1* and *wmquser2* are members of the global group Domain **mqm**. Domain **mqm** is a member of the local **mqm** group on the *mars* and *venus* IBM WebSphere MQ servers. Bu nedenle, *wmquser1* ve *wmquser2*, hem IBM WebSphere MQ sunucularında yerel **mqm** grubunun üyeleridir.

Yordam

1. Etki alanı denetleyicide oturum açın, etki alanı yöneticisi olarak *sun.wmq.example.com* 'ı açın.
2. Create the global group *wmqha*.
 - a) **Server Manager > Roller > Active Directory Etki Alanı Hizmetleri > wmq.example.com > Kullanıcılar** seçeneklerini açın.
 - b) *wmq.example.com\Users* klasörünü açın
 - c) **Kullanıcılar > Yeni > Grup** öğelerini sağ tıklayın.
 - d) **Grup adı** alanına *wmqha* yazın.
 - e) Leave **Genel** clicked as the **Grup kapsamı** and **Güvenlik** as the **Grup tipi**. **Tamam**'ı tıklayın.
3. Add the domain users *wmquser1* and *wmquser2* to the global group, *wmqha*.
 - a) Server Manager gezinme ağacında **Kullanıcılar** seçeneğini tıklayın ve kullanıcı listesinde **wmqha > Properties** (Özellikler) seçeneğini sağ tıklayın.
 - b) *wmqha* Properties (Özellikler) penceresinde **Üyeler** etiketini tıklayın.
 - c) **Ekle ...** düğmesini tıklayın. *wmquser1; wmquser2* yazın ve **Adları Denetle > Tamam > Uygula > Tamam** seçeneklerini tıklayın.

4. Kuyruk yöneticisi verilerini ve günlük dosyalarını içerecek dizin ağacını yaratın.

- Bir komut istemi açın.
- Komutu yazın:

```
md c:\wmq\data , c:\wmq\logs
```

5. Authorize the global group *wmqha* to have full control permission to the *c:\wmq* directories and share.

- Windows Explorer 'da **c:\wmq > Properties**(Özellikler) seçeneğini sağ tıklayın.
- Güvenlik** sekmesini tıklayın ve **Gelişmiş > Düzenle ...**seçeneklerini tıklayın.
- Bu nesnenin iyesinden devralınabilir izinler ekle** onay kutusunun işaretini kaldırın. Windows Güvenliği penceresinde **Kopyala** ögesini tıklayın.
- İzin girişleri** listesinde Kullanıcılar için satırları seçin ve **Kaldır**düğmesini tıklayın. SYSTEM, Administrators ve CREATOR OWNER satırlarını **İzin girişleri**listesinde bırakın.
- Ekle ...**düğmesini tıklayın. and type the name of the global group *wmqha*. **Adları Denetle > Tamam**düğmesini tıklayın.
- wmq* için İzin Girdisi penceresinde, **İzinler**listesinde **Tam Denetim** ögesini seçin.
- Tamam > Uygula > Tamam > Tamam > Tamam**ögesini tıklayın
- Windows Explorer 'da **c:\wmq > Paylaş** ...seçeneğini sağ tıklayın.
 - İleri Düzey Paylaşım ...**düğmesini tıklayın. ve **Bu klasörü paylaş** onay kutusunu seçin. Paylaşım adını *wmq*olarak bırakın.
 - İzinler > Ekle ...**öğelerini tıklayın. and type the name of the global group *wmqha*. **Adları Denetle > Tamam**düğmesini tıklayın.
- Grup ya da kullanıcı adları**listesinde *wmqha* ögesini seçin. **wmqha için izinler**listesinden **Tam Denetim** onay kutusunu seçin; **Uygula**seçeneğini tıklayın.
- Grup ya da kullanıcı adları**listesinde *Administrators* ögesini seçin. **Denetimcileri için izinler**listesinden **Tam Denetim** onay kutusunu seçin; **Uygula > Tamam > Tamam > Kapat**seçeneğini tıklayın.

Sonraki adım

IBM WebSphere MQ sunucularının her birinden paylaşılan dizinleri okuyabilir ve bu dizinlere dosya yazabildiğinizi denetleyin. IBM IBM WebSphere MQ hizmet kullanıcı kimliği (*wmquser1*) ve etkileşimli kullanıcı kimliği (*wmquser2*) ile ilgili bilgileri denetleyin.

1. If you are using remote desktop, you must add *wmq\wmquser1* and *wmquser2* to the local group Remote Desktop Users on *mars*.

- mars* ' ta *wmq\Administrator*olarak oturum açma
- Yerel Kullanıcılar ve Gruplar penceresini açmak için **lusrmgr.msc** komutunu çalıştırın.
- Gruplar**seçeneğini tıklayın. **Uzak Masaüstü Kullanıcıları > Özellikler > Ekle ...**seçeneğini sağ tıklayın. *wmquser1*; *wmquser2* yazın ve **Adları Denetle**düğmesini tıklayın.
- Etki alanı yöneticisinin kullanıcı adını ve parolasını yazın (*wmq\Administrator*) ve **OK > Apply > OK**(Tamam-> Uygula-> Tamam) seçeneklerini tıklayın.
- Yerel Kullanıcılar ve Gruplar penceresini kapatın.

2. *mars* ' da *wmq\wmquser1*olarak oturum açın.

- Bir Windows Gezgini penceresi açın ve `\\sun\wmq`yazın.
Sistem, *sun.wmq.example.com*üzzerindeki *wmq* paylaşımı açılarak yanıt verir ve verileri ve günlük dizinlerini listeler.
- Veri alt dizininde bir dosya oluşturarak, bazı içeriği ekleyerek, okuyarak ve sonra silerek *wmquser1* ' un izinlerini denetleyin.

3. *mars* ' da *wmq\wmquser2*olarak oturum açın ve çekleri yineleyin.

4. Paylaşılan verileri ve günlük dizinlerini kullanmak üzere bir kuyruk yöneticisi yaratmak için sonraki görevi yapın; bkz. “Diğer bir genel güvenlik grubu tarafından yetkilendirilen paylaşılan verileri ve günlük dosyalarını okuma ve yazma” sayfa 353.

Diğer bir genel güvenlik grubu tarafından yetkilendirilen paylaşılan verileri ve günlük dosyalarını okuma ve yazma

Bu görev, **cxrtmqm** komutundaki -a işaretinin nasıl kullanılacağını gösterir. -a işareti, diğer güvenlik grubunu kullanarak uzak dosya paylaşımıyla ilgili günlük ve veri dosyalarına kuyruk yöneticisi erişimi verir.

Bir üretim ölçeği yapılandırmasında, yapılandırmayı var olan bir etki alanına göre uyarlamak zorunda kalabilirsiniz. Örneğin, farklı paylaşımları yetkilendirmek için farklı etki alanı grupları tanımlayabilir ve kuyruk yöneticilerini çalıştıran kullanıcı kimliklerini gruplayabilirsiniz.

Örnek yapılandırma üç sunucudan oluşur:

sun

Bir Windows Server 2008 etki alanı denetleyicisi. *Sun, marsve venus* üzerinden *wmq.example.com* etki alanına sahiptir. Şekil olarak, dosya sunucusu olarak da kullanılır.

mars

İlk IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörünümli kuyruk yöneticisinin bir eşgörünümünü içerir.

venus

İkinci IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörünümli kuyruk yöneticisinin ikinci örneğini içerir.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

Aşağıdaki görevlerdeki adımları gerçekleştirin. Görevler, etki alanı denetleyicisi ve etki alanını oluşturur, IBM WebSphere MQ for Windows ürününü bir sunucuya kurar ve veri ve günlük dosyaları için dosya paylaşımı oluşturur. Var olan bir etki alanı denetleyicisini yapılandırırıyorsanız, yeni bir Windows Server 2008 içindeki adımları denemek için yararlı olabilir. Adımlarınızı etki alanınıza uyarlayabilirsiniz.

1. “IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344.
2. “IBM WebSphere MQ 'un Windows etki alanındaki bir sunucuya ya da iş istasyonuna kurulması” sayfa 347.
3. “Kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan izin yaratılması” sayfa 350.

Bu görev hakkında

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, “Windows etki alanları ve çok eşgörünümli kuyruk yöneticileri” sayfa 340.

Bu görevde, verilerini ve günlüklerini bir dosya sunucusunda uzak bir dizinde saklayan bir kuyruk yöneticisi yaratırsınız. Bu örnekte, dosya sunucusu etki alanı denetleyiciyle aynı sunucudur. Veri ve günlük klasörlerini içeren izin, *wmqhagenel* grubuna verilen tam denetim izniyle paylaşılır.

Yordam

1. Yerel yönetici *mars\Administrator* olarak, etki alanı sunucusunda oturum açın: *mars*.
2. Bir komut penceresi açın.
3. IBM WebSphere MQ hizmetini yeniden başlatın.

Hizmetin altında çalıştığı kullanıcı kimliğinin, kendisi için yapılandığı ek güvenlik kimlik bilgilerini edinmesi için hizmeti yeniden başlatmanız gerekir.

Komutları yazın:

```
endmqsvc  
startmqsvc
```

Sistem yanıtları:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
The MQ service for installation 'Installation1' ended successfully.
```

Ve:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
The MQ service for installation 'Installation1' started successfully.
```

4. Kuyruk yöneticisini yaratın.

```
cdmqm -a wmq\wmqha -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\sun\wmq\data -ld \\sun\wmq\logs  
QMGR
```

You must specify the domain, *wmq*, of the alternative security group *wmqha* by specifying full domain name of the global group "*wmq\wmqha*".

You must spell out the Universal Naming Convention (UNC) name of the share `\\sun\wmq`, and not use a mapped drive reference.

Sistem yanıtı:

```
WebSphere MQ queue manager created.  
Directory '\\sun\wmq\data\QMGR' created.  
The queue manager is associated with installation '1'  
Creating or replacing default objects for queue manager 'QMGR'  
Default objects statistics : 74 created. 0 replaced.  
Completing setup.  
Setup completed.
```

Sonraki adım

Bir iletiyi kuyruğa koyup alarak kuyruk yöneticisini sınavın.

1. Kuyruk yöneticisini başlatın.

```
startmq QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation '1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Bir test kuyruğu yaratın.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Sistemin yapması gereken:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)  
AMQ8006: WebSphere MQ queue created.
```

```
One MQSC command read.  
No commands have a syntax error.  
All valid MQSC commands were processed.
```

3. **amqspu**törnek programını kullanarak bir sinama iletisi koyun.

```
echo 'A test message' | amqspu QTEST QMGR
```

Sistemin yapması gereken:

```
Sample AMQSPUT0 start  
target queue is QTEST  
Sample AMQSPUT0 end
```

4. **amqsge**törnek programını kullanarak sinama iletisini alın.

```
amqsge QTEST QMGR
```

Sistemin yapması gereken:

```
Sample AMQSGET0 start  
message <A test message>  
Wait 15 seconds ...  
no more messages  
Sample AMQSGET0 end
```

5. Kuyruk yöneticisini durdurun.

```
endmqm -i QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ended.
```

6. Kuyruk yöneticisini silin.

```
dltmqm QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' deleted.
```

7. Oluşturduğunuz dizinleri silin.

İpucu: Komut isteminin her bir dosyayı ya da dizini silmesini önlemek için komutlara /Q seçeneğini ekleyin.

```
del /F /S C:\wmq\*.*  
rmdir /S C:\wmq
```

Etki alanı denetleyicilerinde çok eşgörünümlü kuyruk yöneticisi yarat

Örneğin, etki alanı denetleyicilerinde Windows üzerinde çok eşgörünümlü bir kuyruk yöneticisinin nasıl ayarlanacağını gösteren bir örnek. Kurulum, üretim ölçeği olmak yerine ilgili kavramları gösterir. Örnek, Windows Server 2008 'e dayalıdır. Adımlar, Windows Server 'ın diğer sürümlerinde değişiklik gösterebilir.

Yapılandırma, mini etki alanı kavramını ya da "domainlet" kavramını kullanır; bkz. [Windows 2000, Windows Server 2003 ve Windows Server 2008 küme düğümlerini etki alanı denetleyicileri olarak](#). Var olan bir etki alanına çok eşgörünümlü kuyruk yöneticileri eklemek için bkz. ["Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümlü kuyruk yöneticisi yarat" sayfa 341.](#)

Örnek yapılandırma üç sunucudan oluşur:

sun

İlk etki alanı denetleyicisi olarak kullanılan bir Windows Server 2008 sunucusu. Bu, *sun*, *earth* ve *mars* çeren *wmq.example.com* etki alanını tanımlar. Bu belge, QMGRadlı çok eşgörümlü kuyruk yöneticisinin bir eşgörümlü içerir.

earth

İkinci etki alanı denetleyicisi IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, QMGRadlı çok eşgörümlü kuyruk yöneticisinin ikinci örneğini içerir.

mars

Dosya sunucusu olarak kullanılan bir Windows Server 2008.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

1. Windows' ta, kuyruk yöneticisi verilerini depolamayı ve dosya günlük dosyalarını depolamayı planladığınız dosya sistemini doğrulamanız gerekmez. Denetleme yordamı, Paylaşılan dosya sistemi davranışı doğrulanıyor, UNIX and Linux için geçerlidir. Windows' ta denetimler her zaman başarılı olur.
2. Do the steps in “IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344 to create the first domain controller.
3. İkinci bir etki alanı denetleyicisi eklemek için “wmq.example.com etki alanına ikinci bir etki alanı denetleyicisi eklenmesi” sayfa 359 içindeki adımları gerçekleştirin, her iki etki alanı denetleyicisinde IBM WebSphere MQ for Windows ' i kurun ve kuruluşları doğrulayın.
4. İki etki alanı denetleyicisine IBM WebSphere MQ kurmak için “Installing IBM WebSphere MQ on domain controllers in the wmq.example.com domain” sayfa 360 içindeki adımları gerçekleştirin.

Bu görev hakkında

Aynı etki alanındaki bir dosya sunucusunda, kuyruk yöneticisi günlüğü ve veri dizinlerine ilişkin bir paylaşım yaratılır. Daha sonra, etki alanı denetleyicilerinden birinde dosya paylaşımı kullanan çok eşgörümlü bir kuyruk yöneticisinin ilk eşgörümlü yaratın. Diğer etki alanı denetleyicisinde diğer yönetim ortamını yaratın ve son olarak yapılanışı doğrulayın. Bir etki alanı denetleyicide dosya paylaşımı yaratabilirsiniz.

Örnekte, *sun* ilk etki alanı denetleyicisi, *earth* saniye ve *mars* dosya sunucusudur.

Yordam

1. Kuyruk yöneticisi verilerini ve günlük dosyalarını içerecek dizinleri yaratın.

a) *mars*' ta şu komutu yazın:

```
md c:\wmq\data , c:\wmq\logs
```

2. Kuyruk yöneticisi verilerini ve günlük dosyalarını içerecek dizinleri paylaşın.

You must permit full control access to the domain local group mqm, and the user ID you use to create the queue manager. Örnekte, Domain Administrators üyesi olan kullanıcı kimlikleri kuyruk yöneticisi yaratma yetkisine sahiptir.

Dosya paylaşımında, etki alanı denetleyicileriyle aynı etki alanında bulunan bir sunucuda yer almalısınız. In the example, the server *mars* is in the same domain as the domain controllers.

- a) Windows Explorer 'da **c: \wmq > Properties**(Özellikler) seçeneğini sağ tıklayın.
- b) **Güvenlik** sekmesini tıklayın ve **Gelişmiş > Düzenle ...**seçeneklerini tıklayın.
- c) **Bu nesnenin iyesinden devralınabilir izinler ekle** onay kutusunun işaretini kaldırın. Windows Güvenliği penceresinde **Kopyala** ögesini tıklayın.
- d) **İzin girişleri** listesinde Kullanıcılar için satırları seçin ve **Kaldır** düğmesini tıklayın. SYSTEM, Administrators ve CREATOR OWNER satırlarını **İzin girişleri** listesinde bırakın.

- e) **Ekle ...**düğmesini tıktatın. and type the name of the domain local group *wmq*. **Adları Denetle**düğmesini tıktatın.
 - f) Windows Security (Windows Güvenliği) penceresine yanıt olarak, Domain Administrator 'in adını ve parolasını yazın ve **OK > OK**(Tamam-> Tamam) düğmesini tıktatın.
 - g) *wmq* için İzin Girdisi penceresinde, **İzinler**listesinde **Tam Denetim** ögesini seçin.
 - h) **Tamam > Uygula > Tamam > Tamam > Tamam**ögesini tıktatın
 - i) Domain Administratorselemek için **e - h** arasındaki adımları yineleyin.
 - j) Windows Explorer 'da **c: \wmq > Paylaş** ...seçeneğini sağ tıktatın.
 - k) **İleri Düzey Paylaşım ...**düğmesini tıktatın. ve **Bu klasörü paylaş** onay kutusunu seçin. Paylaşım adını *wmq*olarak bırakın.
 - l) **İzinler > Ekle ...**öğelerini tıktatın. and type the name of the domain local group *wmq*; Domain Administrators. **Adları Denetle**düğmesini tıktatın.
 - m) Windows Security (Windows Güvenliği) penceresine yanıt olarak, Domain Administrator 'in adını ve parolasını yazın ve **OK > OK**(Tamam-> Tamam) düğmesini tıktatın.
3. Create the queue manager *QMGR* on the first domain controller, *sun*.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\mars\wmq\data -ld \\mars\wmq\logs QMGR
```

Sistem yanıtı:

```
WebSphere MQ queue manager created.
Directory '\\mars\wmq\data\QMGR' created.
The queue manager is associated with installation 'Installation1'.
Creating or replacing default objects for queue manager 'QMGR'.
Default objects statistics : 74 created. 0 replaced. 0 failed.
Completing setup.
Setup completed.
```

4. Start the queue manager on *sun*, permitting a standby instance.

```
strmqm -x QMGR
```

Sistem yanıtı:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

5. *earth* üzerinde ikinci bir *QMGR* yönetim ortamı yaratın.

- a) *earth* için Prefix ve InstallationName parametrelerinin doğru olup olmadığını denetleyin.

sun' ta **dspmqrinf** komutunu çalıştırın:

```
dspmqrinf QMGR
```

Sistem yanıtı:

```
QueueManager:
  Name=QMGR
  Directory=QMGR
  Prefix=C:\Program Files\IBM\WebSphere MQ
  DataPath=\\mars\wmq\data\QMGR
  InstallationName=Installation1
```

- b) **QueueManager** stanza 'nın makine tarafından okunabilir formunu panoya kopyalayın.

sun üzerinde, `-o` command parametresi ile **dspmqrinf** komutunu yeniden çalıştırın.

```
dspmqrinf -o command QMGR
```

Sistem yanıtı:

```
addmqinf -s QueueManager -v Name=QMGR  
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"  
-v DataPath=\\mars\wmq\data\QMGR
```

- c) On *earth* run the **addmqinf** command from the clipboard to create an instance of the queue manager on *earth*.

Adjust the command, if necessary, to accommodate differences in the `Önek` or `InstallationName` parameters.

```
addmqinf -s QueueManager -v Name=QMGR  
-v Directory=QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"  
-v DataPath=\\mars\wmq\data\QMGR
```

WebSphere MQ configuration information added.

6. *earth* üzerinde kuyruk yöneticisinin yedek yönetim ortamını başlatın.

```
strmqm -x QMGR
```

Sistem yanıtı:

```
WebSphere MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
A standby instance of queue manager 'QMGR' has been started. The active  
instance is running elsewhere.
```

Sonuçlar

Kuyruk yöneticisi anahtarlarının *sun* 'ten *earth*' e geçtiğinden emin olun:

1. *sun* 'ta komutu çalıştırın:

```
endmqm -i -r -s QMGR
```

sun üzerindeki sistem yanıtı:

```
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ended, permitting switchover to  
a standby instance.
```

2. *earth* üzerinde komutu sürekli olarak yazın:

```
dspmqr
```

Sistem yanıtları:

```
QMNAME(QMGR) STATUS(Running as standby)  
QMNAME(QMGR) STATUS(Running as standby)  
QMNAME(QMGR) STATUS(Running)
```

Sonraki adım

Örnek programları kullanarak çok eşgörunümlü bir kuyruk yöneticisini doğrulamak için; bkz. [“Pencerelerüzerinde çok eşgörunümlü kuyruk yöneticisini doğrulayın” sayfa 363.](#)

İlgili görevler

[“wmq.example.com etki alanına ikinci bir etki alanı denetleyicisi eklenmesi” sayfa 359](#)

[“Installing IBM WebSphere MQ on domain controllers in the wmq.example.com domain” sayfa 360](#)

İlgili bilgiler

[Windows 2000, Windows Server 2003 ve Windows Server 2008 küme düğümlerini etki alanı denetleyicileri olarak](#)

wmq.example.com etki alanına ikinci bir etki alanı denetleyicisi eklenmesi

Etki alanı denetleyicilerinde ve dosya sunucularında çok eşgörunümlü kuyruk yöneticilerinin çalıştırılacağı bir Windows etki alanı oluşturmak için, *wmq.example.com* etki alanına ikinci bir etki alanı denetleyicisi ekleyin.

Örnek yapılandırma üç sunucudan oluşur:

sun

İlk etki alanı denetleyicisi olarak kullanılan bir Windows Server 2008 sunucusu. Bu, *sun*, *earth* ve *mars* çeren *wmq.example.com* etki alanını tanımlar. Bu belge, *QMGR* adlı çok eşgörunümlü kuyruk yöneticisinin bir eşgörunümünü içerir.

earth

İkinci etki alanı denetleyicisi IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR* adlı çok eşgörunümlü kuyruk yöneticisinin ikinci örneğini içerir.

mars

Dosya sunucusu olarak kullanılan bir Windows Server 2008.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

1. Do the steps in [“IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344](#) to create a domain controller, *sun*, for the domain *wmq.example.com*. Yapılanışınıza uygun olarak, italik olarak adları değiştirin.
2. Windows Server 2008 'i varsayılan iş grubundaki bir sunucuya (WORKGROUP) kurun. Örneğin, sunucu *earth* adını taşır.

Bu görev hakkında

Bu görevde, *wmq.example.com* etki alanında ikinci bir etki alanı denetleyicisi olarak, *earth* adlı bir Windows Server 2008 'i yapılandırmanız.

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, [“Windows etki alanları ve çok eşgörunümlü kuyruk yöneticileri” sayfa 340.](#)

Yordam

1. Etki alanı denetleyicisini, *sun.wmq.example.com* 'yi bir DNS sunucusu olarak *earth* ' e ekleyin.
 - a) *earth* üzerinde, *earth\Administrator* olarak oturum açın ve **Start**(Başlat) ögesini tıklatın.
 - b) **Ağ > Özellikler > Ağ bağlantılarını yönet** seçeneğini sağ tıklatın.
 - c) Ağ bağdaştırıcısını farenin sağ düğmesiyle tıklatın ve **Properties**(Özellikler) seçeneğini belirleyin.

Sistem, bağlantı kullandığı öğeleri listeleyen Yerel Ağ Bağlantısı Özellikleri pencere ile yanıt verir.

- d) Yerel Ağ Bağlantısı Özellikleri penceresindeki öğeler listesinden **Internet Protocol Sürüm 4** ya da **Internet Protocol Sürüm 6** 'ı seçin. **Özellikler > Gelişmiş ...**seçeneklerini tıklatın. ve **DNS** sekmesini tıklatın.
 - e) DNS sunucusu adresleri altında, **Ekle ...**düğmesini tıklatın.
 - f) DNS sunucusu da olan etki alanı denetleyicisinin IP adresini yazın ve **Ekledüğmesini** tıklatın.
 - g) **Bu DNS soneklerini ekle > Ekle ...**düğmesini tıklatın.
 - h) *wmq.example.com* yazın ve **Ekledüğmesini** tıklatın.
 - i) **Bu bağlantı için DNS soneki alanına** *wmq.example.com* yazın.
 - j) **Bu bağlantının adresini DNS ' de kaydet** ve **DNS kaydındaki bu bağlantının sonekini kullan**seçeneğini belirleyin. **Tamam > Tamam > Kapat**seçeneklerini tıklatın.
 - k) Bir komut penceresi açın ve TCP/IP ayarlarını gözden geçirmek için **ipconfig /all** komutunu yazın.
2. Yerel ya da Workgroup yöneticisi olarak, etki alanı denetleyicide *sunoturum* açın.
- Sunucu etki alanı denetleyicisi olarak önceden yapılandırıldıysa, etki alanı yöneticisi olarak oturum açmanız gerekir.
3. Active Directory Etki Alanı Hizmetleri sihirbazını çalıştırın.
- a) **Başlat > Çalıştır ...**seçeneklerini tıklatın. *dcprmo* yazın ve **Tamam** 'ı tıklatın.
- If the Active Directory binary files are not already installed, Windows installs the files automatically.
4. *earth* ' u *wmq.example.com* etki alanında ikinci etki alanı denetleyicisi olarak yapılandırın.
- a) Sihirbazın ilk penceresinde, **Gelişmiş kip kuruluşunu kullan** onay kutusunun işaretini kaldırın. **İleri > İleri** 'yi tıklatın ve **Var olan bir etki alanına etki alanı denetleyicisi ekle > İleri** 'yi tıklatın.
 - b) Type *wmq* into the **Bu ormandaki herhangi bir etki alanının adını yazın ...** alan. **Diğer kimlik bilgileri** radyo düğmesi tıklatıldı, **Ayarla ...**simgesini tıklatın. Etki alanı denetimcisinin adını ve parolasını yazın ve **Tamam > Sonraki > Sonraki > Sonraki**öğesini tıklatın.
 - c) Ek Etki Alanı Denetleyicisi Seçenekleri penceresinde, seçilen **DNS sunucusu** ve **Genel katalog** seçeneklerini kabul edin; **İleri > İleri** 'yi tıklatın.
 - d) Directory Services Restore Mode Administrator Password (Dizin Hizmetleri Geri Yükleme Kipi Yönetici Parolası) altında, bir **Password** (Parola) ve **Confirm password** (Parolayı onaylayın) yazıp **Next > Next**(İleri)
 - e) **Network Credentials**(Ağ Kimlik Bilgileri) için bilgi isteminde bulunulduğunda, etki alanı yöneticisinin parolasını yazın. Son sihirbaz penceresinde **Tamamlamayı yeniden başlat** seçeneğini belirleyin.
 - f) Bir süre sonra, DNS yetkilendirmesine ilişkin bir **DCPrmo** hatasıyla birlikte bir pencere açılabilir; **Tamam**düğmesini tıklatın. Sunucu yeniden yüklenir.

Sonuçlar

earth yeniden yüklendiğinde, Etki Alanı Yöneticisi olarak oturum açın. *wmq.example.com* etki alanının *earth*olarak eşlenip eşlendiğini denetleyin.

Sonraki adım

IBM WebSphere MQkuruluşuna devam edin; bkz. [“Installing IBM WebSphere MQ on domain controllers in the wmq.example.com domain”](#) sayfa 360.

İlgili görevler

[“IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması”](#) sayfa 344

Installing IBM WebSphere MQ on domain controllers in the wmq.example.com domain

wmq.example.com etki alanındaki her iki etki alanı denetleyicisinde de IBM WebSphere MQ kuruluşunu kurun ve yapılandırın.

Kısa açıklamanızı buraya koyun; ilk paragraf ve özet için kullanılır.

Örnek yapılandırma üç sunucudan oluşur:

sun

İlk etki alanı denetleyicisi olarak kullanılan bir Windows Server 2008 sunucusu. Bu, *sun*, *earthve mars*çeren *wmq.example.com* etki alanını tanımlar. Bu belge, *QMGR*adlı çok eşgörünümlü kuyruk yöneticisinin bir eşgörünümünü içerir.

earth

İkinci etki alanı denetleyicisi IBM WebSphere MQ sunucusu olarak kullanılan bir Windows Server 2008. Bu belge, *QMGR*adlı çok eşgörünümlü kuyruk yöneticisinin ikinci örneğini içerir.

mars

Dosya sunucusu olarak kullanılan bir Windows Server 2008.

Örnekteki italik adlarını, seçtiğiniz adlarla değiştirin.

Başlamadan önce

1. Do the steps in “IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344 to create a domain controller, *sun*, for the domain *wmq.example.com*. Yapılanışınıza uygun olarak, italik olarak adları değiştirin.
2. Do the steps in “*wmq.example.com* etki alanına ikinci bir etki alanı denetleyicisi eklenmesi” sayfa 359 to create a second domain controller, *earth*, for the domain *wmq.example.com*. Yapılanışınıza uygun olarak, italik olarak adları değiştirin.
3. See Windows sistemleri üzerinde donanım ve yazılım gereksinimleri for other Windows versions you can run IBM WebSphere MQ on.

Bu görev hakkında

wmq.example.com etki alanındaki her iki etki alanı denetleyicisinde de IBM WebSphere MQ kuruluşunu kurun ve yapılandırın.

Yordam

1. *sun* ve *earth*üzerine IBM WebSphere MQ kurun.

IBM WebSphere MQ for Windows kuruluş sihirbazını çalıştırma hakkında daha fazla bilgi için; bkz. Installing IBM WebSphere MQ server on Windows .

- a) Hem *sun* hem de *earth*' ta etki alanı yöneticisi olarak oturum açın *wmq\Administrator*.
- b) IBM WebSphere MQ for Windows kuruluş ortamında **Setup** komutunu çalıştırın.

IBM WebSphere MQ Launchpad uygulaması başlar.

- c) Önkoşul yazılımların kurulu olup olmadığını denetlemek için **Yazılım Gereksinimleri** seçeneğini tıklattın.
- d) **Ağ Yapılandırması** > **Hayır**seçeneklerini tıklattın.

Bu kuruluş için bir etki alanı kullanıcı kimliği yapılandırabilir ya da yapılandırabilirsiniz. Yaratılan kullanıcı kimliği, bir etki alanı yerel kullanıcı kimliğidir.

- e) Click **WebSphere MQ Kuruluşu**, select an installation language and click Launch IBM IBM WebSphere MQ Installer.
- f) Lisans sözleşmesini onaylayın ve varsayılan yapılandırmayı kabul etmek için **Next** > **Next** > **Install** (İleri-> Kur) seçeneklerini tıklattın. Kuruluşun tamamlanmasını bekleyin ve **Finish**(Son) düğmesini tıklattın.

Kuruluşun adını değiştirmek, farklı bileşenler kurmak, kuyruk yöneticisi verileri ve günlükler için farklı bir izin yapılandırmak ya da farklı bir dizine kurmak istiyorsanız, **Tipik**yerine **Özel** ögesini tıklattın.

IBM WebSphere MQ kurulu ve kuruluş programı "Prepare IBM WebSphere MQ" sihirbazını başlatır.

IBM WebSphere MQ for Windows kurulumu, bir etki alanı yerel grubu mqmve bir etki alanı grubu Domain mqm yapılandırır. Bu, Domain mqm 'ı mqmüyesi yapar. Aynı etki alanındaki sonraki etki alanı denetleyicileri, mqm ve Domain mqm gruplarını paylaşır.

2. Hem *earth* hem de *sun* üzerinde, "PreparePrepare IBM WebSphere MQ" sihirbazını çalıştırın.

"Hazırlama IBM WebSphere MQ" sihirbazını çalıştırma hakkında daha fazla bilgi için bkz. [Prepare WebSphere MQ sihirbazı ile WebSphere MQ uygulamasını yapılandırma](#).

- a) IBM WebSphere MQ kurulum programı, "Prepare IBM WebSphere MQ" programını otomatik olarak çalıştırır.

To start the wizard manually, find the shortcut to the "Hazırlama IBM WebSphere MQ" in the **Başlat > Tüm programlar > IBM WebSphere MQ** folder. Çoklu kurulum yapılandırmasındaki IBM WebSphere MQ kurulumuna karşılık gelen kısayolu seçin.

- b) Click **Sonraki** and leave **Hayır** clicked in response to the question "Ağda bir Windows 2000 ya da daha sonraki bir etki alanı denetleyicisi olup olmadığını belirleyin"¹.
- c) Sihirbazın son sayfasında, istediğiniz şekilde onay kutularını işaretleyin ya da kutunun işaretini kaldırın ve **Sondüğmesini** tıklayın.

"PreparePrepare IBM WebSphere MQ" sihirbazı, ilk etki alanı denetleyicide bir etki alanı yerel kullanıcısı (MUSR_MQADMIN) ve ikinci etki alanı denetleyicide bulunan başka bir etki alanı yerel kullanıcısı MUSR_MQADMIN1 yaratır. The wizard creates the IBM IBM WebSphere MQ service on each controller, with MUSR_MQADMIN or MUSR_MQADMIN1 as the user that logs on the service.

3. Kuyruk yöneticisi yaratma izni olan bir kullanıcı tanımlayın.

Kullanıcı yerel olarak oturum açma hakkına sahip olmalıdır ve etki alanı yerel mqm grubunun bir üyesi olmalıdır. Etki alanı denetleyicilerinde, etki alanı kullanıcılarının yerel olarak oturum açma hakkı yoktur, ancak denetimciler bunu yapar. Varsayılan olarak, kullanıcının her iki özneliği de yoktur. Bu görevde, etki alanı yöneticilerini etki alanı yerel mqm grubuna ekleyin.

- a) **Server Manager > Roller > Active Directory Etki Alanı Hizmetleri > wmq.example.com > Kullanıcılar** seçeneklerini açın.
- b) **Etki Alanı Yöneticileri > Gruba Ekle ...** seçeneğini sağ tıklayın. ve mqm yazın; **Adları denetle > Tamam > Tamam** düğmesini tıklayın.

Sonuçlar

1. Check that the "Hazırlama IBM WebSphere MQ" created the domain user, MUSR_MQADMIN:

- a. **Server Manager > Roller > Active Directory Etki Alanı Hizmetleri > wmq.example.com > Kullanıcılar** seçeneklerini açın.

- b. Sağ tıklayın **MUSR_MQADMIN > Özellikler ... > Üye**, bunun bir Domain users ve mqmüyesi olduğunu görün.

2. MUSR_MQADMIN ' in hizmet olarak çalıştırma hakkına sahip olup olmadığını denetleyin:

- a. tıklama **Başlat > Çalıştır ...** seçeneklerini tıklayın, **secpo1.msc** komutunu girin ve **Tamam** 'ı tıklayın.

- b. **Güvenlik Ayarları > Yerel İlkeler > Kullanıcı Hakları Atamaları** 'ı açın. İlkeler listesinde, **Hizmet olarak oturum aç > Özellikler** ve bkz. MUSR_MQADMIN , hizmet olarak oturum açma hakkına sahip olarak listeleniyor. **Tamam** 'ı tıklayın.

Sonraki adım

1. Do the task, "Yerel mqm grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması" sayfa 370, to verify that the installation and configuration are working correctly.

¹ Etki alanına ilişkin kurulumu yapılandırabilirsiniz. Bir etki alanı denetleyicindeki tüm kullanıcılar ve grupların etki alanı kapsamı varsa, bu herhangi bir fark yapmaz. IBM WebSphere MQ , etki alanında değilse, asürünü kurmak daha kolay olur.

2. Go back to the task, “Etki alanı denetleyicilerinde çok eşgörünümlü kuyruk yöneticisi yarat” sayfa 355, to complete the task of configuring a multi-instance queue manager on domain controllers.

İlgili kavramlar

Bir WebSphere MQ Windows Hizmeti için gereken kullanıcı hakları

Pencereler üzerinde çok eşgörünümlü kuyruk yöneticisini doğrulayın

Çok eşgörünümlü bir kuyruk yöneticisi yapılanışını doğrulamak için **amqsgbac**, **amqspbac** ve **amqsmbac** örnek programlarını kullanın. Bu konuda, Pencereler Server 2003 üzerinde çok eşgörünümlü bir kuyruk yöneticisi yapılandırmasını doğrulamak için bir örnek yapılandırma sağlanmaktadır.

Yüksek kullanılabilirlik örnek programları, otomatik istemci bağlantısını kullanır. Bağlı kuyruk yöneticisi başarısız olduğunda, istemci, aynı kuyruk yöneticisi grubundaki bir kuyruk yöneticisine yeniden bağlanmayı dener. The description of the samples, Yüksek kullanılabilirlik örnek programları, demonstrates client reconnection using a single instance queue manager for simplicity. Çok eşgörünümlü kuyruk yöneticisi yapılanışlarının doğrulanabilmesi için, çok eşgörünümlü kuyruk yöneticileriyle aynı örnekleri kullanabilirsiniz.

Bu örnek, “Etki alanı denetleyicilerinde çok eşgörünümlü kuyruk yöneticisi yarat” sayfa 355 içinde açıklanan çok eşgörünümlü yapılandırmayı kullanır. Çok eşgörünümlü kuyruk yöneticisinin yedek yönetim ortamına geçeceğini doğrulamak için yapılandırmayı kullanın. Kuyruk yöneticisini **endmqm** komutuyla durdurun ve -s, geçiş (switchover) seçeneğini kullanın. İstemci programları yeni kuyruk yöneticisi örneğine yeniden bağlanır ve küçük bir gecikmeden sonra yeni yönetim ortamıyla çalışmaya devam eder.

İstemci, Windows XP Service Pack 2 çalıştıran 400 MB 'lik bir VMware görüntüsünün içine kurulur. Güvenlik nedenleriyle, çok eşgörünümlü kuyruk yöneticisini çalıştıran etki alanı sunucularıyla aynı VMware anasistemine bağlı ağ ağlarında bağlanır. Yapılandırmayı basitleştirmek için istemci bağlantı çizelgesini içeren /MQHA klasörünü paylaşır.

Verifying failover using WebSphere MQ Explorer

Hata durumunda yedek sisteme geçişi doğrulamak üzere örnek uygulamaları kullanmadan önce, her bir sunucudaki WebSphere MQ Explorer 'ı çalıştırın. **Uzak Kuyruk Yöneticisi Ekle > Doğrudan birden çok yönetim ortamı kuyruk yöneticisine bağlan** sihirbazını kullanarak her bir kaşife her iki kuyruk yöneticisi yönetim ortamını ekleyin. Her iki örneğin de çalıştığından emin olun, izin veriniz. Close the window running the VMware image with the active instance, virtually powering off the server, or stop the active instance, allowing switchover to standby instance and reconnectable clients to reconnect.

Not: If you power off the server, make sure that it is not the one hosting the MQHA folder!

Not: Yedek yönetim ortamına geçiş yapılmasına izin ver seçeneği, **Kuyruk Yöneticisini Durdur** iletişim kutusunda bulunmayabilir. Kuyruk yöneticisi tek bir eşgörünüm kuyruğu yöneticisi olarak çalıştığı için bu seçenek eksik. Bunu, **Bekleme eşgörünümüne izin ver** seçeneği olmadan başlatmış olmanız gerekir. Kuyruk yöneticisini durdurma isteğiniz reddedilirse, **Ayrıntılar** penceresine bakın; büyük olasılıkla çalışmakta olan bir yedek yönetim ortamı yok.

Örnek programlar kullanılarak yedeği devreye sokma doğrulaması

Etkin yönetim ortamını çalıştırmak için bir sunucu seçin

MQHA dizinini ya da dosya sistemini barınmak için sunuculardan birini seçmiş olabilirsiniz. If you plan to test failover by closing the VMware window running the active server, make sure that it is not the one hosting MQHA!

Etkin kuyruk yöneticisi yönetim ortamını çalıştıran sunucuda

1. *ipaddr1* ve *ipaddr2* seçeneklerini değiştirin ve N: \hasample.tst. içinde aşağıdaki komutları kaydedin.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNNAME('ipaddr1(1414),ipaddr2(1414)') QMNAME(QM1) REPLACE
```

```
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

Not: MCAUSER parametresi boş bırakılarak, istemci kullanıcı kimliği sunucuya gönderilir. İstemci kullanıcı kimliğinin, sunucularda doğru izinlere sahip olması gerekir. Diğer bir seçenek de, SVRCONN kanalındaki **MCAUSER** parametresini, sunucuda yapılandırdığınız kullanıcı kimliğine ayarlamandır.

2. N: \ yolu ile bir komut istemi açın ve komutu çalıştırın:

```
runmqsc -m QM1 < hasample.tst
```

3. Verify that the listener is running and has queue manager control, either by inspecting the output of the **runmqsc** command.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

Or, using the WebSphere MQ Explorer that the TCPIP listener is running and has Control = Queue Manager.

İstemcide

1. Sunucudaki C: \MQHA paylaşılan dizinini, istemcideki N: \ dizinine eşleyin.
2. Open a command prompt with the path N: \ . MQCHLLIB ortam değişkenini sunucu üzerindeki istemci kanal tanımlama çizelgesini (CCDT) gösterecek şekilde ayarlayın:

```
SET MQCHLLIB=N:\data\QM1\@ipcc
```

3. Komut bilgi istemine şu komutları yazın:

```
start amqsgnac TARGET QM1
start amqsmnac -s SOURCE -t TARGET -m QM1
start amqspnac SOURCE QM1
```

Not: Sorun varsa, bir komut isteminde uygulamaları başlatın, böylece neden kodu konsolda yazdırılır ya da AMQERR01.LOG dosyasını N: \data\QM1\errors klasöründeki bir dosyaya yazın.

Etkin kuyruk yöneticisi yönetim ortamını çalıştıran sunucuda

1. Aşağıdakilerden birini yapın:
 - Close the window running the VMware image with the active server instance.
 - WebSphere MQ Explorer olanağını kullanarak, etkin kuyruk yöneticisi yönetim ortamını durdurun ve yedek yönetim ortamına geçiş yapma ve yeniden bağ oluşturma istemcilerinin yeniden bağlanmasını bildirir.
2. En sonunda üç istemci bağlantıyı algılar ve sonra yeniden bağlansın. Bu yapılandırmada, sunucu penceresini kapadığınızda, üç bağlantının yeniden kurulması yaklaşık yedi dakika sürüyor. Bazı bağlantılar diğerlerinden önce iyi bir şekilde yeniden kuruldu.

Sonuçlar

```
N:\>amqspshac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsgshac TARGET QM1
Sample AMQSGHAC start
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli

Bu konuda, bir genel alternatif güvenlik grubu kullanarak kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan bir konumu nasıl güvenceye alabileceğiniz açıklanmaktadır. Farklı sunucularda çalışan bir kuyruk yöneticisinin farklı eşgörünümleri arasında yer paylaşımı yapabilirsiniz.

Genellikle kuyruk yöneticisi verileri ve günlük dosyaları için paylaşılan bir konum ayarlamazsınız. IBM WebSphere MQ for Windows' ı kurduğunuzda, kuruluş programı o sunucuda oluşturulan kuyruk yöneticilerine ilişkin seçtiğiniz bir ana izin yaratır. Dizinlerin yerel mqm grubuyla güvenliğini sağlar ve dizinlere erişmek için IBM WebSphere MQ hizmeti için bir kullanıcı kimliği yaplandırır.

Bir paylaşılan klasörü bir güvenlik grubuyla güvenli bir şekilde koruduğunuzda, klasöre erişmesine izin verilen bir kullanıcının, grubun kimlik bilgilerine sahip olması gerekir. Uzak dosya sunucusundaki bir klasörün, *mars* sadlı sunucudaki yerel mqm grubuyla güvenli kılındığını varsayın. Kuyruk yöneticisi çalıştıran kullanıcıyı, *mars* üzerindeki yerel mqm grubunun bir üyesini işler. Kullanıcı, uzak dosya sunucusundaki klasörün kimlik bilgileriyle eşleşen kimlik bilgilerine sahip olmalıdır. Bu kimlik bilgilerini kullanarak kuyruk yöneticisi, verilerine erişebiliyor ve klasördeki dosyaları günlüğe kaydedebiliyor. Farklı bir sunucudaki kuyruk yöneticisi işlemlerini çalıştıran kullanıcı, eşleşen kimlik bilgilerine sahip olmayan farklı bir yerel mqm grubunun bir üyesidir. When the queue manager runs on a different server to *mars*, it cannot access the data and log files it created when it ran on *mars*. Even if you make the user a domain user, it has different credentials, because it must acquire the credentials from the local mqm group on *mars*, and it cannot do that from a different server.

Kuyruk yöneticisinin genel bir alternatif güvenlik grubuyla sağlanması sorunu çözer; [Şekil 64 sayfa 366](#) konusuna bakın. Uzak klasörü genel bir grupta güvenli bir şekilde sağlayın. Pass the name of the global group to the queue manager when you create it on *mars*. Pass the global group name as the alternative security group using the `-a[ı]` parameter on the `crtmqm` command. Kuyruk yöneticisini farklı bir

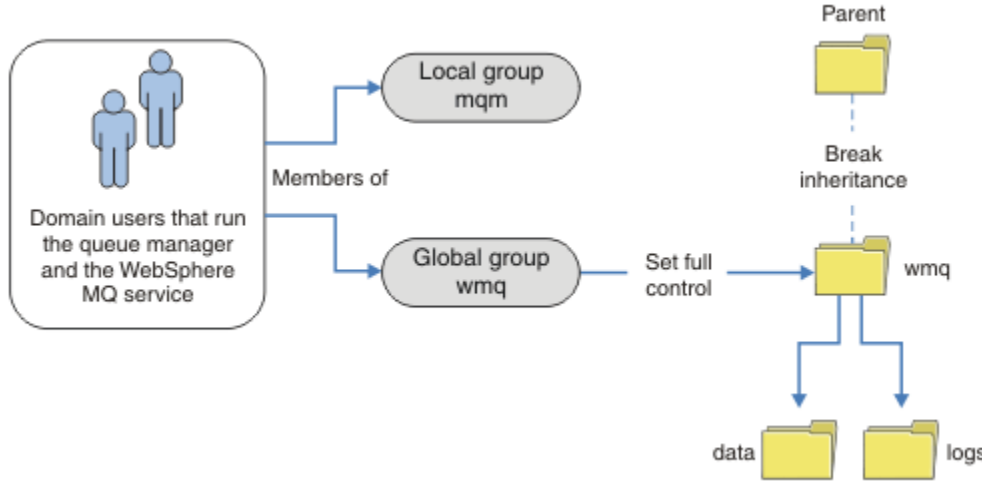
sunucuda çalıştırılacak şekilde aktarırsanız, güvenlik grubunun adı da bu sunucuda aktarılır. Ad, qm.ini dosyasındaki **AccessMode** stanza içinde bir SecurityGroup olarak aktarılır; örneğin:

```
AccessMode:  
SecurityGroup=wmq\wmq
```

qm.ini içindeki **AccessMode** Stanza, RemoveMQMAccess örneğini de içerir; örneğin:

```
AccessMode:  
RemoveMQMAccess=<true/false>
```

Bu öznitelik true değeriyle belirtilirse ve bir erişim grubu da verildiyse, yerel mqm grubuna kuyruk yöneticisi veri dosyalarına erişim izni verilmez.

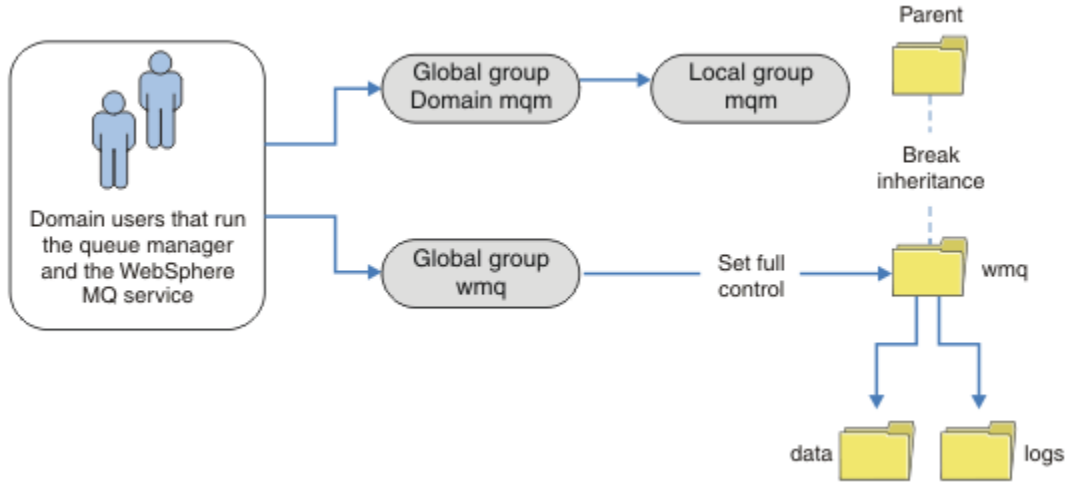


Şekil 64. Alternatif bir genel güvenlik grubu kullanarak kuyruk yöneticisi verilerinin ve günlüklerinin güvenliğini sağlama (1)

Kuyruk yöneticisi işlemlerinin, genel güvenlik grubunun eşleşen kimlik bilgilerine sahip olmak için çalıştırılacağı kullanıcı kimliği için, kullanıcı kimliğinin de genel kapsamı olması gerekir. Bir yerel grubu ya da asıl adı bir genel grubun üyesi yapamazsınız. Şekil 64 sayfa 366 ' ta, kuyruk yöneticisi işlemlerini çalıştıran kullanıcılar etki alanı kullanıcıları olarak gösterilir.

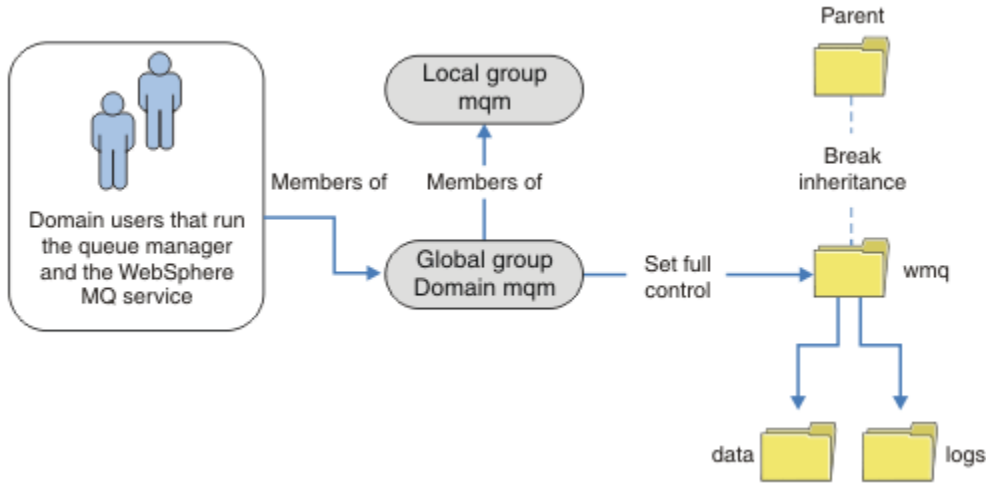
Birçok IBM WebSphere MQ sunucusunu devreye alıyorsanız, Şekil 64 sayfa 366 içindeki kullanıcıların gruplandırması uygun değildir. Her bir IBM WebSphere MQ sunucusu için yerel gruplara kullanıcı ekleme işlemini yinelemeniz gerekir. Bunun yerine, etki alanı denetleyicide bir Domain mqm genel grubu yaratın ve Domain mqm grubunun IBM WebSphere MQ üyelerini çalıştıran kullanıcıları yapın; bkz. Şekil 65 sayfa 367 . When you install IBM WebSphere MQ as a domain installation, the "Hazırlama IBM WebSphere MQ" wizard automatically makes the Domain mqm group a member of the local mqm group. The same users are in both the global groups Domain mqm and wmq.

İpucu: The same users can run IBM WebSphere MQ on different servers, but on an individual server you must have different users to run IBM WebSphere MQ as a service, and run interactively. Bir sunucudaki her kuruluş için farklı kullanıcılardan da sahip olmanız gerekir. Bu nedenle Domain mqm , genellikle bir dizi kullanıcı içerir.



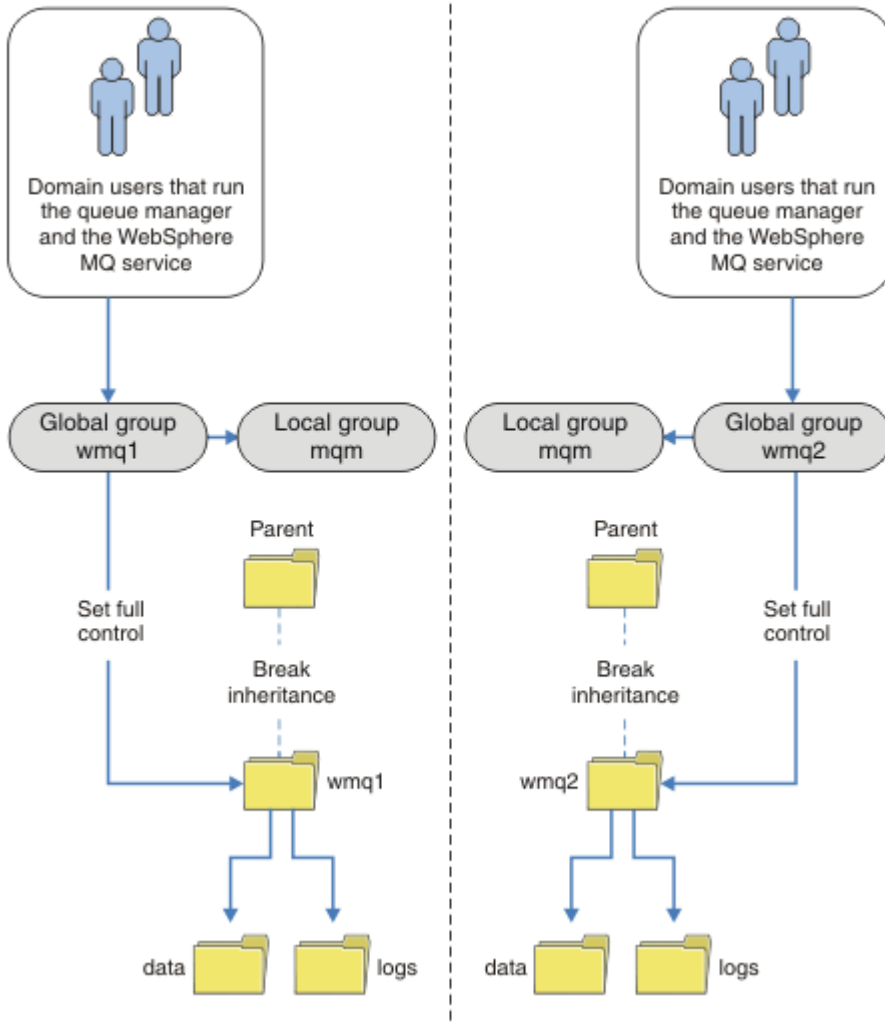
Şekil 65. Diğer bir genel güvenlik grubu kullanarak kuyruk yöneticisi verilerinin ve günlüklerinin güvenliğini sağlama (2)

Şekil 65 sayfa 367 içindeki kuruluş, gerektiği gibi gereksiz bir şekilde karmaşıktır. Düzenlemenin benzer üyeleri olan iki genel grubu var. Kuruluşu basitleştirebilir ve yalnızca bir genel grup tanımlayabilir; bkz. Şekil 66 sayfa 367.



Şekil 66. Alternatif bir genel güvenlik grubu kullanarak kuyruk yöneticisi verilerinin ve günlüklerinin güvenliğini sağlama (3)

Diğer bir seçenek olarak, farklı kuyruk yöneticilerine farklı klasörlere erişebilmek için kısıtlanmış farklı kuyruk yöneticilerine sahip, daha iyi bir erişim denetimine gereksinim duyabilirsiniz; bkz. Şekil 67 sayfa 368. Şekil 67 sayfa 368 uygulamasında, farklı kuyruk yöneticisi günlüğü ve veri dosyalarını güvenli kılmak için ayrı genel gruplarda iki etki alanı kullanıcısı grubu tanımlanır. Farklı IBM WebSphere MQ sunucularında olması gereken iki farklı yerel mqm grubu gösterilir. Bu örnekte, kuyruk yöneticileri iki kümeyle ayrılan farklı kullanıcılarla iki kümeyle bölümlenir. İki küme, test ve üretim kuyruğu yöneticilerine sahip olabilir. Diğer güvenlik grupları wmq1 ve wmq2 olarak adlandırılır. You must manually add the global groups wmq1 and wmq2 to the correct queue managers according to whether they are in the test or production department. The configuration cannot take advantage that the installation of IBM WebSphere MQ propagates Domain mqm to the local mqm group as in Şekil 66 sayfa 367, because there are two groups of users.



Şekil 67. Alternatif bir genel güvenlik birincil kullanıcısı (4) kullanarak kuyruk yöneticisi verilerinin ve günlüklerinin güvenliğini sağlama

İki bölümü bölümleniz için alternatif bir yol, bunları iki pencere etki alanına yerleştirmek olacaktır. Bu durumda, Şekil 66 sayfa 367' ta gösterilen daha basit modeli kullanmaya geri dönebilirsiniz.

Secure unshared queue manager data and log directories and files on Windows

Bu konuda, hem yerel mqm grubu hem de diğer bir güvenlik grubu kullanılarak, kuyruk yöneticisi verileri ve günlük dosyaları için diğer bir yeri nasıl güvenli hale getirebileceğinizi açıklar.

Genellikle kuyruk yöneticisi verileri ve günlük dosyaları için alternatif bir konum ayarlamazsınız. IBM WebSphere MQ for Windows' u kurduğunuzda, kuruluş programı oluşturulan kuyruk yöneticilerine ilişkin seçtiğiniz bir ana dizin yaratır. Dizinlerin yerel mqm grubuyla güvenliğini sağlar ve dizinlere erişmek için IBM WebSphere MQ hizmeti için bir kullanıcı kimliği yapılandırır.

İki örnek, IBM WebSphere MQ için erişim denetiminin nasıl yapılandırılacağını gösterir. Bu örnekler, veri ve günlüklerde, kuruluş tarafından yaratılan veri ve günlük yollarında yer alan veri ve günlüklerle kuyruk yöneticisi yaratılmasını gösterir. İlk örnekte, “Yerel mqm grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması” sayfa 370, yerel mqm grubu tarafından yetkilendirilerek kuyruğa ve günlük dizinlerine erişim izni veriniz. İkinci örnek, “Alternatif bir yerel güvenlik grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması” sayfa 373, dizinlere erişimde farklı bir güvenlik grubu tarafından yetkilendirilmiş olup olmadığını gösterir. Dizinlere yalnızca bir sunucu üzerinde çalışan bir kuyruk yöneticisi tarafından erişildiğinde, verileri ve günlük dosyalarını alternatif güvenlik grubuyla güvenli kılmak, farklı yerel gruplar ya da asıl adlarla farklı kuyruk yöneticilerini güvenli

altına alma seçeneğini size sağlar. Dizinlere, çok eşgörünümlü bir kuyruk yöneticisiyle olduğu gibi farklı sunucularda çalışan bir kuyruk yöneticisi tarafından erişildiğinde, verileri ve günlük dosyalarını diğer güvenlik grubuyla güvenli kılmak tek seçenektir; bkz. [“Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli” sayfa 365.](#)

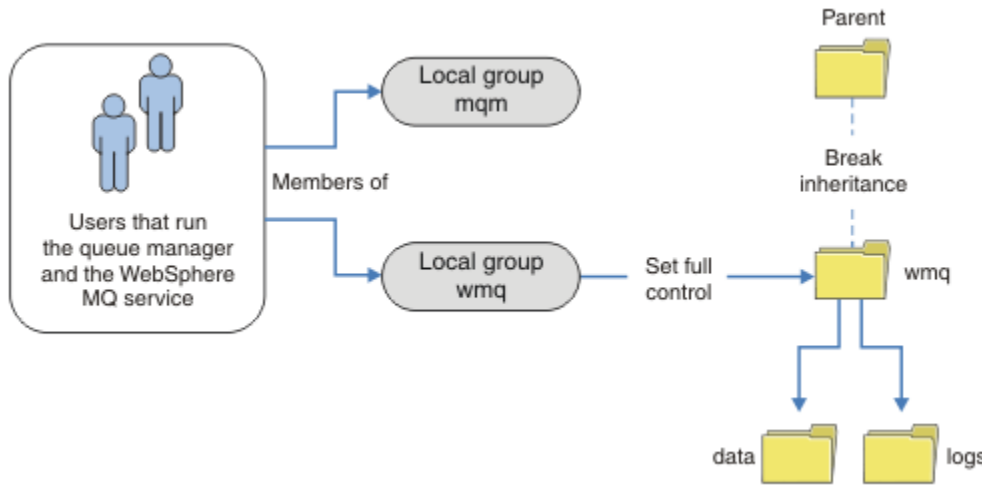
Kuyruk yöneticisi verilerinin ve günlük dosyalarının güvenlik izinlerinin yapılandırılması, Windows üzerinde ortak bir görev değildir. IBM WebSphere MQ for Windows' u kurduğunuzda, kuyruk yöneticisi verileri ve günlükler için dizinler belirtin ya da varsayılan dizinleri kabul edin. Kuruluş programı bu dizinleri yerel mqm grubuyla otomatik olarak sabitler ve tam denetim izni verir. Kuruluş işlemi, kuyruk yöneticilerini çalıştıran kullanıcı kimliğinin, yerel mqm grubunun bir üyesi olduğundan emin olur. Dizinlerdeki diğer erişim izinlerini, erişim gereksinimlerinizi karşılayacak şekilde değiştirebilirsiniz.

Verileri ve günlük dosyaları dizinini yeni konumlara taşırsanız, yeni konumların güvenliğini yapılandırmanız gerekir. Bir kuyruk yöneticisini yedeklemeniz ve farklı bir bilgisayara geri yüklemek ya da kuyruk yöneticisini çok eşgörünümlü bir kuyruk yöneticisi olacak şekilde değiştirirseniz dizinlerin yerini değiştirebilirsiniz. Kuyruk yöneticisi verilerini ve günlük dizinlerini yeni konumlarında güvenceye almak için iki yol seçeneğiniz vardır. Yerel mqm grubuna erişimi kısıtlayarak dizinleri güvenceye almak için, ya da seçtiğiniz herhangi bir güvenlik grubuna erişimi sınırlandırabilirsiniz.

Yerel mqm grubunu kullanarak dizinlerin güvenliğini sağlamak için en az adım sayısı gerekir. Yerel mqm grubu tam denetime izin vermek için veri ve günlük dizinlerine ilişkin izinleri ayarlayın. Tipik bir yaklaşım var olan izin kümesini kopyalamak, üstten devralmayı kaldırmak. Daha sonra diğer birincil kullanıcıların izinlerini kaldırabilir ya da kısıtlayabilirsiniz.

Kuyruk yöneticisini PreparePrepare IBM WebSphere MQ sihirbazı tarafından ayarlanan hizmete farklı bir kullanıcı kimliği altında çalıştırırsanız, bu kullanıcı kimliğinin yerel mqm grubunun bir üyesi olması gerekir. The task, [“Yerel mqm grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması” sayfa 370,](#) takes you through the steps.

Ayrıca, kuyruk yöneticisi verilerini ve günlük dosyalarını, diğer bir güvenlik grubunu kullanarak da güvenceye alabilirsiniz. Kuyruk yöneticisi verilerini ve günlük dosyalarını alternatif güvenlik grubuyla güvenlik altına alma işlemi, [Şekil 68 sayfa 369'](#) e gönderme yapan bir dizi adımdan oluşan bir dizi adımla. wmq adlı yerel grup, bir alternatif güvenlik grubu örneğidir.



Şekil 68. Securing queue manager data and logs using an alternative local security group, wmq

1. Kuyruk yöneticisi verileri ve günlükleri, ortak bir dizin ya da ortak bir üst dizin için ayrı dizinler yaratın.
2. Dizinlere ya da üst dizine ilişkin var olan edinilmiş izinler kümesini kopyalayıp gereksinimlerinize göre değiştirin.
3. Diğer grubu (wmq), dizinlere tam denetim izni vererek, kuyruk yöneticisini ve günlüklerini içerecek dizinleri güvenli hale getirin.

4. Kuyruk yöneticisi çalıştıran tüm kullanıcı kimliklerini, diğer güvenlik grubunun ya da birincil kullanıcının kimlik bilgilerini işleme sokarsa:
 - a. Diğer güvenlik birincil kullanıcısı olarak bir kullanıcı tanımlarsanız, kullanıcı kuyruk yöneticisiyle aynı kullanıcı tarafından çalıştırılır. Kullanıcı, yerel mqm grubunun bir üyesi olmalıdır.
 - b. Alternatif güvenlik grubu olarak bir yerel grup tanımlarsanız, kuyruk yöneticisinin diğer grup altında çalışacağını kullanıcıyı ekleyin. Kullanıcı, yerel mqm grubunun bir üyesi de olmalıdır.
 - c. Alternatif güvenlik grubu olarak bir genel grup tanımlarsanız, bkz. "[Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli](#)" sayfa 365.
5. **crtmqm** komutunda alternatif güvenlik grubunu ya da birincil kullanıcıyı belirterek, -a parametresiyle kuyruk yöneticisini yaratın.

Yerel mqm grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması

Bu görev, seçtiğiniz herhangi bir dizinde saklanan veri ve günlük dosyalarıyla bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Dosyalara erişim, yerel mqm grubu tarafından korunmaktadır. Dizin paylaşılmıyor.

Başlamadan önce

1. Birincil kuruluş olarak IBM WebSphere MQ for Windows ürününü kurun.
2. "Hazırlık IBM WebSphere MQ" sihirbazını çalıştırın. Bu görev için, kuruluşu yerel bir kullanıcı kimliği ya da bir etki alanı kullanıcı kimliği ile çalışacak şekilde yapılandırın. Sonuç olarak, "[Windows etki alanları ve çok eşgözümlü kuyruk yöneticileri](#)" sayfa 340 içindeki tüm görevleri tamamlamak için kuruluşun bir etki alanı için yapılandırılması gerekir.
3. Görevin ilk kısmını gerçekleştirmek için Yönetici yetkisiyle oturum açın.

Bu görev hakkında

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, "[Windows etki alanları ve çok eşgözümlü kuyruk yöneticileri](#)" sayfa 340.

Windows üzerinde, seçtiğiniz herhangi bir dizinde IBM WebSphere MQ for Windows için varsayılan verileri ve günlük yollarını oluşturabilirsiniz. Kuruluş ve yapılanış sihirbazı otomatik olarak yerel mqm grubuna ve kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliğine dizinlere erişim verir. Kuyruk yöneticisi verileri ve günlük dosyaları için farklı izinler belirten bir kuyruk yöneticisi yaratırsanız, izinler için tam denetim izni yapılandırmanız gerekir.

Bu örnekte, yerel mqm grubuna `c : \wmq` dizini için izin vererek kuyruk yöneticisine verileri ve günlük dosyaları üzerinde tam denetim verirsiniz.

crtmqm komutu, iş istasyonu IBM WebSphere MQ hizmetini kullanmaya başladığında otomatik olarak başlayan bir kuyruk yöneticisi yaratır.

Görev açıklayıcı; değiştirebileceğiniz belirli değerleri kullanır. Değiştirebileceğiniz değerler italiktir. Görevin sonunda, yaptığınız tüm değişiklikleri kaldırmak için yönergeleri izleyin.

Yordam

1. Bir komut istemi açın.
2. Komutu yazın:

```
md c:\wmq\data , c:\wmq\logs
```

3. Yerel mqm grubu okuma ve yazma erişimine izin vermek için dizinlere ilişkin izinleri ayarlayın.

```
cacls c:\wmq /T /E /G mqm:F
```

Sistemin yapması gereken:

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

4. İsteğe bağlı: Yerel mqm grubunun üyesi olan bir kullanıcı kimliğine geçin.

Yönetici olarak devam edebilirsiniz, ancak gerçekçi bir üretim yapılandırması için daha sınırlı haklara sahip bir kullanıcı kimliğiyle devam edin. Kullanıcı kimliği en azından yerel mqm grubunun bir üyesi olmalıdır.

IBM WebSphere MQ kurulumu bir etki alanının parçası olarak yapılandırıldıysa, kullanıcı kimliğini Domain mqm grubunun üyesi yapın. "Hazırla IBM WebSphere MQ" sihirbazı Domain mqm genel grubunu yerel mqm grubunun üyesi yapar; bu nedenle, kullanıcı kimliğini doğrudan yerel mqm grubunun üyesi yapmanıza gerek yoktur.

5. Kuyruk yöneticisini yaratın.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager created.
Directory 'c:\wmq\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

6. Kuyruk yöneticisi tarafından yaratılan dizinlerin c:\wmq dizininde olup olmadığını denetleyin.

```
dir c:\wmq /D /B /S
```

7. Dosyaların yerel mqm grubu için okuma ve yazma ya da tam denetim iznine sahip olup olmadığını denetleyin.

```
cacls c:\wmq\*.*
```

Sonraki adım

Bir iletiyi kuyruğa koyup alarak kuyruk yöneticisini sınavın.

1. Kuyruk yöneticisini başlatın.

```
strmqm QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Bir test kuyruğu yaratın.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Sistemin yapması gereken:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)
AMQ8006: WebSphere MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. **amqspu**törnek programını kullanarak bir sinama iletisi koyun.

```
echo 'A test message' | amqspu QTEST QMGR
```

Sistemin yapması gereken:

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

4. **amqsge**törnek programını kullanarak sinama iletisini alın.

```
amqsge QTEST QMGR
```

Sistemin yapması gereken:

```
Sample AMQSGET0 start
message <A test message>
Wait 15 seconds ...
no more messages
Sample AMQSGET0 end
```

5. Kuyruk yöneticisini durdurun.

```
endmqm -i QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' ending.
WebSphere MQ queue manager 'QMGR' ended.
```

6. Kuyruk yöneticisini silin.

```
dltmqm QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' deleted.
```

7. Oluşturduğunuz dizinleri silin.

İpucu: Komut isteminin her bir dosyayı ya da dizini silmesini önlemek için komutlara /Q seçeneğini ekleyin.

```
del /F /S C:\wmq\*. *
rmdir /S C:\wmq
```

İlgili kavramlar

[“Windows etki alanları ve çok eşgörümlü kuyruk yöneticileri” sayfa 340](#)

Windows üzerinde çok eşgörümlü bir kuyruk yöneticisi, verilerinin ve günlüklerin paylaşılmasını gerektirir. Paylaşımın, kuyruk yöneticisinin farklı sunucularda ya da iş istasyonlarında çalışan tüm eşgörümlerine erişebilmesi gerekir. Kuyruk yöneticilerini yapılandırın ve bir Windows etki alanının bir parçası olarak paylaşın. Kuyruk yöneticisi bir etki alanı iş istasyonunda ya da sunucuda ya da etki alanı denetleyicisinde çalışabilir.

İlgili görevler

[“Alternatif bir yerel güvenlik grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması” sayfa 373](#)

Bu görev, **crtmqm** komutunda -a işaretinin nasıl kullanılacağını gösterir. Bu işaret, kuyruk yöneticisine günlük ve veri dosyalarına erişim vermek için alternatif bir yerel güvenlik grubu sağlar.

“Diğer bir genel güvenlik grubu tarafından yetkilendirilen paylaşılan verileri ve günlük dosyalarını okuma ve yazma” sayfa 353

“Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümlü kuyruk yöneticisi yarat” sayfa 341

Alternatif bir yerel güvenlik grubu tarafından yetkilendirilen veri ve günlük dosyalarının okunması ve yazılması

Bu görev, **crtmqm** komutunda -a işaretinin nasıl kullanılacağını gösterir. Bu işaret, kuyruk yöneticisine günlük ve veri dosyalarına erişim vermek için alternatif bir yerel güvenlik grubu sağlar.

Başlamadan önce

1. Birincil kuruluş olarak IBM WebSphere MQ for Windows ürününü kurun.
2. "Hazırlık IBM WebSphere MQ" sihirbazını çalıştırın. Bu görev için, kuruluşu yerel bir kullanıcı kimliği ya da bir etki alanı kullanıcı kimliği ile çalışacak şekilde yapılandırın. Sonuç olarak, “Windows etki alanları ve çok eşgörünümlü kuyruk yöneticileri” sayfa 340 içindeki tüm görevleri tamamlamak için kuruluşun bir etki alanı için yapılandırılması gerekir.
3. Görevin ilk kısmını gerçekleştirmek için Yönetici yetkisiyle oturum açın.

Bu görev hakkında

Bu görev, kuyruk yöneticisi verilerine ve günlük dosyalarına erişimi gösteren ilgili görevler kümesinden biridir. Görevler, seçtiğiniz bir dizinde saklanan verileri ve günlük dosyalarını okuma ve yazma yetkisi olan bir kuyruk yöneticisinin nasıl yaratılacağını gösterir. Görevle birlikte, “Windows etki alanları ve çok eşgörünümlü kuyruk yöneticileri” sayfa 340.

Windows üzerinde, seçtiğiniz herhangi bir dizinde IBM WebSphere MQ for Windows için varsayılan verileri ve günlük yollarını oluşturabilirsiniz. Kuruluş ve yapılanış sihirbazı otomatik olarak yerel mqm grubuna ve kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliğine dizinlere erişim verir. Kuyruk yöneticisi verileri ve günlük dosyaları için farklı dizinler belirten bir kuyruk yöneticisi yaratırsanız, dizinler için tam denetim izni yapılandırmanız gerekir.

Bu örnekte, kuyruk yöneticisine dizinler üzerinde tam denetim yetkisi olan alternatif bir güvenlik yerel grubu sağlıyorsunuz. Diğer güvenlik grubu, kuyruk yöneticisine dizindeki dosyaları yönetme izni verir. Alternatif güvenlik grubunun birincil amacı, alternatif bir güvenlik genel grubuna yetki vermektir. Çok eşgörünümlü bir kuyruk yöneticisi ayarlamak için diğer bir güvenlik genel grubunu kullanın. Bu örnekte, bir etki alanına IBM WebSphere MQ ürününü kurmadan diğer bir güvenlik grubu kullanımını tanımak için yerel bir grup yapılandırırsınız. Yerel bir grubu alternatif bir güvenlik grubu olarak yapılandırmak alışılmadık bir durumdur.

crtmqm komutu, iş istasyonu IBM WebSphere MQ hizmetini kullanmaya başladığında otomatik olarak başlayan bir kuyruk yöneticisi yaratır.

Görev açıklayıcı; değiştirebileceğiniz belirli değerleri kullanır. Değiştirebileceğiniz değerler italiktir. Görevin sonunda, yaptığınız tüm değişiklikleri kaldırmak için yönergeleri izleyin.

Yordam

1. Alternatif bir güvenlik grubu ayarlayın.

Alternatif güvenlik grubu genellikle bir etki alanı grubudur. Örnekte, yerel bir diğer güvenlik grubunu kullanan bir kuyruk yöneticisi yaratırsınız. Yerel bir diğer güvenlik grubuyla, görevi bir etki alanının parçası olmayan bir IBM WebSphere MQ kuruluşuyla yapabilirsiniz.

- a) Yerel Kullanıcılar ve Gruplar penceresini açmak için **lusrmgr.msc** komutunu çalıştırın.
- b) **Gruplar > Yeni Grup ...** ögesini sağ tıklayın.
- c) **Grup adı** alanına *al tmqm* yazın ve **Oluştur > Kapat** seçeneğini tıklayın.
- d) IBM WebSphere MQ hizmetini çalıştıran kullanıcı kimliğini tanımlayın.

- i) **Başlat > Çalıştır ...**düğmesini tıklatın, `services.msc` yazın ve **Tamam**düğmesini tıklatın.
 - ii) Hizmetler listesinde IBM IBM WebSphere MQ hizmetini tıklatın ve Oturum Açma sekmesini tıklatın.
 - iii) Kullanıcı kimliğini hatırlayın ve Hizmet Gezini 'ni kapatın.
- e) IBM IBM WebSphere MQ hizmetini çalıştıran kullanıcı kimliğini `al tmqm` grubuna ekleyin. Ayrıca, bir kuyruk yöneticisi yaratmak için oturum açtığınız kullanıcı kimliğini ekleyin ve etkileşimli olarak çalıştırın.

Windows , kuyruk yöneticisi işlemlerini çalıştıran kullanıcı kimliğinin yetkisini denetleyerek, veri ve günlük dizinlerine erişmek için kuyruk yöneticisinin yetkisini denetler. Kullanıcı kimliği, doğrudan ya da dolaylı olarak dizinlere yetki veren `al tmqm` grubunun bir üyesi olmalıdır. IBM WebSphere MQ ürününü bir etki alanının parçası olarak kurduysanız ve “Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümli kuyruk yöneticisi yarat” sayfa 341 içindeki görevleri gerçekleştirecekseniz, “IBM WebSphere MQ için Active Directory ve DNS etki alanı yaratılması” sayfa 344 içinde oluşturulan etki alanı kullanıcı kimlikleri `wmquser1` ve `wmquser2` olur.

Kuyruk yöneticisini bir etki alanının parçası olarak kurmadıysanız, IBM IBM WebSphere MQ hizmetini çalıştıran varsayılan yerel kullanıcı kimliği `MUSR_MQADMIN` olur. Görevleri Yönetici yetkisi olmadan gerçekleştirmek istiyorsanız, yerel `mqm` grubunun üyesi olan bir kullanıcı yaratın. `al tmqm`' e `wmquser1` ve `wmquser2` eklemek için aşağıdaki adımları izleyin. Yapılandırmanız farklıysa, adlarınızı kullanıcı kimlikleriyle ve grupla değiştirin.

- i) Grup listesinde **al tmqm > Özellikler > Ekle ...**seçeneğini sağ tıklatın.
 - ii) Select Users, Computers ya da Groups (Kullanıcı, Bilgisayar ya da Grup Seç) penceresinde `wmquser1`; `wmquser2` yazın ve **Check Names**(Adları Denetle) seçeneğini tıklatın.
 - iii) Windows Güvenlik penceresinde bir etki alanı yöneticisinin adını ve parolasını yazın ve **Tamam > Tamam > Uygula > Tamam**düğmesini tıklatın.
2. Bir komut istemi açın.
3. IBM IBM WebSphere MQ hizmetini yeniden başlatın.

Hizmetin altında çalıştığı kullanıcı kimliğinin, kendisi için yapılandığı ek güvenlik kimlik bilgilerini edinmesi için hizmeti yeniden başlatmanız gerekir.

Komutları yazın:

```
endmqsvc
startmqsvc
```

Sistem yanıtları:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' ended successfully.
Ve:
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' started successfully.
```

4. Komutu yazın:

```
md c:\wmq\data , c:\wmq\logs
```

5. `user` yerel kullanıcısının okuma ve yazma erişimine izin vermek için dizinlerdeki izinleri ayarlayın.

```
cacls c:\wmq /T /E /G al tmqm:F
```

Sistemin yapması gereken:

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

6. İsteğe bağlı: Yerel `mqm` grubunun üyesi olan bir kullanıcı kimliğine geçin.

Yönetici olarak devam edebilirsiniz, ancak gerçekçi bir üretim yapılandırması için daha sınırlı haklara sahip bir kullanıcı kimliğiyle devam edin. Kullanıcı kimliği en azından yerel mqm grubunun bir üyesi olmalıdır.

IBM WebSphere MQ kurulumu bir etki alanının parçası olarak yapılandırıldıysa, kullanıcı kimliğini Domain mqm grubunun üyesi yapın. "Hazırla IBM WebSphere MQ" sihirbazı Domain mqm genel grubunu yerel mqm grubunun üyesi yapar; bu nedenle, kullanıcı kimliğini doğrudan yerel mqm grubunun üyesi yapmanıza gerek yoktur.

7. Kuyruk yöneticisini yaratın.

```
crtmqm -a altmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq/logs QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager created.  
Directory 'c:\wmq1\data\QMGR' created.  
The queue manager is associated with installation '1'  
Creating or replacing default objects for queue manager 'QMGR'  
Default objects statistics : 74 created. 0 replaced.  
Completing setup.  
Setup completed.
```

8. Kuyruk yöneticisi tarafından yaratılan dizinlerin c:\wmq dizininde olup olmadığını denetleyin.

```
dir c:\wmq /D /B /S
```

9. Dosyaların yerel mqm grubu için okuma ve yazma ya da tam denetim iznine sahip olup olmadığını denetleyin.

```
cacls c:\wmq\*.*
```

Sonraki adım

Bir iletiyi kuyruğa koyup alarak kuyruk yöneticisini sınavın.

1. Kuyruk yöneticisini başlatın.

```
strmqm QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation '1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
WebSphere MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Bir test kuyruğu yaratın.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Sistemin yapması gereken:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)  
AMQ8006: WebSphere MQ queue created.  
One MQSC command read.
```

No commands have a syntax error.
All valid MQSC commands were processed.

3. **amqspu**törnek programını kullanarak bir sinama iletisi koyun.

```
echo 'A test message' | amqspu QTEST QMGR
```

Sistemin yapması gereken:

```
Sample AMQSPUT0 start  
target queue is QTEST  
Sample AMQSPUT0 end
```

4. **amqsge**törnek programını kullanarak sinama iletisini alın.

```
amqsge QTEST QMGR
```

Sistemin yapması gereken:

```
Sample AMQSGET0 start  
message <A test message>  
Wait 15 seconds ...  
no more messages  
Sample AMQSGET0 end
```

5. Kuyruk yöneticisini durdurun.

```
endmqm -i QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' ending.  
WebSphere MQ queue manager 'QMGR' ended.
```

6. Kuyruk yöneticisini silin.

```
dltmqm QMGR
```

Sistemin yapması gereken:

```
WebSphere MQ queue manager 'QMGR' deleted.
```

7. Oluşturduğunuz dizinleri silin.

İpucu: Komut isteminin her bir dosyayı ya da dizini silmesini önlemek için komutlara /Q seçeneğini ekleyin.

```
del /F /S C:\wmq\*.*  
rmdir /S C:\wmq
```

Linux üzerinde çok eşgörümlü bir kuyruk yöneticisi yarat

Örneğin, Linux üzerinde çok eşgörümlü bir kuyruk yöneticisinin nasıl ayarlanacağını gösteren bir örnek. Kuruluş, ilgili kavramları göstermek için küçük bir ayardır. Örnek, Linux Red Hat Enterprise 5'e dayanır. Bu adımlar, diğer UNIX altyapılarında farklılık gösterir.

Örnek, 3 GB RAM çalıştıran 2 GHz dizüstü bilgisayarda Windows XP Service Pack 2'nin çalıştığı 2 GHz'lik bir dizüstü bilgisayarda ayarlanır. Two VMware virtual machines, Server1 and Server2, run Linux Red Hat Enterprise 5 in 640 MB images. Server1, ağ dosya sistemini (NFS), kuyruk yöneticisi günlüklerini ve bir HA eşgörümlüğünü barındırır. NFS Server sunucusunun kuyruk yöneticisi yönetim ortamlarından birine de ev sahipliği yapmak için olağan bir uygulama değildir; bu, örneği basitleştirmeniz içindir. Server2 mounts Server1's queue manager logs with a standby instance. Bir WebSphere MQ MQI istemcisi, Windows XP Service Pack 2 çalıştıran ve örnek yüksek kullanılabilirlik uygulamalarını çalıştıran ek 400 MB VMware görüntülerine kurulur. Tüm sanal makineler, güvenlik nedenleriyle ilgili olarak, VMware anasisteminin bir parçası olarak yapılandırılır.

Not: Yalnızca NFS sunucusuna kuyruk yöneticisi verileri koymanız gerekir. NFSişletim sisteminde, sistemi güvenli kılmak için takma komutuyla aşağıdaki üç seçeneği kullanın:

noexec

Bu seçeneği kullanarak, ikili dosyaların NFSüzerinde çalıştırılmasını durdurarak, uzak kullanıcının sistemde istenmeyen kod çalıştırmasını önler.

nosuid

Bu seçeneği kullanarak, uzak kullanıcının daha yüksek ayrıcalıklar kazanmasını önleyen, set-user-identifier ve set-group-identifier bitlerinin kullanılmasını engelliyorsunuz.

düğümv

Bu seçeneği kullanarak, karakter ve öbek özel aygıtlarının kullanılmasını ya da tanımlanmasını engelleyerek uzak kullanıcının bir chroot hapisanesinden çıkmasını önlemenizi sağlar.

Örnek

Çizelge 30. Linuxüzerinde görsel olarak kullanılan çok eşgörünümlü kuyruk yöneticisi yapılandırması	
Server1	Server2
<i>rootolarak oturum açın</i>	
WebSphere MQ'u kurmak için IBM WebSphere MQkuruluyor içindeki yönergeleri izleyin, bu kullanıcı yoksa mqm kullanıcısını ve grubu oluşturun ve /var/mqm' yi tanımlayın. Check what uid and gid in /etc/passwd on the first machine displays for mqm; for example, mqm:x:501:100:MQ User:/var/mqm:/bin/bash Match the uid and gid for mqm in /etc/passwd on the second machine to ensure that these values are identical. Değerleri değiştirmeniz gerekiyorsa, bu makineyi yeniden başlatın.	
Dosya sisteminin çok eşgörünümlü kuyruk yöneticilerini desteklediğinden emin olmak için Paylaşılan dosya sistemi davranışını doğrulama görevini tamamlayın.	
Ortak bir klasörde (/MQHA), paylaşılacak günlük ve veri dizinlerini yaratın. Örneğin: 1. mkdir /MQHA 2. mkdir /MQHA/logs 3. mkdir /MQHA/qmgrs	Paylaşılan dosya sistemini yüklemek için /MQHAKlasörünü oluşturun. Yolu, Server1; ' te olduğu gibi tutun; örneğin: 1. mkdir /MQHA
MQHA dizinlerinin kullanıcı ve grup mqm 'in iyeliğinde olduğundan ve erişim izinlerinin kullanıcı ve grup için rwx olarak ayarlandığından emin olun; örneğin, ls -al görüntülenir, drwxrwxr-x mqm mqm 4096 Nov 27 14:38 MQDATA 1. chown -R mqm:mqm /MQHA 2. chmod -R ug+rwx /MQHA	
Kuyruk yöneticisini yarat: crtmqm -ld /MQHA/logs -md /MQHA/qmgrs QM1	
Ekle ² /MQHA *(rw, sync, no_wdelay, fsid=0) - /etc/exports	

² '*' , okuma/yazma için bu sisteme/MQHA ' ya ulaşabilen tüm makinelere izin verir. Bir üretim makinesinde erişimi kısıtlayın.

Çizelge 30. Linux üzerinde görsel olarak kullanılan çok eşgörümlü kuyruk yöneticisi yapılandırması (devamı var)

Server1	Server2
NFS yardımcı programını başlatın: <code>/etc/init.d/nfs start</code>	Mount the exported file system /MQHA: <code>mount -t nfs4 -o hard,intr Server1:/ /MQHA</code>
Kuyruk yöneticisi yapılanış ayrıntılarını Server1: içinden kopyalayın. <code>dspmqinf -o command QM1</code> ve sonucu panoya kopyalayın: <code>addmqinf -s QueueManager -v Name=QM1 -v Directory=QM1 -v Prefix=/var/mqm -v DataPath=/MQHA/qmgrs/QM1</code>	Kuyruk yöneticisi yapılanış komutunu Server2: içine yapıştırın. <code>addmqinf -s QueueManager -v Name=QM1 -v Directory=QM1 -v Prefix=/var/mqm -v DataPath=/MQHA/qmgrs/QM1</code>
Start the queue manager instances, in either order, with the -x parameter: <code>strmqm -x QM1</code> Kuyruk yöneticisi eşgörümlerini başlatmak için kullanılan komut, addmqinf komutuyla aynı IBM WebSphere MQ kurulumundan yayınlanmalıdır. Kuyruk yöneticisini farklı bir kurulumdan başlatmak ve durdurmak için, önce setmqm komutunu kullanarak kuyruk yöneticisiyle ilişkili kuruluşu ayarlamanız gerekir. Ek bilgi için bkz. setmqm .	

Linux üzerinde çok eşgörümlü kuyruk yöneticisi doğrulanıyor

Çok eşgörümlü bir kuyruk yöneticisi yapılanışını doğrulamak için **amqsgbac**, **amqspbac** ve **amqsmbac** örnek programlarını kullanın. Bu konu, Linux Red Hat Enterprise 5 üzerinde çok eşgörümlü bir kuyruk yöneticisi yapılandırmasını doğrulamak için bir örnek yapılandırma sağlar.

Yüksek kullanılabilirlik örnek programları, otomatik istemci bağlantısını kullanır. Bağlı kuyruk yöneticisi başarısız olduğunda, istemci, aynı kuyruk yöneticisi grubundaki bir kuyruk yöneticisine yeniden bağlanmayı dener. The description of the samples, Yüksek kullanılabilirlik örnek programları, demonstrates client reconnection using a single instance queue manager for simplicity. Çok eşgörümlü kuyruk yöneticisi yapılanışlarının doğrulanabilmesi için, çok eşgörümlü kuyruk yöneticileriyle aynı örnekleri kullanabilirsiniz.

Bu örnek, "Linux üzerinde çok eşgörümlü bir kuyruk yöneticisi yarat" sayfa 376 içinde açıklanan çok eşgörümlü yapılandırmayı kullanır. Çok eşgörümlü kuyruk yöneticisinin yedek yönetim ortamına geçeceğini doğrulamak için yapılandırmayı kullanın. Kuyruk yöneticisini **endmqm** komutuyla durdurun ve **-s**, geçiş (switchover) seçeneğini kullanın. İstemci programları yeni kuyruk yöneticisi örneğine yeniden bağlanır ve küçük bir gecikmeden sonra yeni yönetim ortamıyla çalışmaya devam eder.

Bu örnekte, istemci bir Windows XP Service Pack 2 sisteminde çalışır. Sistem, çok eşgörümlü kuyruk yöneticisini çalıştıran iki VMware Linux sunucusunu barındırıyor.

Verifying failover using WebSphere MQ Explorer

Hata durumunda yedek sisteme geçişi doğrulamak üzere örnek uygulamaları kullanmadan önce, her bir sunucudaki WebSphere MQ Explorer 'ı çalıştırın. **Uzak Kuyruk Yöneticisi Ekle > Doğrudan birden çok yönetim ortamı kuyruk yöneticisine bağlan** sihirbazını kullanarak her bir kaşife her iki kuyruk yöneticisi yönetim ortamını ekleyin. Her iki örneğin de çalıştığından emin olun, izin veriniz. Close the window running the VMware image with the active instance, virtually powering off the server, or stop the active instance, allowing switchover to standby instance.

Not: If you power off the server, make sure that it is not the one hosting /MQHA!

Not: Yedek yönetim ortamına geçiş yapılmasına izin ver seçeneği, **Kuyruk Yöneticisini Durdur** iletişim kutusunda bulunmayabilir. Kuyruk yöneticisi tek bir eşgörünüm kuyruğu yöneticisi olarak çalıştığı için bu seçenek eksik. Bunu, **Bekleme eşgörünümüne izin ver** seçeneği olmadan başlatmış olmanız gerekir. If your request to stop the queue manager is rejected, look at the **Ayrıntılar** window, it is possibly because there is no standby instance running.

Örnek programlar kullanılarak yedeği devreye sokma doğrulaması

Etkin yönetim ortamını çalıştırmak için bir sunucu seçin

MQHA dizinini ya da dosya sistemini barınmak için sunuculardan birini seçmiş olabilirsiniz. If you plan to test failover by closing the VMware window running the active server, make sure that it is not the one hosting MQHA!

Etkin kuyruk yöneticisi yönetim ortamını çalıştıran sunucuda

Not: Running the SVRCONN channel with the MCAUSER set to mqm, is a convenience to reduce the number of configuration steps in the example. Başka bir kullanıcı kimliği seçildiyse ve sisteminiz örnekte kullanılan farklı bir şekilde ayarlandıysa, erişim izni sorunlarıyla karşılaşabilirsiniz. Do not use mqm as a MCAUSER on an exposed system; it is likely to compromise security greatly.

1. *ipaddr1* ve *ipaddr2* seçeneklerini değiştirin ve /MQHA/hasamples.tst içinde aşağıdaki komutları kaydedin.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER('mqm') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNNAME('ipaddr1(1414),ipaddr2
(1414)') QMNAME(QM1) REPLACE
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
START LISTENER(LISTENER.TCP)
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

2. /MQHA yolu ile bir uçbirim penceresi açın ve komutu çalıştırın:

```
runmqsc -m QM1 < hasamples.tst
```

3. Verify that the listener is running and has queue manager control, either by inspecting the output of the **runmqsc** command.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

Or, using the WebSphere MQ Explorer that the TCPIP listener is running and has Control = Queue Manager.

İstemcide

1. Copy the client connection table AMQCLCHL.TAB from /MQHA/qmgrs/QM1.000/@ipcc on the server to C:\ on the client.
2. C:\ yolu ile bir komut istemi açın ve MQCHLLIB ortam değişkenini istemci kanal tanımlama çizelgesine (CCDT) işaret edecek şekilde ayarlayın.

```
SET MQCHLLIB=C:\
```

3. Komut bilgi istemine şu komutları yazın:

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

Etkin kuyruk yöneticisi yönetim ortamını çalıştıran sunucuda

1. Aşağıdakilerden birini yapın:

- Close the window running the VMware image with the active server instance.
 - WebSphere MQ Explorer olanağını kullanarak, etkin kuyruk yöneticisi yönetim ortamını durdurun ve yedek yönetim ortamına geçiş yapma ve yeniden bağ oluşturma istemcilerinin yeniden bağlanmasını bildirir.
2. En sonunda üç istemci bağlantıyı algılar ve sonra yeniden bağlansın. Bu yapılandırmada, sunucu penceresini kapadığınızda, üç bağlantının yeniden kurulması yaklaşık yedi dakika sürüyor. Bazı bağlantılar diğerlerinden önce iyi bir şekilde yeniden kuruldu.

Sonuçlar

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsgnac TARGET QM1
Sample AMQSGHAC start
message <Message 1>
message <Message 2>
message <Message 3>
message <Message 4>
message <Message 5>
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message <Message 6>
message <Message 7>
message <Message 8>
message <Message 9>
```

Çok eşgörünümlü bir kuyruk yöneticisinin silinmesi

To delete a multi-instance queue manager completely, you need to use the **dltmqm** command to delete the queue manager, and then remove instances from other servers using either the **rmvmqinf** or **dltmqm** commands.

Diğer sunucularda tanımlı eşgörünümleri olan bir kuyruk yöneticisini silmek için **dltmqm** komutunu çalıştırın; kuyruk yöneticisinin tanımlı olduğu herhangi bir sunucuda. **dltmqm** komutunu, üzerinde oluşturduğunuz sunucuda çalıştırmaz gerekir. Daha sonra, kuyruk yöneticisi tanımlaması olan diğer tüm sunucularda **rmvmqinf** ya da **dltmqm** komutunu çalıştırın.

Bir kuyruk yöneticisini yalnızca durdurulmuş olduğunda silebilirsiniz. Sildiğiniz zaman hiçbir eşgörünüm çalıştırılmadı ve kuyruk yöneticisi tek bir çoklu eşgörünüm kuyruk yöneticisi değil, yalnızca kuyruk yöneticisi verileri ve uzak bir paylaşımına ilişkin günlükleri olan bir kuyruk yöneticisidir. Bir kuyruk yöneticisini sildiğinizde, kuyruk yöneticisi verileri ve günlükleri silinir ve kuyruk yöneticisi stanza, **dltmqm** komutunu yayınladığınız sunucudaki **mq.s.in.i** dosyasından kaldırılır. Kuyruk yöneticisini sildiğinizde kuyruk yöneticisi verilerini içeren ağ paylaşımına ve günlüklere erişiminiz olmalıdır.

Kuyruk yöneticisinin önceden yaratılmış olduğu diğer sunucularda, bu sunuculardaki `mqs.ini` dosyalarında da girdiler vardır. You need to visit each server in turn, and remove the queue manager stanza by running the command **rmvmqinf** *Kuyruk yöneticisi stanza adı*.

UNIX and Linux sistemlerinde, ağ depolamaya ortak bir `mqs.ini` dosyası yerleştirdiyseniz ve her sunucuda `AMQ_MQS_INI_LOCATION` ortam değişkenini ayarlayarak tüm sunuculardan bunu gönderme yapılmışsa, güncellenecek tek bir `mqs.ini` dosyası olduğu için, bu sunuculardan yalnızca birinden kuyruk yöneticisini silmeniz gerekir.

Örnek

İlk sunucu

dltmqm QM1

Yönetim ortamlarının tanımlandığı diğer sunucular

rmvmqinf QM1ya da

dltmqm QM1

Çok eşgörünümlü bir kuyruk yöneticisinin başlatılması ve durdurulması

Bir kuyruk yöneticisinin tek bir yönetim ortamı ya da çok eşgörünümlü bir kuyruk yöneticisi olarak yapılandırılması başlıyor ve durduruluyor.

Bir sunucu çiftinde çok eşgörünümlü bir kuyruk yöneticisi tanımladığınızda, kuyruk yöneticisini tek bir eşgörünüm kuyruğu yöneticisi olarak ya da çok eşgörünümlü bir kuyruk yöneticisi olarak sunucuda çalıştırabilirsiniz.

To run a multi-instance queue manager, start the queue manager on one of the servers using the **strmqm** `-x QM1` command; the `-x` option permits the instance to failover. Bu, *etkin yönetim ortamı* olur. Yedek yönetim ortamını, aynı **strmqm** `-x QM1` komutunu kullanarak diğer sunucuda başlatın; `-x` seçeneği, yönetim ortamının yedek olarak başlamasını sağlar.

Kuyruk yöneticisi şimdi tüm istekleri işleyen etkin bir yönetim ortamıyla çalışıyor ve etkin yönetim ortamı başarısız olursa devralmaya hazır bir yedek yönetim ortamı. Etkin yönetim ortamına, kuyruk yöneticisi verileri ve günlüklerine özel erişim verilir. Beklemedeki bekleme işlemleri, kuyruk yöneticisi verilerine ve günlüklerine özel erişim izni verilmesini bekler. Yedek veritabanına dışlayıcı erişim verildiğinde, etkin yönetim ortamı olur.

Etkin yönetim ortamında **endmqm** `-s` komutunu vererek, denetimi yedek yönetim ortamına el ile de değiştirebilirsiniz. **endmqm** `-s` komutu, etkin yönetim ortamını beklemedeki yedekte kapatmadan kapatır. Kuyruk yöneticisi verileri ve günlüklerindeki paylaşmasız erişim kilidi serbest bırakılır ve bekleme işlemi sona erir.

Ayrıca, tek bir yönetim ortamı kuyruk yöneticisi olarak farklı sunucularda birden çok yönetim ortamı ile yapılandırılmış bir kuyruk yöneticisini başlatabilir ve durdurabilirsiniz. Kuyruk yöneticisini **strmqm** komutunda `-x` seçeneğini kullanmadan başlatıyorsanız, diğer makinelerde yapılandırılan kuyruk yöneticisinin eşgörünümlerinin yedek yönetim ortamları olarak başlatılması önlenir. Başka bir yönetim ortamını başlatma girişiminde bulunursanız, kuyruk yöneticisi yönetim ortamının yedek olarak çalıştırılmasına izin verilmediği yanıtı alırsınız.

If you stop the active instance of a multi-instance queue manager using the **endmqm** command without the `-S` option, then the active and standby instances both stop. Beklemedeki yönetim ortamını **endmqm** komutunu `-x` seçeneğiyle birlikte durdurursanız, bu komut bekleme kipine geçmeyi durdurur ve etkin yönetim ortamı çalışmaya devam eder. Beklemedeki `-x` seçeneği olmadan **endmqm** komutunu yayınlamayabilirsiniz.

Aynı anda yalnızca iki kuyruk yöneticisi eşgörünümü çalıştırılabilir; biri etkin yönetim ortağıysa diğeri de yedek yönetim ortağıdır. İki yönetim ortamını aynı anda başlatıyorsanız, WebSphere MQ 'da hangi yönetim ortamının etkin yönetim ortamı olduğu üzerinde denetim yoktur; bu, ağ dosya sistemi tarafından belirlenir. Kuyruk yöneticisi verilerine özel erişim elde etmek için ilk eşgörünüm etkin yönetim ortamı olur.

Not: Başarısız bir kuyruk yöneticisini yeniden başlatmadan önce, kuyruk yöneticisinin bu yönetim ortamından uygulamalarının bağlantısını kesmeniz gerekir. Bunu yapmazsanız, kuyruk yöneticisi doğru olarak yeniden başlatılamayabilir.

Paylaşılan kütük sistemi

Çok eşgörünümlü bir kuyruk yöneticisi, kuyruk yöneticisi eşgörünümlerini yönetmek için ağ üzerinde bir dosya sistemi kullanır.

Çok eşgörünümlü bir kuyruk yöneticisi, dosya sistemi kilitleri ve paylaşılan kuyruk yöneticisi verileri ve günlüklerinin bir birleşimini kullanarak hata durumunda yedek sisteme geçiş işlemini otomatikleştirir. Bir kuyruk yöneticisinin yalnızca bir yönetim ortamı, paylaşılan kuyruk yöneticisi verilerine ve günlüklere özel erişime sahip olabilir. Erişim elde edildiğinde, etkin yönetim ortamı olur. Dışlayıcı erişim elde edilmesinde başarılı olamayan diğer yönetim ortamı, kuyruk yöneticisi verileri ve günlüklere kullanılabilir duruma gelinceye kadar yedek yönetim ortamı olarak bekler.

Etkin kuyruk yöneticisi yönetim ortamı için tuttuğu kilitlerin serbest bırakılmasından ağ üzerinde çalışılan dosya sistemi sorumlu olur. Etkin yönetim ortamı bir şekilde başarısız olursa, ağ üzerinde çalışan kütük sistemi etkin yönetim ortamı için tuttuğu kilitleri serbest bırakır. Dışlayıcı kilit serbest bırakıldığında, bekleme kuyruk yöneticisi kilit girişimlerini bekliyor. Başarılı olursa, etkin yönetim ortamı olur ve kuyruk yöneticisi verilerine ve paylaşılan dosya sistemindeki günlüklere özel erişim sağlar. O zaman başlamaya devam eder.

İlgili konu, [Planlama dosya sistemi desteği](#) , dosya sisteminizin çok eşgörünümlü kuyruk yöneticilerini desteklemesini nasıl ayarlayacağını ve denetlemeyi açıklar.

Çok eşgörünümlü bir kuyruk yöneticisi, dosya sistemindeki bir başarısızlığa karşı sizi korumaz. Verilerinizi korumanız için bir dizi yol vardır.

- Yedek disk dizileri (RAID) gibi güvenilir depolama alanına yatırım yapın ve bunları ağ esnekliği olan bir ağ dosyası sistemine dahil edin.
- Diğer ortamlara WebSphere MQ doğrusal günlüklere yedekle ve birincil günlük ortamınız başarısız olursa, diğer ortamdaki günlüklere kullanarak kurtarma işlemini geri yükleyin. Bu işlemi denetlemek için bir yedek kuyruk yöneticisi kullanabilirsiniz.

Birden çok kuyruk yöneticisi eşgörünümlü

Çok eşgörünümlü bir kuyruk yöneticisi, başarısız olduktan sonra kuyruk yöneticisi kullanılabilirliğini geri yüklemek için beklemedeki bir kuyruk yöneticisi yönetim ortamını kullandığı için dirençtir.

Kuyruk yöneticisi eşgörünümlerinin eşlenmesi, kuyruk yöneticisi işlemlerinin kullanılabilirliğini artırmak için çok etkili bir yöntemdir. Basit bir kullanılabilirlik modeli kullanılarak, yalnızca şekil için: bir kuyruk yöneticisinin bir örneğinin güvenilirliği %99 (bir yıl boyunca, birikimli kapalı kalma süresi 3.65 gün), daha sonra kuyruk yöneticisinin başka bir eşgörünümlü eklenirse, kullanılabilirliği 99.99% (bir yıl boyunca, yaklaşık bir saatlik birikimli kapalı kalma süresi) artırır.

Bu, size pratik sayısal kullanılabilirlik tahminlerini vermek için çok basit bir modeldir. Kullanılabilirliği gerçekçi bir şekilde modellemek için, arızalar arasındaki ortalama süre (MTBF) ve onarım için ortalama süre (MTTR) ile arıza ve onarım süreleri arasındaki zamanın olasılık dağılımını toplamanız gerekir.

Çok eşgörünümlü kuyruk yöneticisi terimi, kuyruk yöneticisi verilerini ve günlüklerini paylaşan kuyruk yöneticisinin etkin ve beklemedeki eşgörünümlerinin birleşimini gösterir. Çok eşgörünümlü kuyruk yöneticileri, bir sunucuda kuyruk yöneticisinin bir yönetim ortamı ve başka bir sunucuda bulunan kuyruk yöneticisinin başka bir yönetim ortamı tarafından kuyruk yöneticisi işlemlerinin başarısızlığına karşı sizi koruyabilir. Bu durumda, etkin yönetim ortamının otomatik olarak başarısız olması gerekir.

Yedek sisteme geçiş ya da geçiş

Beklemedeki bir kuyruk yöneticisi yönetim ortamı istek üzerine (switchover) ya da etkin yönetim ortamı başarısız olduğunda (hata durumunda yedek sisteme geçiş) devralıyor.

- *Switchover* , etkin kuyruk yöneticisi yönetim ortamına verilmekte olan **endmqm** - s komutuna yanıt olarak bir yedek yönetim ortamı başlatıldığında gerçekleşir. You can specify the **endmqm** parameters -c, - or -p to control how abruptly the queue manager is stopped.

Not: Yedek sisteme geçiş yalnızca, bir bekleme kuyruğu yöneticisi yönetim ortamı önceden başlatıldıysa gerçekleşir. **endmqm** -s komutu, etkin kuyruk yöneticisi kilidini serbest bırakır ve yedek sisteme geçiş izni verir: beklemedeki bir kuyruk yöneticisi yönetim ortamı başlatmaz.

- **Hata Durumunda Yedek Sisteme Geçiş** , yönetim ortamı beklenmedik bir şekilde durdurulduğu için (yani, bir **endmqm** komutu verilmeden), etkin eşgörünüm tarafından tutulan kuyruk yöneticisi verileri üzerindeki kilit serbest bırakıldığında ortaya çıkar.

Yedek yönetim ortamı etkin yönetim ortamı olarak devraldığında, kuyruk yöneticisi hata günlüğüne bir ileti yazar.

Bir kuyruk yöneticisi arızalandığında ya da anahtarları değiştirdiğinde, yeniden bağlanabilir istemciler otomatik olarak yeniden bağlanmaya devam eder. İstemci yeniden bağlantı isteğini istemek için **endmqm** komutuna -r işaretini eklemenize gerek yoktur. Otomatik istemci yeniden bağlanması, Java için WebSphere MQ sınıfları tarafından desteklenmez.

Hata durumunda yedek sisteme geçiş işlemi gerçekleşmiş ve yedek yönetim ortamı etkin duruma gelse de başarısız olan bir yönetim ortamını yeniden başlatamıyorsanız, başarısız olan bir yönetim ortamını yeniden başlatamazsınız; yerel olarak başarısız olan uygulamaların başarısız olan yönetim ortamına bağlı olarak, başarısız olan yönetim ortamından bağlantısı kesilmediğini doğrulayın. Başarısız olan yönetim ortamının yeniden başlatılabildiğinden emin olmak için yerel olarak bağlı uygulamalar sona erdirir ya da başarısız bir kuyruk yöneticisi yönetim ortamından bağlantıyı keser. Paylaşılan bağ tanımları (varsayılan ayar olan) kullanılarak yerel olarak bağlı uygulamalar, yönetim ortamının yeniden başlatılmasını önlemek için başarısız olan bir eşgörünümle bağlantıda tutulan bir bağlantıdır. Yerel olarak bağlı olan uygulamaların sona erdirilmemesi ya da yerel kuyruk yöneticisi yönetim ortamı başarısız olduğunda bunların bağlantısını kesmelerini sağlamak olanaklıysa, yalıtılmış bağ tanımlarını kullanmayı göz önünde bulundurun. Yalıtılmış bağ tanımları kullanan yerel olarak bağlı uygulamalar, bağlantı kesmemeleri durumunda bile yerel kuyruk yöneticisi yönetim ortamının yeniden başlatılmasını engellemektedir.

Kanal ve istemci yeniden bağlantı

Kanal ve istemci yeniden bağlantısı, beklemedeki bir kuyruk yöneticisi yönetim ortamı etkin duruma geldikten sonra ileti işleminin geri yüklenmesinin temel bir parçasıdır.

Çok eşgörünümlü kuyruk yöneticisi eşgörünümleri, farklı ağ adreslerine sahip sunucularda kurulur. You need to configure IBM WebSphere MQ channels and clients with connection information for all queue manager instances. Bir yedek yedekleme işlemi devraldığında, istemciler ve kanallar yeni ağ adresindeki yeni etkin kuyruk yöneticisi örneğine otomatik olarak yeniden bağlanır. Otomatik istemci yeniden bağlanması, Java için WebSphere MQ sınıfları tarafından desteklenmez.

Tasarım, HA-CMP çalışması gibi yüksek düzeyde kullanılabilirlik ortamlarından farklıdır. HA-CMP, küme için sanal bir IP adresi sağlar ve adresi etkin sunucuya aktarır. WebSphere MQ yeniden bağlantısı, IP adreslerini değiştirmiyor ya da yeniden yönlendirmiyor. Kanal tanımlarında ve istemci bağlantılarında tanımladığınız ağ adreslerini kullanarak yeniden bağlanarak çalışır. Denetimci olarak, kanal tanımlamalarındaki ağ adreslerini ve istemci bağlantılarını çok eşgörünümlü tüm kuyruk yöneticilerinin tüm yönetim ortamlarına tanımlamanız gerekir. Ağ adreslerini çok eşgörünümlü bir kuyruk yöneticisine yapılandırmanın en iyi yolu bağlantıya bağlıdır:

Kuyruk yöneticisi kanalları

Kanalların CONNAME özneliği, bağlantı adları virgülle ayrılmış bir listedir; örneğin, CONNAME (' 127.0.0.1(1234) , 192.0.2.0(4321) '). Bağlantılar, bağlantı başarıyla kuruluncaya kadar, bağlantı listesinde belirtilen sırayla denir. Hiçbir bağlantı başarılı olmadıysa, kanal yeniden bağlanmayı dener.

Küme kanalları

Tipik olarak, çok eşgörünümlü kuyruk yöneticilerinin bir kümede çalışmasını sağlamak için ek yapılandırma gerekmez.

Kuyruk yöneticisi bir havuz kuyruğu yöneticisine bağlıysa, havuz kuyruk yöneticisinin ağ adresini keşfeder. Bu, kuyruk yöneticisinde CLUSRCVR kanalının CONNAME ' i anlamına gelir. TCP/IP üzerinde, kuyruk yöneticisi otomatik olarak CONNAME ögesini atarsanız, otomatik olarak ayarlar ya da boşluk olarak yapılandırır. Yedek yönetim ortamı devraldığında, IP adresi önceki etkin eşgörünümün IP adresinin yerini CONNAMEolarak değiştirir.

If it is necessary, you can manually configure AD₁ with the list of network addresses of the queue manager instances.

İstemci bağlantıları

İstemci bağlantıları, diğer bağlantıları seçmek için bağlantı listelerini ya da kuyruk yöneticisi gruplarını kullanabilir. Çok eşgörünümlü bir kuyruk yöneticisine istemci yeniden bağlanmaya ilişkin ek bilgi edinmek için “Otomatik istemci yeniden bağlantısı” sayfa 304' e bakın. İstemcilerin WebSphere MQ Sürüm 7.0.1 istemci kitaplıklarıyla ya da daha iyi bir şekilde çalışması için derlenmeleri gerekir. Bunlar en az bir Sürüm 7.0.1 kuyruk yöneticisine bağlı olmalıdır.

Hata durumunda yedek sisteme geçiş gerçekleştiğinde, yeniden bağlantı bir süre daha alır. Beklemedeki kuyruk yöneticisi başlatma işlemi tamamlamak zorundadır. Başarısız olan kuyruk yöneticisine bağlı olan istemciler bağlantı hatasını algılar ve yeni bir istemci bağlantısı başlatmaya başlar. Yeni bir istemci bağlantısı, yedek kuyruk yöneticisini yeni etkin duruma getirdiyse, istemcinin aynı kuyruk yöneticisine yeniden bağlandığı olur.

İstemci, yeniden bağlantı sırasında bir MQI çağrısının ortasındaysa, çağrı tamamlanmadan önce genişletilmiş bir bekleme toleransına sahiptir.

Hata, bir ileti kanalındaki toplu aktarma sırasında gerçekleşirse, toplu iş geri alınır ve yeniden başlatılır.

Geçiş işlemi başarısız olmaktan daha hızlıdır ve kuyruk yöneticisinin bir örneğini durdurması ve başka bir yönetim ortamını durdurması kadar uzun sürer. Yeniden yürütülecek yalnızca birkaç günlük kaydı bulunan bir kuyruk yöneticisi için, en iyi geçiş işlemi birkaç saniye içinde sıralanabilir. Hata durumunda yedek sisteme geçişin ne kadar uzun sürdüğünü tahmin etmek için, hatanın saptanması için gereken süreyi eklememeniz gerekir. Algılama en iyi şekilde 10 saniye sürer ve ağ ve dosya sistemine bağlı olarak birkaç dakika olabilir.

Uygulama kurtarma

Uygulama kurtarma işlemi, hata durumunda yedek sisteme geçiş işleminden sonra uygulamanın otomatik olarak devam ettirilmesinden Hata durumunda yedek sisteme geçiş işlemi izleyen uygulama kurtarma işlemi dikkatli bir Bazı uygulamaların hata durumunda yedek sisteme geçiş işlemi gerçekleşmiş olduğundan haberdar olması gerekir.

Uygulama kurtarmanın amacı, uygulamanın yalnızca kısa bir gecikmeyle işlemeye devam etmesi içindir. Yeni işleme devam etmeden önce, uygulama yeniden dışarı çıkmalı ve başarısızlık sırasında işlendiği iş birimini yeniden göndermelidir.

Uygulama kurtarma sorunu, WebSphere MQ MQI istemcisi ile kuyruk yöneticisi arasında paylaşılan ve kuyruk yöneticisinde saklanan bağlamı kaybetmektedir. The WebSphere MQ MQI client restores most of the context, but there are some parts of the context that cannot be reliably restored. Aşağıdaki kısımlarda, uygulama kurtarmanın bazı özellikleri ve bunların çok eşgörünümlü bir kuyruk yöneticisine bağlı uygulamaların kurtarılması nasıl etkileneceği açıklanmaktadır.

İşlem ileti alışverişi

Hata durumunda yedek sisteme geçiş perspektifinden, WebSphere MQ ileti sisteminin kalıcı özellikleri değiştirilmez. İletiler kalıcı ve iş birimleri içinde doğru olarak yönetiliyorsa, hata durumunda geçiş işlemi sırasında iletiler kaybolmaz.

Hareket işleme perspektifinden, işlemler yedek sisteme yedeklenir ya da hata durumunda yedek sisteme geçiş işleminden sonra kesinleştirilir.

Kesinleştirilmemiş hareketler geri alınır. Hata durumunda yedek denetleyiciye geçiş işleminden sonra, yeniden bağlanabilir bir uygulama, hareketin başarısız olduğunu göstermek için bir MQRC_BACKED_OUT neden kodu alır ve MQBACKkomutunu vererek işlemi geriye işlemeniz gerekir. Daha sonra işlemi yeniden başlatmanız gerekir.

Committed transactions are transactions that have reached the second phase of a two-phase commit, or single phase (message only) transactions that have begun MQCMIT .

Kuyruk yöneticisi hareket eşgüdümcarıysa ve MQCMIT , başarısızlığa başlamadan önce iki aşamalı kesinleştirmesinin ikinci aşamasına başladıysa, işlem başarıyla tamamlanır. Tamamlanma, kuyruk

yöneticisinin denetimi altındadır ve kuyruk yöneticisi yeniden çalışır durumda olduğunda devam eder. Yeniden bağlanabilir bir uygulamada, MQCMIT çağrısı olağan şekilde tamamlanır.

Yalnızca iletileri içeren tek bir aşamalı kesinleştirme işleminde, kesinleştirme işlemeye başlayan bir işlem, olağan durumda kuyruk yöneticisinin denetimi altında bir kez daha çalışır bir durumda tamamlanır. Yeniden bağlanabilir bir uygulamada MQCMIT olağan şekilde tamamlanır.

Yeniden bağlantılanabilir istemciler, hareket eşgüdücüsü olarak kuyruk yöneticisinin denetimi altında tek aşamalı işlemleri kullanabilir. Genişletilmiş işlemsel istemci yeniden bağlanmaya destek vermiyor. İşlemsel istemci bağlandığında yeniden bağlantı istenirse, bağlantı başarılı olur, ancak yeniden bağlantı kurulmadan yeniden bağlantı kurulur. Bağlantı, yeniden bağlanmamış gibi hareket eder.

Uygulamanın yeniden başlatılması ya da devam

Hata durumunda geçiş işlemi bir uygulamayı durduruyor. Bir uygulamanın başından itibaren yeniden başlayamaması ya da kesilmenin ardından işlenmeye devam edebilmesi. Bu ikincinin adı *otomatik istemci yeniden bağlantısı* olarak adlandırılır. Otomatik istemci yeniden bağlanması, Java için WebSphere MQ sınıfları tarafından desteklenmez.

Bir WebSphere MQ MQI istemcisi uygulaması ile, istemciyi otomatik olarak yeniden bağlamak için bir bağlantı seçeneği ayarlayabilirsiniz. Seçenekler şunlardır: MQCNO_RECONNECT ya da MQCNO_RECONNECT_Q_MGR . Herhangi bir seçenek belirlenmezse, istemci otomatik olarak yeniden bağlanmayı denemez ve kuyruk yöneticisi hatası MQRC_CONNECTION_BROKEN değerini istemciye döndürür. Yeni bir MQCONN ya da MQCONNX çağrısı yayınlayarak, istemciyi yeni bir bağlantı başlatmak için tasarlayabilir ve yeniden başlatabilirsiniz.

Sunucu programlarının yeniden başlatılması gerekir; kuyruk yöneticisi ya da sunucu başarısız olduğunda, bu programlar kuyruk yöneticisiyle otomatik olarak yeniden bağlanamaz. WebSphere MQ server programs are typically not restarted on the standby queue manager instance when a multi-instance queue manager instance fails.

Yedek sunucuda yeniden başlatmak için bir WebSphere MQ sunucu programını otomatik olarak otomatikleştirebilirsiniz:

1. Sunucu uygulamanızı kuyruk yöneticisi hizmeti olarak paketle. Beklemedeki kuyruk yöneticisi yeniden başlatıldığında, bu işlem yeniden başlatılır.
2. Hata durumunda yedek sisteme geçiş mantığınızı (örneğin, bir beklemedeki kuyruk yöneticisi yönetim ortamı tarafından yazılan hata durumunda yedek sisteme geçiş günlük ileti) yazmak için tetiklenip tetiklenir. Daha sonra, uygulama yönetim ortamının kuyruk yöneticisiyle bağlantı yaratmak için başlatıldıktan sonra MQCONN ya da MQCONNX adını araması gerekir.

Yedek sisteme geçiş

Bazı uygulamaların hata durumunda yedek sisteme geçmenin farkında olması gerekir, ancak diğerleri hata durumunda değildir. Bu iki örneği düşünün.

1. Bir ileti alışverişi kanalı üzerinden ileti alan ya da alan bir ileti alışverişi uygulaması, kanalın diğer ucundaki kuyruk yöneticisinin çalışır durumda olmasını gerektirmez: Kanal yönetim ortamının diğer ucundaki kuyruk yöneticisi bir yedek yönetim ortamında yeniden başlatılırsa, bu durumdan etkilenme olasılığının düşük olması beklenmez.
2. A WebSphere MQ MQI client application processes persistent message input from one queue and puts persistent message responses onto another queue as part of a single unit of work: if it handles an MQRC_BACKED_OUT reason code from MQPUT, MQGET or MQCMIT within sync point by issuing MQBACK and restarting the unit of work, then no messages are lost. Buna ek olarak, uygulamanın bir bağlantı hatasıyla başa çıkmak için özel bir işleme yapması gerekmez.

Suppose however, in the second example, that the application is browsing the queue to select the message to process by using the MQGET option, MQGMO_MSG_UNDER_CURSOR. Yeniden bağlantı, göz atma imlecini sıfırlar ve MQGET çağrısı doğru iletiyi döndürmez. Bu örnekte, uygulamanın hata durumunda yedek sisteme geçiş işlemi gerçekleştirildiğini dikkate almak gerekir. Ayrıca, imlecini altındaki ileti için başka bir MQGET yayınlamadan önce, uygulamanın göz atma imlecini geri yüklemesi gerekir.

Göz atma imlecini kaybetmek, uygulama bağlamı değişikliklerinin yeniden bağlantıyı nasıl değiştirdiğini gösteren bir örnektir. Diğer durumlar “Otomatik olarak yeniden bağlanan bir istemcinin kurtarılması” sayfa 386 içinde belgelenir.

Hata durumunda yedek sisteme geçiş işlemini izleyen WebSphere MQ MQI istemci uygulamaları için üç farklı tasarım örüntünüz vardır. Bunlardan yalnızca birinin hata durumunda yedek sisteme geçişi saptamasına gerek yoktur.

Yeniden bağlantı yok

Bu örüntüde, bağlantı kesildiğinde, uygulama yürürlükteki bağlantıdaki tüm işlemleri durdurur. Uygulamanın işlemeye devam etmesi için, kuyruk yöneticisiyle yeni bir bağlantı kurması gerekir. Uygulama, yeni bağlantıda işlemeye devam etmek için gerekli olan durum bilgilerini aktarmaktan tamamen sorumludur. Bağlantı kaybolduktan sonra kuyruk yöneticisiyle yeniden bağlantı kurulan var olan istemci uygulamaları bu şekilde yazılır.

The client receives a reason code, such as MQRC_CONNECTION_BROKEN , or MQRC_Q_MGR_NOT_AVAILABLE from the next MQI call after the connection is lost. Uygulama, kuyruk çekme noktaları gibi tüm WebSphere MQ durum bilgilerini atmalı ve yeni bir bağlantı kurmak için yeni bir MQCONN ya da MQCONNX çağrısı yayınlamalıdır ve ardından işlenmesi gereken WebSphere MQ nesnelerini yeniden açmalıdır.

Kuyruk yöneticisiyle bağlantı kesildikten sonra, varsayılan MQI davranışı kuyruk yöneticisi bağlantısı tanıtıcısı kullanılamaz duruma gelir. Varsayılan değer, hata durumunda yedek sisteme geçiş işleminden sonra uygulama yeniden bağlanmasını önlemek için MQCONNX üzerindeki MQCNO_RECONNECT_DISABULD seçeneğinin belirlenmesine eşdeğerdir.

Yedek sunucuya geçiş

Uygulamayı, hata durumunda yedek sisteme geçiş işlemi tarafından etkilenmemiş olacak şekilde yazın. Bazen, hata işlemeye dikkat etmek, hata durumunda yedek sisteme geçiş ile uğraşmak için yeterlidir.

Yeniden bağlantı etkin

Kuyruk yöneticisiyle bir MQCBT_EVENT_HANDLER olay işleyicisini kaydedin. İstemci sunucuya yeniden bağlanmayı denemeye başladığında ve başarılı bir yeniden bağlantıdan sonra MQRC_RECONNECTED ile birlikte olay işleyici MQRC_RECONNECTING ile gönderilir. Daha sonra, istemci uygulamasının işlemeye devam edebilmesi için, öngörülebilir bir durumu yeniden oluşturmak için bir yordamı çalıştırabilirsiniz.

Otomatik olarak yeniden bağlanan bir istemcinin kurtarılması

Hata durumunda yedek sisteme geçiş beklenmeyen bir olaydır ve otomatik olarak yeniden bağlanan bir istemcinin, yeniden bağlanma sonuçlarının tahmin edilebilir olması gerektiği şekilde tasarlandığı için çalışmaya devam etmesi gerekir.

Beklenmedik bir hatayı öngörülebilir ve güvenilir bir şekilde dönüştürmenin önemli bir unsuru, işlemlerin kullanımsıdır.

In the previous section, an example, “2” sayfa 385, was given of an WebSphere MQ MQI client using a local transaction to coordinate MQGET and MQPUT. İstemci bir MQRC_BACKED_OUT hatasına yanıt olarak bir MQCMIT ya da MQBACK çağrısı yayınlamalı ve yedeklenen işlemi yeniden gönderir. Kuyruk yöneticisi hatası, hareketin geriletilmesine neden olur ve istemci uygulamasının davranışı hiçbir işlemin gerçekleştirilmemesini ve hiçbir ileti kaybedilmemesini sağlar.

Tüm program durumu bir işlemin bir parçası olarak yönetilmiyor ve dolayısıyla yeniden bağlantı sonuçlarının anlaşılması daha zor hale geliyor. Kuyruk yöneticisi hata durumunda yedek sisteme geçiş işlemi için istemci uygulamanızı tasarlamak üzere, bir WebSphere MQ MQI istemcinin durumunu nasıl değiştirdiğini bilmeniz gerekir.

Özel bir hata durumunda yedek sisteme geçiş kodu olmadan uygulamanızı tasarlamaya karar verebilir ve diğer hatalarla aynı mantıkla yeniden bağlantı hatalarını ele almanıza karar verebilirsiniz. Alternatively, you might choose to recognize that reconnection requires special error processing, and register an event handler with WebSphere MQ to run a routine to handle failover. Yordam yeniden bağlanma

işlemini kendisi işleyebilir ya da ana program iş parçacığını işlemeyi sürdürürken, kurtarma işlemini gerçekleştirmesi gerektiğini belirten bir işaret belirler.

The WebSphere MQ MQI client environment is aware of failover itself, and restores as much context as it can, following reconnection, by storing some state information in the client, and issuing additional MQI calls on behalf of the client application to restore its WebSphere MQ state. Örneğin, hata noktasında açık olan nesnelerin tanıtıcıları geri yüklenir ve geçici dinamik kuyruklar aynı adla açılır. Ancak önlenemez değişiklikler var ve bu değişikliklerle başa çıkmak için tasarınıza ihtiyacınız var. Değişiklikler beş tür olarak kategorilere ayrılabilir:

1. Yeni ya da daha önce tanılanmamış hatalar, uygulama programı tarafından tutarlı bir yeni bağlam durumu geri yükleninceye kadar, MQI çağrılarına döndürülür.

Yeni bir hata alma örneği, yeniden bağlanmadan önce bağlam kaydedildikten sonra bağlam geçirilmeye çalışılırken MQRC_CONTEXT_NOT_AVAILABLE dönüş kodudur. Güvenlik bağlamı yetkisiz bir istemci programına geçirilmediğinden, yeniden bağlantı yeniden bağlandıktan sonra bağlam geri yüklenemez. Bunu yapmak için, kötü amaçlı bir uygulama programının güvenlik bağlamını almasına izin verirdi.

Genellikle, uygulamalar, sık kullanılan ve öngörülebilir hataları dikkatlice tasarlanmış bir şekilde işler ve genel bir hata işleyiciye olağan dışı hataları yeniden yakardır. Hata işleyici, WebSphere MQ bağlantısını kesebilir ve yeniden bağlayabilir ya da programı tümüyle durdurabilir. Devamlılığı artırmak için, bazı hatalarla farklı bir şekilde başa çıkabilmeye gereksinim duyabilirsiniz.

2. Kalıcı olmayan iletiler kaybolabilir.
3. Hareketler geri alındı.
4. Bir eşitleme noktası dışında kullanılanMQGET ya da MQPUT çağrıları, bir iletinin olası kaybedilmesi ile kesintiye uğrayabilir.
5. Bir MQI çağrısında uzun süre beklenmesi nedeniyle, zamanlama kaynaklı hatalar oluştu.

Kayıp bağlamla ilgili bazı ayrıntılar aşağıdaki bölümde listelenir.

- NPMCLASS (HIGH) seçeneğiyle bir kuyruğa konmadıkça, kalıcı olmayan iletiler atılır ve kuyruk yöneticisi başarısız olduğunda, kalıcı olmayan iletilerin kapanma sırasında saklanmasını kesintiye uğratmaz.
- Bir bağlantı kesildiğinde, kalıcı olmayan bir abonelik kaybedilir. Yeniden bağlanma sırasında, yeniden kurulur. Dayanıklı bir abonelik kullanmayı düşünün.
- The get-wait interval is recomputed; if its limit is exceeded it returns MQRC_NO_MSG_AVAILABLE. Benzer şekilde, abonelik süre bitimi de aynı genel süre bitimi için yeniden hesaplanmaktadır.
- Bir kuyrukta göz atma imlecinin konumu kaybedilir; genellikle ilk iletilerden önce yeniden oluşturulur.
 - MQGMO_BROWSE_MSG_UNDER_CURSOR ya da MQGMO_MSG_UNDER_CURSORbelirtenMQGET çağrıları, MQRC_NO_MSG_AVAILABLEneden koduyla başarısız olur.
 - Göz atma için kilitli olan iletiler kilitli değil.
 - Tanıtıcı kapsamı içeren işaretli iletilere göz atın imlenmez ve yeniden göz atılabilir.
 - İşlenmiş olarak işaretlenen iletilere göz atma çoğu durumda işaretsiz olarak gösterilir.
- Güvenlik bağlamı kaybedildi. Attempts to use saved message context, such as putting a message with MQPMO_PASS_ALL_CONTEXT fail with MQRC_CONTEXT_NOT_AVAILABLE.
- İleti simgeleri kaybedilir. MQGET using a message token returns the reason code MQRC_NO_MSG_AVAILABLE .

Not: *MsgId* and *CorrelId*, as they are part of the message, are preserved with the message during failover, and so MQGET using *MsgId* or *CorrelId* work as expected.

- Kesinleştirilmemiş bir harekette eşitleme noktası altında bir kuyruğa konulan iletiler artık kullanılamaz.
- İletileri mantıksal bir sırayla ya da bir ileti grubunda işleme, yeniden bağlantıdan sonra MQRC_RECONNECT_INCOMPATIBLE dönüş kodunda sonuçlanır.
- An MQI call might return MQRC_RECONNECT_FAILED rather than the more general MQRC_CONNECTION_BROKEN that clients typically receive today.

- Reconnection during an MQPUT call outside sync point returns MQRC_CALL_INTERRUPTED if the WebSphere MQ MQI client does not know if the message was delivered to the queue manager successfully. MQCMIT sırasında yeniden bağlanma da benzer şekilde davranır.
- MQRC_CALL_INTERRUPTED , başarılı bir yeniden bağlanma işleminden sonra, WebSphere MQ MQI istemcisi, kuyruk yöneticisinden herhangi bir yanıt almadıysa, bu durum başarısız olur ya da başarısız olur.
 - Bir MQPUT çağrısını eşitleme noktası dışında kullanarak kalıcı bir iletinin teslim edilmesi.
 - Bir kalıcı iletinin teslim edilmesi ya da varsayılan kalıcılık içeren bir iletinin, dış eşitleme noktası dışında bir MQPUT1 çağrısı kullanılarak sağlanması.
 - MQCMIT çağrısını kullanarak bir hareketin kesinleştirilmesini sağlar. Yanıt yalnızca başarılı bir yeniden bağlandıktan sonra geri döndürülmektedir.
- Kanallar yeni yönetim ortamları olarak yeniden başlatılır (farklı kanallar da olabilir) ve hiçbir kanal çıkış durumu saklanmaz.
- Geçici dinamik kuyruklar, geçici dinamik kuyrukları açık olan yeniden bağlanabilir olan istemcilerin kurtarıldığı sürecin bir parçası olarak geri yüklenir. Geçici dinamik kuyruklardaki hiçbir ileti geri yüklenmez, ancak kuyruğu açık olan ya da kuyruğun adını anımsamış olan uygulamalar işlemeye devam edebilmektedir.

Kuyruk, onu yaratan uygulama dışında bir uygulama tarafından kullanılıyorsa, bir sonraki başvuruda bulunulduğunda bu uygulamanın var olması için yeterince hızlı bir şekilde geri yüklenmemiş olması olasılığı vardır. Örneğin, bir istemci yanıt kuyruğu olarak geçici bir dinamik kuyruk yarattıysa ve bir kanal tarafından kuyruğa bir yanıt iletisi yerleştirilecekse, kuyruk zamanında kurtarılamayabilir. Bu durumda, kanal genellikle yanıtlama iletisini ölü mektup kuyruğuna yerleştirecektir.

Yeniden bağlanabilir istemci uygulaması ada göre geçici bir dinamik kuyruk açarsa (başka bir uygulama bunu zaten oluşturduğundan), yeniden bağlantı oluştuğunda WebSphere MQ MQI istemcisi, geçici dinamik kuyruğu yaratmak için model olmadığı için, geçici dinamik kuyruğu yeniden oluşturamaz. MQI 'da, geçici dinamik kuyruğu model temelinde yalnızca bir uygulama açabilir. Geçici dinamik kuyruğu kullanmak isteyen diğer uygulamalar MQPUT1'ya da sunucu bağ tanımlarını kullanmalı ya da başarısız olursa yeniden bağlantıyı yeniden deneyebilmelidir.

Geçici bir dinamik kuyruğa yalnızca kalıcı olmayan iletiler konabilir ve hata durumunda yedek sisteme geçiş sırasında bu iletiler kaybedilir; bu kayıp, yeniden bağlantı sırasında MQPUT1 kullanılarak geçici bir dinamik kuyruğa konabilecek iletiler için geçerlidir. Hata durumunda yedek sisteme geçiş işlemi MQPUT1 sırasında oluşursa, MQPUT1 başarılı olsa da ileti yerleştirilmeyebilir. Bu sorunun geçici çözümü kalıcı dinamik kuyruklar kullanmandır. Herhangi bir sunucu bağ tanımı uygulaması, yeniden bağlanabilir olmadığı için geçici dinamik kuyruğu ada göre açabilir.

Veri kurtarma ve yüksek kullanılabilirlik

Çok eşgörünümli kuyruk yöneticilerini kullanan yüksek kullanılabilirlik çözümleri, depolama hatasından sonra verileri kurtarmak için bir mekanizma içermelidir.

Çok eşgörünümli bir kuyruk yöneticisi, kuyruk yöneticisi işlemlerinin kullanılabilirliğini artırır, ancak kuyruk yöneticisinin iletileri saklamak için kullandığı dosya sistemi gibi diğer bileşenlerin kullanılabilirliğini ve diğer bilgileri de artırır.

Verileri yüksek düzeyde kullanılabilir duruma getirmenin bir yolu, ağa bağlı esnek veri depolaması kullanmandır. Ağ üzerinde çalışan bir dosya sistemi ve esnek veri depolama kullanarak kendi çözümünüzü oluşturabilir ya da bütünleştirilmiş bir çözüm satın alabilirsiniz. Esnekliği olağanüstü durumdan kurtarma ile birleştirmek istiyorsanız, on ya da yüzlerce kilometre boyunca disk çoğaltmasına izin veren zamanuyumsuz disk eşleme olanağı kullanılabilir.

Farklı WebSphere MQ dizinlerinin depolama ortamlarıyla eşlendiğini, ortamın en iyi şekilde kullanılmasını sağlayacak şekilde yapılandırabilirsiniz. *çoklu eşgörünüm* kuyruk yöneticileri için iki tip WebSphere MQ dizini ve dosya arasında önemli bir ayrım vardır.

Kuyruk yöneticisi örnekleri arasında paylaşılması gereken dizinler.

Bir kuyruk yöneticisinin farklı örnekleri arasında paylaşılması gereken bilgiler iki dizinde yer alıyor: qmgrp ve logs dizinleri. Dizinlerin, paylaşılan bir ağ dosya sisteminde olması gerekir. Sürekli yüksek

kullanılabilirlik ve mükemmel performans sağlayan bir depolama ortamı kullanmanız önerilir; çünkü iletiler sürekli olarak ileti olarak değişir ve silinir.

Kuyruk yöneticisi yönetim ortamları arasında paylaşılması için sahip olmayan dizinler ve dosyalar.

Diğer bazı dizinlerin, kuyruk yöneticisinin farklı eşgörünümleri arasında paylaşılması gerekmez ve ikizlenmiş bir dosya sistemi kullanılarak hızla geri yüklenmeleri gerekir.

- WebSphere MQ yürütülebilir dosyaları ve araçlar dizini. Yeniden kurarak ya da yedekleyerek ve yedeklenmiş bir dosya arşivinden geri yükleyerek değiştirin.
- Bir bütün olarak kuruluş için değiştirilen yapılandırma bilgileri. The configuration information is either managed by WebSphere MQ, such as the `mqsc.ini` file on Windows, UNIX and Linux systems, or part of your own configuration management such as **MQSC** configuration scripts. Dosya arşivini kullanarak yedekleyin ve geri yükleyin.
- İzler, hata günlükleri ve FFDC dosyaları gibi kuruluş genelinde çıkış. Dosyalar, varsayılan veri dizinindeki `errors` ve `trace` alt dizinlerinde saklanır. UNIX and Linux sistemlerindeki varsayılan veri dizini şudur: `/var/mqm`. Windows 'ta varsayılan veri dizini, WebSphere MQ kuruluş dizinidir.

Ayrıca, doğrusal günlük kaydı kullanarak çok eşgörünümlü bir kuyruk yöneticisinin düzenli ortam yedeklemelerini almak için bir yedekleme kuyruk yöneticisi de kullanabilirsiniz. Yedek kuyruk yöneticisi, ikizlenmiş dosya sisteminden olduğu kadar hızlı kurtarma sağlamıyor ve son yedeklemeden bu yana değişiklikleri kurtarmıyor. Yedek kuyruk yöneticisi mekanizması, yerleştirilmiş bir depolama hatasından sonra kuyruk yöneticisini kurtarmaktan ziyade, yerinde olmayan olağanüstü durumdan kurtarma senaryolarında kullanım için daha uygundur.

IBM WebSphere MQ Kullanılabilirlik çözümlerinin birleştirilmesi

Uygulamalar, kullanılabilirliği artırmak için diğer IBM WebSphere MQ yeteneklerini kullanır. Çok eşgörünümlü kuyruk yöneticileri, diğer yüksek kullanılabilirlik yeteneklerini tamamlar.

IBM WebSphere MQ Kümeleri kuyruk kullanılabilirliğini artırır

Küme kuyruğunda birden çok tanımlama yaratarak kuyruk kullanılabilirliğini artırabilir; kümedeki her bir yöneticide bulunan her kuyruktan birine kadar.

Kümeden bir üye üyesinin başarısız olduğunu ve daha sonra, küme kuyruğuna yeni bir ileti gönderildiğini varsayın. Unless the message *has* to go to the queue manager that has failed, the message is sent to another running queue manager in the cluster that has a definition of the queue.

Kümeler kullanılabilirliği büyük ölçüde artırsa da, iletilerin ertelenmesiyle sonuçlanan iki ilgili hata senaryosu vardır. Çok eşgörünümlü kuyruk yöneticileriyle bir küme oluşturulması, bir iletinin geciktirilmesinin olasılığını azaltır.

Marooned iletileri

Kümedeki bir kuyruk yöneticisi başarısız olursa, kümedeki diğer kuyruk yöneticilerine yönlendirilebilecek başka ileti yok, başarısız olan kuyruk yöneticisine yönlendirilir. Gönderilmekte olan iletiler, başarısız olan kuyruk yöneticisi yeniden başlatılıncaya kadar giderilir.

Zenginlikler

Benzerlik, iki ayrı ayrı hesaplama arasında paylaşılan bilgileri tanımlamak için kullanılan terimdir. Örneğin, bir uygulamanın bir sunucuya istek ileti göndermesi ile aynı uygulamanın yanıtı işlemeyi beklemesi arasında bir benzerlik vardır. Diğer bir örnek ileti dizisi, önceki iletilere bağlı olarak her iletinin işlenmesini içerir.

Kümelenmiş kuyruklara ileti gönderdiğinizde, yakınlıkları göz önünde bulundurmanız gerekir. Aynı kuyruk yöneticisine art arda iletiler göndermeli ya da her ileti kümenin herhangi bir üyeye gidebilir mi?

Kümedeki aynı kuyruk yöneticisine ileti göndermeniz gerekiyorsa ve başarısız olursa, başarısız olan küme kuyruğu yöneticisi yeniden çalıştırılıncaya kadar, iletiler gönderenin iletim kuyruğunda bekler.

Küme çok eşgörünümlü kuyruk yöneticileriyle yapılandırıldıysa, başarısız olan kuyruk yöneticisinin yeniden başlatılması için bekleme süresi, bir dakika ya da yedek bekleme süresi sona ererken sınırlandırılır. Yedekleme işlemi çalışırken, iletilerin sürdürülmesi devam ederken, yeni etkinleştirilen kuyruk yöneticisi eşgörünümlünün kanalları başlatılır ve iletim kuyruklarında bekleyen iletiler akmaya başlar.

Bir kümenin başarısız bir kuyruk yöneticisi tarafından geciktirilmekte olan iletilerin aşılması için olası bir yöntem, kümedeki her bir sunucuya iki farklı kuyruk yöneticisi konuşlandırılmalıdır ve biri etkin, diğeri ise farklı kuyruk yöneticilerinin yedek yönetim ortamı olarak sıralanır. Bu, etkin-beklemedeki bir yapılandırma ve kümenin kullanılabilirliğini artırır.

Küme yönetimi ve ölçeklenebilirliğin avantajlarından yararlanmanın yanı sıra kümeler, çok eşgörsümlü kuyruk yöneticilerini tamamlamak için ek kullanılabilirlik unsurları sağlamaya devam etmektedir. Kümeler, bir kuyruk yöneticisinin hem etkin hem de beklemedeki eşgörsümlerini etkileyen diğeri hata tiplerine karşı korur.

Kesintisiz hizmet

Küme, kesintisiz bir hizmet sağlar. Küme tarafından alınan yeni iletiler, işlenecek etkin kuyruk yöneticilerine gönderilir. Kesintisiz bir hizmet sağlamak için çok eşgörsümlü bir kuyruk yöneticisine güvenmeyin; bunun nedeni, beklemedeki kuyruk yöneticisinin hatayı saptaması ve başlatma işleminin tamamlanmasını, kanallarının yeniden bağlanmasını ve başarısız olan iletilerin yeniden gönderilmesine ilişkin zamanı almalarını sağlar.

Yerelleştirilmiş kesinti

Kabul edilebilir performansı sağlamak için milisaniye hızlarında etkileşim kurmaları gerektiği için, etkin, beklemede ve dosya sistemi sunucularının birbirinden ne kadar uzak olduğu konusunda pratik sınırlamalar söz edilir.

Kümelenmiş kuyruk yöneticileri, birçok saniye sırasının etkileşim hızlarını gerektirir ve dünyanın herhangi bir yerinde coğrafi olarak dağılır.

İşletim hatası

Kullanılabilirliği artırmak için iki farklı mekanizma kullanarak, bir insan hatası gibi bir operasyon hatasının, kullanılabilirlik çabalarınızı ortaya çıkarması olasılığını azalmanızı sağlar.

Kuyruk paylaşım grupları ileti işleme kullanılabilirliğini artırır

Yalnızca z/OS üzerinde sağlanan kuyruk paylaşım grupları, kuyruk yöneticilerinden oluşan bir grubun bir kuyruğa hizmet vermelerine olanak sağlar. Bir kuyruk yöneticisi başarısız olursa, diğeri kuyruk yöneticileri kuyruklardaki tüm iletileri işlemeye devam eder. Çok eşgörsümlü kuyruk yöneticileri, z/OS üzerinde desteklenmez ve kuyruk paylaşım gruplarını daha geniş bir ileti alışverişi mimarisinin bir parçası olarak tamamlar.

WebSphere MQ İstemcileri uygulama kullanılabilirliğini artırır

WebSphere MQ MQI istemci programları, kuyruk yöneticisi kullanılabilirliği, bağlantı ağırlıklandırma ve zenginlikler temel alınarak bir kuyruk yöneticisi grubundaki farklı kuyruk yöneticilerine bağlanabilir. Kuyruk yöneticisinin çalıştığı bir makineden farklı bir makinede bir uygulama çalıştırılarak, bir çözümün genel olarak kullanılabilirliğini artırmak için, kuyruk yöneticisi yönetim ortamı başarısız olursa, uygulamanın yeniden bağlanmanın bir yolu olduğu sürece, bir çözümün genel olarak kullanılabilirliğini artırabilirsiniz.

Kuyruk yöneticisi grupları, durdurulan bir kuyruk yöneticisinden istemcinin bağlantısını keserek istemci kullanılabilirliğini artırmak için kullanılır ve bir IP püskürtmesi gibi bir grup kuyruk yöneticisi grubunda bulunan istemci bağlantılarını dengeleme işlemi sağlar. İstemci uygulamasının, belirli bir kuyruğa bağımlılık gibi, başarısız olan kuyruk yöneticisiyle ilgili yakınlıkları yoksa ya da işleme devam edememlidir.

Otomatik istemci yeniden bağlantısı ve çok eşgörsümlü kuyruk yöneticileri, bazı benzerlikli sorunları çözerek istemci kullanılabilirliğini artırır. Otomatik istemci yeniden bağlanması, Java için WebSphere MQ sınıfları tarafından desteklenmez.

Bir istemcinin aynı kuyruk yöneticisine yeniden bağlanmasını sağlamak için, MQCNO seçeneğini MQCNO_RECONNECT_Q_MGRayarlayabilirsiniz.

1. Önceden bağlanan tek bir yönetim ortamı kuyruk yöneticisi çalışmıyorsa, kuyruk yöneticisi yeniden çalışır durumda oluncaya kadar bağlantıyı yeniden deneyip çalıştıramamaktadır.
2. Kuyruk yöneticisi çok eşgörsümlü bir kuyruk yöneticisi olarak yapılandırıldıysa, istemci hangi yönetim ortamına etkin olursa olsun yeniden bağlanır.

Aynı kuyruk yöneticisine otomatik olarak yeniden bağlanılarak, kuyruk yöneticisinin açık olduğu kuyruklar ve abone olduğu konu gibi, istemci adına tuttuğu durum bilgilerinin büyük kısmı geri yüklenir. İstemci, bir isteğe yanıt almak için dinamik bir yanıt kuyruğu açsaydı, yanıtlama kuyruğuna ilişkin bağlantı da geri yüklenir.

İletilerin kaybedilmediğinden emin olmak (günlüğe kaydetme)

WebSphere MQ , bir kuyruk yöneticisi hatasından kurtarmak için gerek duyduğunuz tüm bilgileri günlüğe kaydeder.

WebSphere MQ , bir kurtarma günlüğünde kuyruk yöneticisi tarafından denetlenen verilerde önemli olan tüm değişiklikleri kaydeder.

Bu, nesne yaratma ve silme, kalıcı ileti güncellemeleri, hareket durumları, nesne özniteliklerinde yapılan değişiklikler ve kanal etkinlikleri içerir. Günlük, ileti kuyruklarına ilişkin tüm güncellemeleri kurtarmak için gereksinim duyduğunuz bilgileri içerir:

- Kuyruk yöneticisi değişikliklerinin kayıtlarını tutma
- Yeniden başlatma işlemi tarafından kullanılmak üzere kuyruk güncellemelerinin kayıtlarını tutma
- Bir donanım ya da yazılım hatasından sonra verileri geri yükleyebilmenizi sağlar

Ancak, WebSphere MQ , dosyalarını bulunduran disk sistemini de kullanır. Disk sistemi güvenilir değilse, günlük bilgileri de içinde olmak üzere bilgiler yine de kaybedilebilir.

Günlüklerin görünüşü

Günlükler, birincil ve ikincil dosyalardan oluşur ve bir denetim dosyası içerir. Günlük dosyalarının sayısını ve boyutunu ve bunların dosya sisteminde depolandığı yeri tanımlırsınız.

Bir WebSphere MQ günlüğü iki bileşenden oluşur:

1. Günlük verileri için bir ya da daha çok dosya.
2. Günlük denetim dosyası

Günlük verileri dosyası, günlük kapsamı olarak da bilinir.

Kaydedilmekte olan verileri içeren bir dizi günlük dosyası vardır. Sayı ve boyutu (“IBM WebSphere MQ ve kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi” sayfa 410' ta açıklandığı gibi) tanımlayabilir ya da üç dosyanın sistem varsayılanını ele alabilirsiniz.

WebSphere MQ for Pencereleriçinde, üç dosyanın her biri varsayılan olarak 1 MB ' ye ayarlanır. WebSphere MQ for UNIX and Linux sistemleri için üç dosyanın her biri varsayılan değer olarak 4 MB ' ye ayarlanır.

Bir kuyruk yöneticisi yarattığınızda, tanımladığınız günlük kütüklerinin sayısı, ayrılan *birincil* günlük kütüklerinin sayısıdır. Bir sayı belirlemezseniz, varsayılan değer kullanılır.

WebSphere MQ ' da Windowsiçin günlük yolunu değiştirmediyse, günlük dosyaları dizin altında yaratılır:

```
C:\Program Files\IBM\WebSphere MQ\log\<QMgrName>
```

In WebSphere MQ for UNIX and Linux systems, if you have not changed the log path, log files are created under the directory:

```
/var/mqm/log/<QMgrName>
```

WebSphere MQ bu birincil günlük dosyalarıyla başlar, ancak birincil günlük alanı yeterli değilse, *ikincil* günlük kütüklerini ayırır. Bu, dinamik olarak bunu yapar ve günlük alanı talebi azaldığında bunları kaldırır. Varsayılan değer olarak, en çok iki ikincil günlük dosyası ayrılabilir. Bu varsayılan ayırmayı, “IBM WebSphere MQ ve kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi” sayfa 410'içinde açıklandığı gibi değiştirebilirsiniz.

Günlük denetim dosyası

Günlük denetim dosyası, büyüklük ve konum gibi günlük kütüklerinin kullanımını denetlemek için gereken bilgileri ve kullanılacak sonraki dosyanın adını içerir.

Günlük denetim dosyası, yalnızca iç kuyruk yöneticisi kullanımı içindir.

Kuyruk yöneticisi, kurtarma günlüğünün durumuyla ilişkili denetim verilerini günlük denetim dosyasında saklar ve günlük denetim dosyasının içeriğini değiştirmemelisiniz.

Not: Bir kuyruk yöneticisini başlattığınızda yaratılan günlüklerin, uygulamalarınızın işleyeceği iletilerin boyutunu ve hacmiyle uyumlu olacak kadar büyük olduğundan emin olun. Gereksinimlerinizi karşılamak için büyük olasılıkla varsayılan günlük numaralarını ve büyüklüklerini değiştirmeniz gerekir. Daha fazla bilgi için [“Günlüğün büyüklüğünün hesaplanması” sayfa 395](#) başlıklı konuya bakın.

Günlüğe kaydetme tipleri

WebSphere MQ' da, günlüğe kaydetme için gereken dosya sayısı, dosya boyutuna, aldığınız ileti sayısına ve iletilerin uzunluğuna bağlıdır. Kuyruk yöneticisi etkinliklerinin kayıtlarının bakımının iki yolu vardır: döngüsel günlük kaydı ve doğrusal günlüğe kaydetme.

Döngüsel kaydetme

Sistem durduğunda, devam etmekte olan işlemleri geri almak için günlüğü kullanarak, tüm istediğiniz kurtarma işlemini yeniden başlatmak istiyorsanız döngüsel günlük kaydı kullanın.

Döngüsel günlük kaydı, tüm yeniden başlatma verilerini bir günlük dosyaları halkasında tutar. Günlüğe kaydetme, halkadaki ilk dosyayı doldurur, sonra tüm dosyalar doluncaya kadar sonraki, vb. ' ye geçer. Daha sonra, halkadaki ilk dosyaya döner ve yeniden başlar. Bu, ürün kullanımında olduğu sürece devam eder ve günlük dosyalarından hiç çıkmayacağınız avantajına sahiptir.

WebSphere MQ , kuyruk yöneticisi verilerinin kurtarılmasını sağlamak için gerekmedikçe, kuyruk yöneticisini veri kaybı olmadan yeniden başlatmak için gereken günlük girişlerini tutar. Yeniden kullanım için günlük dosyalarını serbest bırakma mekanizması, [“Tam kurtarmanın sağlanması için denetim işaretinin kullanılması” sayfa 393](#) içinde açıklanmaktadır.

Doğrusal kaydetme

Hem yeniden başlatma kurtarma hem de ortam kurtarma (günlüğün içeriğini yeniden oynayarak kaybedilen ya da zarar gören verileri yeniden yaratmak) istiyorsanız, doğrusal günlüğe kaydetme olanağını kullanın. Doğrusal günlük kaydı, günlük verilerini sürekli bir dosya dizisi içinde tutar. Alan yeniden kullanılmaz; bu nedenle, silinmemiş herhangi bir günlük kapsamında günlüğe kaydedilen kayıtları her zaman alabilirsiniz.

Disk alanı sonlu olduğu için, bir tür arşivleme işlemi hakkında düşünmeniz gerekebilir. Günlük için disk alanınızı yönetmek, var olan alanı yeniden kullanmak ya da genişletmek için gereken şekilde bir denetim görevidir.

Doğrusal günlüğe kaydetme ile kullanılan günlük dosyalarının sayısı, ileti akışınıza ve kuyruk yöneticinizin yaşına bağlı olarak çok büyük olabilir. Ancak, *etkin* olduğu söylenen bir dizi dosya var. Etkin dosyalar, kuyruk yöneticisini yeniden başlatmak için gereken günlük girişlerini içerir. Toplu olarak etkin günlük dosyaları *etkin günlük* olarak bilinir. Etkin günlük kütüklerinin sayısı genellikle, yapılanış kütüklerinde tanımlandığı gibi, birincil günlük kütüklerinden az olur. (Numarayı tanımlamaya ilişkin bilgi için bkz. [“Günlüğün büyüklüğünün hesaplanması” sayfa 395](#) .)

Bir günlük dosyasının etkin etkin mi, yoksa bir *denetim noktası* olup olmadığını denetleyen anahtar olayı. WebSphere MQ denetim noktası, kurtarma günlüğü ve nesne dosyaları arasında tutarlılık noktasıdır. Denetim noktası, kurtarma işlemini yeniden başlatmak için gereken günlük kütüklerinin kümesini belirler. Etkin olmayan günlük dosyaları yeniden başlatma kurtarması için gerekli değildir ve etkinlik dışı olarak terimlenir. Bazı durumlarda, ortam kurtarma işlemi için etkin olmayan günlük dosyaları gerekir. (Onay işaretiyle ilgili daha fazla bilgi için bkz. [“Tam kurtarmanın sağlanması için denetim işaretinin kullanılması” sayfa 393](#) .)

Etkin olmayan günlük dosyaları, yeniden başlatma kurtarma için gerekmediği için arşivlenebilirler. Ortam kurtarma işlemi için gerekli olmayan etkinlik dışı günlük dosyaları, gereksiz günlük dosyaları olarak değerlendirilebilir. İşlemine artık ilgi göstermiyorlarsa, gereksiz günlük dosyaları silebilirsiniz. Günlük dosyalarının yok edilmesi hakkında daha fazla bilgi için [“Günlükleri yönetme” sayfa 396](#) başlıklı konuya bakın.

Yeni bir denetim noktası ikinci ya da daha sonraki bir birincil günlük dosyasına kaydedilirse, ilk dosya etkin değil ve yeni bir birincil dosya biçimlendirilir ve birincil havuzun sonuna eklenir ve günlük kaydı için kullanılabilir olan birincil dosya sayısını geri yükler. Bu şekilde, birincil günlük kütüğü havuzu, günlük kütüklerinin uzaması gereken bir listesinde yürürlükteki kütük kümesi olarak görülebilir. Yine, etkin olmayan dosyaları işleminizin gereklerine göre yönetmek için kullanılan bir denetim görevidir.

İkincil günlük dosyaları doğrusal günlüğe kaydetme için tanımlansa da, bunlar olağan işletimde kullanılmaz. Büyük olasılıkla uzun ömürlü hareketler nedeniyle bir durum ortaya çıkar, ancak yeniden başlatma için hala gerekli olabileceği için, bir dosyayı etkin havuzdan ücretsiz olarak almak mümkün değildir. İkincil dosyalar biçimlendirilir ve etkin günlük dosyası havuzuna eklenir.

Kullanılabilir ikincil dosya sayısı kullanılırsa, günlük etkinliği gerektiren en fazla işlem için istekler, uygulamaya geri döndürülmekte olan bir MQRC_RESOURCE_PROBLEM dönüş kodu ile reddedilir.

Her iki tip de günlüğe kaydetme, herhangi bir donanım arızası olmadığını varsayarak beklenmeyen güç kayblarıyla başa çıkabilir.

Tam kurtarmanın sağlanması için denetim işaretinin kullanılması

Denetim noktaları kuyruk yöneticisi verilerini ve günlük dosyalarını uyumlulaştırır ve günlük kayıtlarının atılabileceği bir tutarlılığı işaret eder. Sık denetim, kurtarma işlemi hızlandırır.

İleti kuyruklarına ilişkin kalıcı güncellemeler iki aşamada gerçekleşir. Önce, güncellemeyi gösteren kayıtlar günlüğe yazılır, daha sonra kuyruk dosyası güncellenir. Böylece günlük dosyaları, kuyruk dosyalarından daha güncel bir tarihe dönüşebilir. Yeniden başlatma işlemlerinin tutarlı bir noktadan başladığından emin olmak için, WebSphere MQ denetim noktalarını kullanır. Denetim noktası, günlükte açıklanan kaydın kuyruktaki kayıtlarla aynı olduğu bir noktadır. Denetim noktası kendisi, kuyruk yöneticisini yeniden başlatmak için gerekli olan günlük kaydı dizilerinden oluşur; örneğin, denetim noktası sırasında tüm işlemlerin durumu (iş birimleri) etkin.

WebSphere MQ , otomatik olarak denetim noktaları oluşturur.Kuyruk yöneticisi kapandığında, günlüğe kaydetme alanı düşük olduğunda ve günlüğe kaydedilen her 10 000 işlemten sonra alınır.

Kuyruklar daha fazla ileti işledikçe, denetim noktası kaydı kuyrukların yürürlükteki durumuyla tutarsız hale gelir.

WebSphere MQ yeniden başlatıldığında, günlükteki en son denetim noktası kaydını bulur. Bu bilgiler, her denetim noktasının sonunda güncellenen denetim noktası (checkpoint) dosyasında tutulur. Denetim noktası kaydı, günlük ile veri arasındaki en son tutarlılık noktasını temsil eder. kontrol noktasından bu yana gerçekleşen tüm operasyonlar ileriye dönük olarak yeniden canlandırılıyor. Bu, yeniden yürütme aşaması olarak bilinir. Yeniden yürütme aşaması, kuyrukları, sistem arızası ya da kapanmadan önce girdikleri mantıksal duruma geri getirir. Yeniden yürütme aşamasında, sistem hatası ya da sona erdirme gerçekleştiğinde uçuş sırasında olan hareketlerden oluşan bir liste oluşturulur. Messages AMQ7229 and AMQ7230 are issued to indicate the progression of the replay phase.

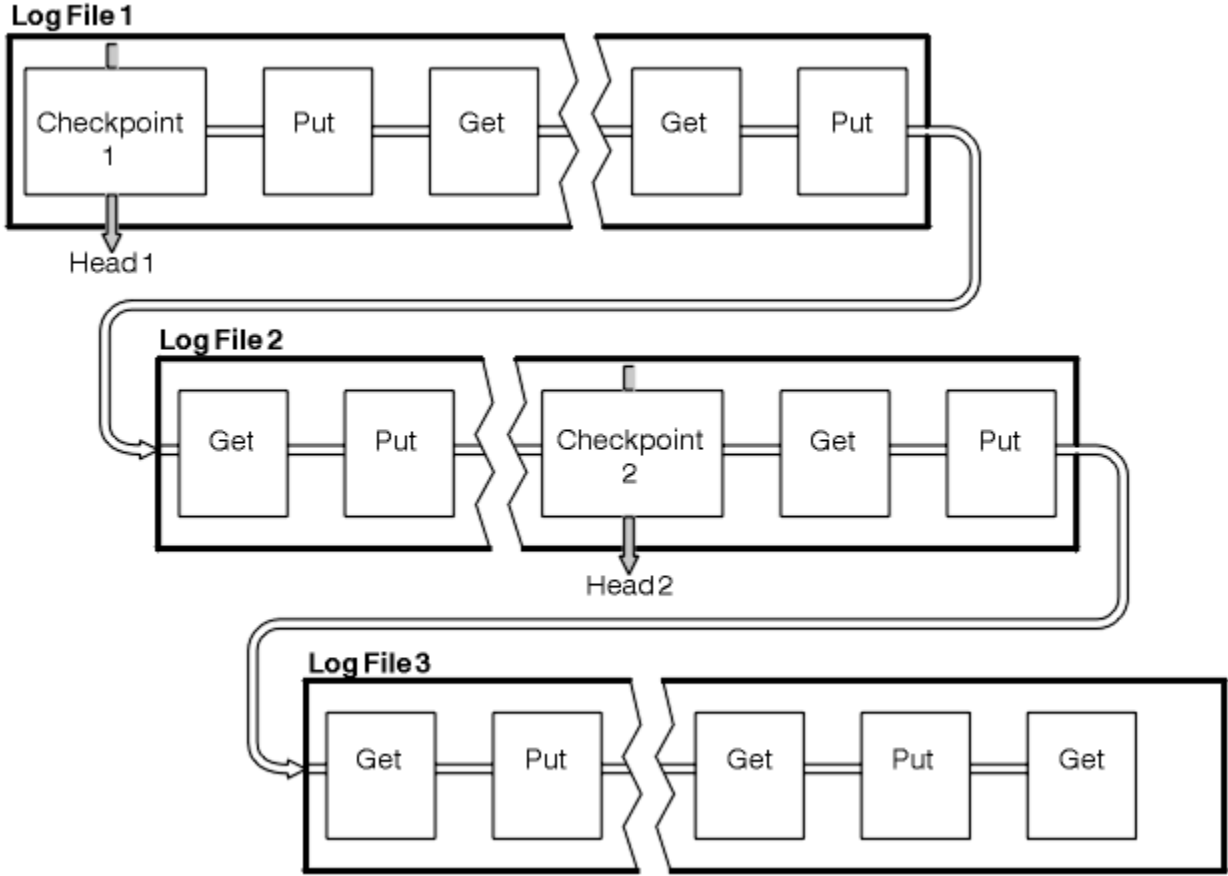
Hangi işlemlerin yedekleneceğini ya da kesinleştirileceğini bilmek için, WebSphere MQ , uçuş içi bir işlemle ilişkilendirilmiş her bir etkin günlük kaydına erişir. Bu, kurtarma aşaması olarak bilinir. Kurtarma aşamasını göstermek için AMQ7231, AMQ7232 ve AMQ7234 iletisi gönderilir.

Kurtarma aşaması sırasında gerekli tüm günlük kayıtlarına erişildiğinde, her etkin hareket çözülmeye çalışılır ve işlemle ilişkili her işlem geriletir ya da kesinleştirilir. Bu, çözüm aşaması olarak bilinir. Çözüm evresinin ilerleyişini göstermek için AMQ7233 iletisi yayınlanır.

WebSphere MQ , günlüğün başına ve kuyruğuna ilişkin iç işaretçileri tutar. Baş göstergesini, ileti verilerinin kurtarılması ile tutarlı en son denetim noktasına taşır.

Denetim noktaları, kurtarma işlemi daha verimli hale getirmek ve birincil ve ikincil günlük kütüklerinin yeniden kullanımını denetlemek için kullanılır.

Şekil 69 sayfa 394' ta, en son denetim noktasından (Checkpoint 2) önce tüm kayıtlar WebSphere MQ tarafından artık gerekli değildir. Kuyruklar, denetim noktası bilgilerinden ve daha sonraki günlük girişlerinden kurtarılabilir. Döngüsel günlüğe kaydetme için, denetim noktası yeniden kullanılabilir olmadan önce serbest bırakılan dosyalar yeniden kullanılabilir. Doğrusal bir günlük için, serbest bırakılan günlük dosyalarına artık olağan çalışma için erişilmesine ve devre dışı bırakılmasına gerek yoktur. Örnekte, kuyruk kafası göstergesi en son denetim noktasını (Checkpoint 2) gösterecek şekilde taşınır ve daha sonra yeni kuyruk başı olur, Head 2. Günlük dosyası 1 yeniden kullanılabilir.



Şekil 69. İşaretleme

Uzun süren hareketlerle işaretleme

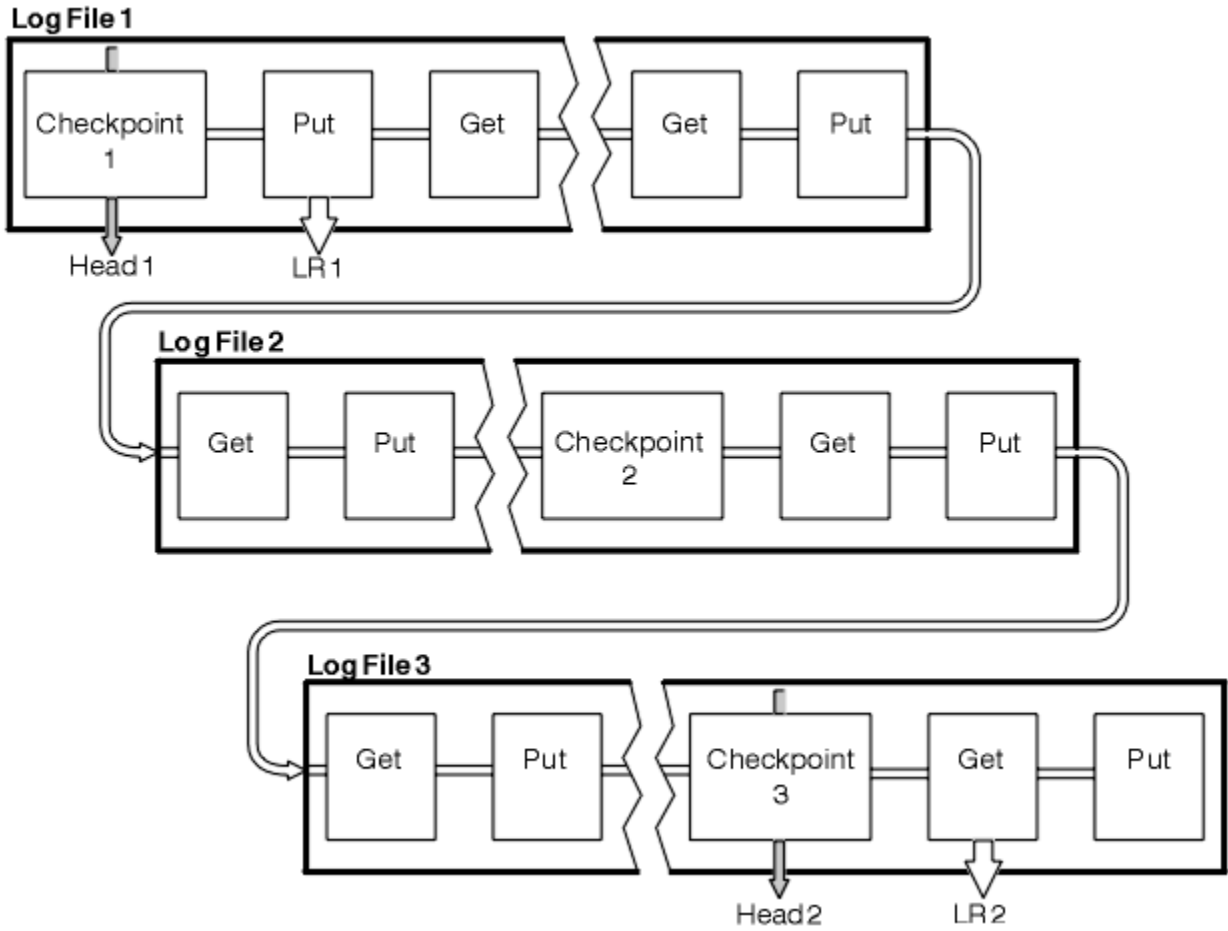
Uzun süren bir işlem günlük dosyalarının yeniden kullanımını nasıl etkiler.

Şekil 70 sayfa 395 , uzun süredir çalışan bir hareketin günlük dosyalarının yeniden kullanımını nasıl etkilediğini gösterir. Örnekte, uzun süredir çalışan bir işlem, gösterilen ilk denetim noktasından sonra, LR 1 olarak gösterilen, günlüğe bir giriş yapmış olabilir. İşlem, üçüncü denetim noktasından sonraya kadar tamamlanmaz (LR 2 numaralı noktada). LR 1 'den başlayarak tüm günlük bilgileri, tamamlanmaya kadar, ilgili işlemin kurtarılmasına izin verecek şekilde saklanır.

Uzun süren hareketin tamamlanmasından sonra, LR 2 'de günlüğün başı, en son günlüğe kaydedilen denetim noktası olan Checkpoint 3 'e taşınır. Checkpoint 3, Head 2 'den önceki günlük kayıtlarını içeren dosyalar artık gerekli değildir. Dairesel günlük kaydı kullanıyorsanız, boşluk yeniden kullanılabilir.

Birincil günlük dosyaları, uzun süren işlem tamamlanmadan önce tam olarak dolduysa, günlüklerin dolmasını önlemek için ikincil günlük dosyaları kullanılır.

Günlük başı taşındığında ve çevrimsel günlük kaydı kullanıyorsanız, birincil günlük dosyaları yeniden kullanım için uygun olabilir ve günlüğe kaydedici, geçerli dosyayı doldurduktan sonra, bu dosyanın kullanılabilir olduğu ilk birincil dosyayı yeniden kullanır. Doğrusal günlüğe kaydetme kullanıyorsanız, günlük kafası hala etkin havuz aşağı taşınır ve ilk dosya devre dışı olur. Yeni bir birincil dosya biçimlendirilir ve gelecekteki günlüğe kaydetme etkinliklerine hazır olarak havuzun alt kısmına eklenir.



Şekil 70. Uzun süredir çalışan bir işleme sahip denetim

Günlüğün büyüklüğünün hesaplanması

Bir kuyruk yöneticisi için günlük kaydı gereksinmesi büyüklüğünün hesaplanması.

Kuyruk yöneticisinin dairesel ya da doğrusal günlük kaydı kullanıp kullanmayacağına karar verdikten sonra, kuyruk yöneticisinin gereksinim duyduğu günlüğün büyüklüğünü tahmin etmek gerekir. Günlüğün büyüklüğü, aşağıdaki günlük yapılanışı değiştirgeleriyle belirlenir:

LogFileSayfaları

4K sayfaları birimlerindeki her bir birincil ve ikincil günlük dosyasının boyutu

LogPrimaryDosyaları

Önceden ayrılmış birincil günlük kütüklerinin sayısı

LogSecondaryDosyaları

Birincil günlük dosyaları dolunca kullanılmak üzere yaratılabilecek ikincil günlük dosyalarının sayısı

Çizelge 31 sayfa 396 , çeşitli işlemler için kuyruk yöneticisi günlüklerinin veri miktarını gösterir.

Kuyruk yöneticisi işlemlerinin çoğu en az miktarda günlük alanına gereksinim duyarlar. Ancak, kalıcı bir ileti bir kuyruğa konduğunda, iletinin kurtarılması için ileti verilerinin günlüğe yazılması **all** (tüm) olmalıdır. Günlüğün büyüklüğü tipik olarak, kuyruk yöneticisinin işlenmesi gereken kalıcı iletilerin sayısı ve büyüklüğünün üzerine değişir.

Çizelge 31. Günlük girişi boyutları (tüm değerler yaklaşık olarak)	
İşlem	Büyükük
Kalıcı ileti koy	750 bayt + ileti uzunluğu İleti büyükse, 261844 baytın kesimlerine bölünerek her bir kesim fazladan 300 bayt ekler.
İleti Al	260 bayt
Eşitleme noktası, kesinleştirme	750 bayt
Eşitleme noktası, geriye işleme	Her biri için 1000 bayt + 12 bayt ya da geri döndürülemeniz için
Nesne yarat	1500 bayt
Nesneyi Sil	300 bayt
Öznitelikleri değiştir	1024 bayt
Ortam görüntüsünü kaydet	800 bayt + resim Resim 260 000 baytın kesimlerine bölünerek, her bir kesim fazladan 300 bayt ekler.
Denetim Noktası	Her bir kuyruk yöneticisi için birden çok küme iletim kuyruğu kullanıyorsanız, her bir küme gönderici kanalı için her etkin iş birimi + 380 bayt için 750 bayt + 200 bayt. Başarım nedenlerinden dolayı, ek veriler kesinleştirilmemiş tüm takılar için günlüğe kaydedilebilir ya da bu veriler arabelleğe alınır. Küme gönderici kanallarınız varsa, her bir denetim noktasında küme-gönderici kanalı başına günlük günlüğüne fazladan 380 bayt yazılır.

Not:

1. Kuyruk yöneticisinin her başlatışısında birincil ve ikincil günlük kütüklerinin sayısını değiştirebilirsiniz.
2. Günlük dosyası boyutunu değiştiremezsiniz; kuyruk yöneticisini yaratmadan önce **önce** bunu belirlemeniz gerekir.
3. Birincil günlük kütüklerinin sayısı ve günlük dosyası büyüklüğü, kuyruk yöneticisi yaratıldığında önceden ayrılmış olan günlük alanı miktarını belirler.
4. The total number of primary and secondary log files cannot exceed 511 on UNIX and Linux systems, or 255 on Pencereler, which in the presence of long-running transactions, limits the maximum amount of log space available to the queue manager for restart recovery. Kuyruk yöneticisinin ortam kurtarma işlemi için gereksinim duyabilir olabileceği günlük alanı miktarı bu sınırı paylaşmaz.
5. *dairese* günlük kaydı kullanılırken, kuyruk yöneticisi birincil günlük alanını yeniden kullanır. Başka bir deyişle, kuyruk yöneticisinin günlüğü, kuyruk yöneticisinin günlüğe kaydedileceği tahmin edilen veri miktarından küçük olabilir. Kuyruk yöneticisi bir sınıra kadar, bir günlük dosyası dolu olduğunda ikincil bir günlük dosyası ayırır ve sıradaki birincil günlük dosyası kullanılamaz.
6. Birincil günlük dosyaları, denetim noktası sırasında yeniden kullanılmak üzere kullanıma sunulur. Günlük alanı miktarı düşük olduğu için, kuyruk yöneticisi denetim noktasını almadan önce hem birincil hem de ikincil günlük alanını göz önünde bulundurur.
İkincil günlük kütüklerinden daha fazla birincil günlük kütüğü tanımlamadıysanız, kuyruk yöneticisi, denetim noktası alınmadan önce ikincil günlük kütüklerini ayırabilir. Bu, birincil günlük dosyalarının yeniden kullanım için kullanılabilir olmasını sağlar.

Günlükleri yönetme

Günlükler neredeyse kendi kendini yönetir, ancak bazen alan sorunlarını çözmek için yönetilmesi gerekir.

Zaman içinde, yazılan günlük kayıtlarının bazıları kuyruk yöneticisinin yeniden başlatılması için gereksiz hale gelir. Çevrimsel günlük kaydı kullanıyorsanız, kuyruk yöneticisi günlük dosyalarında boş yer açar. Bu etkinlik kullanıcı tarafından görülmez ve ayrılan alan hızla yeniden kullanıldığından, genellikle kullanılan disk alanı miktarının azaldığını görmezsiniz.

Kuyruk yöneticisini yeniden başlatmak için, günlük kayıtlarından yalnızca son tam denetim noktasının başlangıcından bu yana yazılan kayıtlar ve etkin hareketler tarafından yazılan kayıtlar gerekir. Bu nedenle, bir denetim noktası uzun süre alınmamışsa ya da uzun süre çalışan bir işlem uzun zaman önce bir günlük kaydı yazdıysa, günlük doldurabilir. Kuyruk yöneticisi, ilk sorunu önlemek için denetim noktalarını yeterince sık almayı dener.

Uzun süreli bir işlem günlüğü doldurduğunda, günlük kayıtlarını yazma girişimi başarısız olur ve bazı MQI çağrıları MQRC_RESOURCE_PROBLEM değerini döndürür. (Alan, tüm uçuştaki hareketleri kesinleştirmek ya da geriye işlemek için ayrılmıştır, bu nedenle **MQCMIT** ya da **MQBACK** başarısız olmamalıdır.)

Kuyruk yöneticisi, çok fazla günlük alanı tüketen hareketleri geri alıyor. Bu şekilde hareket içeren bir uygulama, aynı hareket altında eşitleme noktası belirten sonraki **MQPUT** ya da **MQGET** işlemlerini gerçekleştiremez. Bu durumda bir iletiyi eşitleme noktası altına koyma ya da alma girişimi MQRC_BACKED_OUT değerini döndürür. Daha sonra uygulama, MQRC_BACKED_OUT ya da **MQBACK** değerini döndüren **MQCMIT** komutunu verebilir ve yeni bir hareket başlatabilir. Çok fazla günlük alanı tüketen hareket geriye işlendiğinde, günlük alanı serbest bırakılır ve kuyruk yöneticisi olağan şekilde çalışmaya devam eder.

Günlük doluysa, AMQ7463 iletisi yayınlanır. Ayrıca, uzun süreli bir hareket alanının serbest bırakılmasını engellediği için günlük doluysa, AMQ7465 iletisi yayınlanır.

Son olarak, kayıtlar günlüğe günlüğün işleyebileceğinden daha hızlı yazılırsa, AMQ7466 iletisi yayınlanır. Bu iletiyi görürseniz, günlük dosyalarının sayısını artırın ya da kuyruk yöneticisi tarafından işlenmekte olan veri miktarını azaltın.

Bir disk dolunca ne olur

Kuyruk yöneticisi günlüğe kaydetme bileşeni, tam bir diskle ve tam günlük dosyalarıyla başa çıkabilir. Günlük dolularını içeren disk doluysa, kuyruk yöneticisi AMQ6708 iletisini ve bir hata kaydı alır.

Günlük dosyaları, günlük kayıtları yazıldığı gibi uzatılmaktansa, büyüklük üst sınırı olarak yaratılır. Bu, WebSphere MQ 'nun yalnızca yeni bir dosya oluştururken disk alanından çıkabileceği anlamına gelir; günlüğe bir kayıt yazılırken bu alan boş alan dışında çalışamaz. WebSphere MQ , var olan günlük dosyalarındaki alanın ne kadar olduğunu her zaman bilir ve dosyalar içindeki alanı uygun şekilde yönetir.

Günlük dosyalarını içeren sürücüyü doldurursanız, bazı disk alanlarını serbest bırakmanız mümkün olabilir. Doğrusal bir günlük kullanıyorsanız, günlük dizininde etkin olmayan günlük dosyaları olabilir ve bu dosyaları başka bir sürücüye ya da aygıtta kopyalayabilirsiniz. Hala boş alanınız tükeniyorsa, kuyruk yöneticisi yapılanış kütüğündeki günlüğün yapılanışının doğru olup olmadığını denetleyin. Günlük, birincil ya da ikincil günlük kütüklerinin sayısını azaltabilir, böylece günlük kullanılabilir alanı büyütmez. Varolan bir kuyruk yöneticisine ilişkin günlük kütüklerinin büyüklüğünü değiştiremezsiniz. Kuyruk yöneticisi, tüm günlük dosyalarının aynı boyutta olduğunu varsayar.

Günlük dosyalarının yönetilmesi

Günlük dosyalarınız için yeterli yer ayırın. Doğrusal günlüğe kaydetme için, eski günlük dosyalarını artık gerekmediği zaman silebilirsiniz.

Dairesel günlük kaydı kullanıyorsanız, sisteminizi yapılandırırken günlük dosyalarını tutmak için yeterli alan olduğundan emin olun (bkz. "IBM WebSphere MQ için günlük varsayılanları" sayfa 418 ve "Kuyruk yöneticisi günlükleri" sayfa 426). Günlük tarafından kullanılan disk alanı miktarı, gerektiğinde yaratılacak ikincil kütüklere ilişkin alan da içinde olmak üzere, yapılandırılan büyüklüğün ötesinde bir değer elde etmez.

Doğrusal bir günlük kullanıyorsanız, günlük dosyaları veri günlüğe kaydedildiği gibi sürekli olarak eklenir ve kullanılan disk alanı miktarı zaman ile birlikte artar. Günlüğe kaydedilmekte olan veri hızı yüksekse, disk alanı yeni günlük dosyaları tarafından hızla kullanılır.

Zaman içinde, doğrusal bir günlüğe ilişkin eski günlük kütüklerinin artık kuyruk yöneticisini yeniden başlatmak ya da zarar görmüş nesnelere ortam kurtarmasını gerçekleştirmek zorunda kalmaması gerekir. Aşağıda, hangi günlük kütüklerinin hala gerekli olduğunu belirlemek için kullanılan yöntemler şunlardır:

Günlüğe kaydedici olay iletileri

Etkinleştirildiğinde günlüğe kaydedici olay iletileri, kuyruk yöneticileri günlük kayıtlarını yeni bir günlük dosyasına yazmaya başladıklarında oluşturulur. Günlüğe kaydedici olay iletilerinin içeriği, kuyruk yöneticisi yeniden başlatma işlemi ve ortam kurtarma için hala gerekli olan günlük dosyalarını belirtir. Günlüğe kaydedici olay iletilerine ilişkin daha fazla bilgi için [Olayları günlüğe kaydetme](#) başlıklı konuya bakın.

Kuyruk yöneticisi durumu

MQSC komutunun çalıştırılması, DISPLAY QMSTATUS ya da PCF komutu, Queue Queue Manager Durumu, gereken günlük kütüklerinin ayrıntıları da içinde olmak üzere, kuyruk yöneticisi bilgilerini döndürür. MQSC komutlarına ilişkin ek bilgi için bkz. [Script \(MQSC\) Commands](#) ve PCF komutlarıyla ilgili bilgi için [Yönetim görevlerini otomatikleştirme](#) başlıklı konuya bakın.

Kuyruk yöneticisi iletileri

Kuyruk yöneticisi belirli aralıklarla günlük kütüklerinden hangilerinin gerekli olduğunu belirtmek için bir ileti çifti yayınlar:

- AMQ7467 iletisi, kuyruk yöneticisini yeniden başlatmak için gereken en eski günlük kütüğünün adını verir. Bu günlük dosyası ve daha yeni tüm günlük dosyaları, kuyruk yöneticisinin yeniden başlatılması sırasında kullanılabilir olmalıdır.
- AMQ7468 iletisi, ortam kurtarma işlemi için gereken en eski günlük kütüğünün adını verir.

Yalnızca, kuyruk yöneticisi yeniden başlatma işlemi, etkin günlük dosyaları için gereken günlük kütüklerinin çevrimiçi olması gerekir. Etkin olmayan günlük dosyaları, olağanüstü durumdan kurtarma işlemi için manyetik bant gibi bir arşiv ortamına kopyalanabilir ve günlük dizininden kaldırılır. Ortam kurtarma işlemi için gerekli olmayan etkinlik dışı günlük dosyaları, gereksiz günlük dosyaları olarak değerlendirilebilir. İşleminize artık ilgi göstermiyorlarsa, gereksiz günlük dosyaları silebilirsiniz.

"Eski" ve "daha yeni" günlük dosyalarını belirlemek için, dosya sistemi tarafından uygulanan değişiklik sürelerini değil, günlük dosyası numarasını kullanın.

If any log file that is needed cannot be found, operator message AMQ6767 is issued. Günlük dosyasını ve sonraki tüm günlük dosyalarını kuyruk yöneticisi tarafından kullanılabilir duruma getirip işlemi yeniden deneyin.

Not: Ortam kurtarma işlemi gerçekleştirilirken, gerekli tüm günlük dosyaları günlük kütüğü dizininde aynı anda kullanılabilir olmalıdır. Tüm gerekli günlük dosyalarını tutmak için disk alanı dışında kalmamak üzere, kurtarmak isteyebileceğiniz nesnelere düzenli ortam görüntülerini aldığınızdan emin olun. Kuyruk yöneticinizdeki tüm nesnelere ilişkin bir ortam görüntüsünü almak için, aşağıdaki örneklerde gösterildiği gibi **rcdmqimg** komutunu çalıştırın:

Windows üzerinde

```
rcdmqimg -m QMNAME -t all *
```

UNIX and Linux üzerinde

```
rcdmqimg -m QMNAME -t all "*"
```

rcdmqimg çalıştırılırken, ortam günlük sıra numarası (LSN) ileri doğru hareket eder. Günlük sıra numaralarına ilişkin daha ayrıntılı bilgi için bkz. ["dmpmqlog komutunu kullanarak günlüğün içeriğini boşaltma"](#) sayfa 402. **rcdmqimg** otomatik olarak çalıştırılmaz, bu nedenle el ile ya da oluşturduğunuz bir otomatik görevden çalıştırılmalıdır. Bu komutla ilgili daha fazla bilgi için bkz. [rcdmqimg](#) ve [dmpmqlog](#).

Not: Messages AMQ7467 and AMQ7468 can also be issued at the time of running the **rcdmqimg** command.

Gereksiz günlük dosyalarının belirlenmesi

Doğrusal günlük dosyalarını yönetirken, hangi dosyaların silinebileceğini ya da arşivlenebileceğinden emin olmak önemlidir. Bu bilgiler, bu kararı vermenize yardımcı olur.

"Eski" günlük dosyalarını belirlemek için dosya sisteminin deęişiklik sürelerini kullanmayın. Yalnızca günlük kütüğü numarasını kullanın. Kuyruk yöneticisinin günlük dosyaları kullanımı, günlük dosyalarının gereksinim duyulmadan ayrılması ve biçimlendirilmesi de dahil olmak üzere karmaşık kuralları izler. Görelî yaşı saptamak için bu zamanları kullanmaya çalışırsanız, deęişiklik süreleriyle ilgili günlük dosyalarını görmeniz gerekebilir.

Kuyruk yöneticisini yeniden başlatmak için gereken en eski günlük kütüğünü saptamak için, DISPLAY QMSTATUS RECLOGkomutunu verin.

Ortam kurtarma işlemini gerçekleştirmek için gereken en eski günlük kütüğünü saptamak için, DISPLAY QMSTATUS MEDIALOGkomutunu verin.

Genel olarak daha düşük bir günlük dosyası numarası daha eski bir günlüğü belirtir. 10 yıl boyunca her gün 3000 günlük dosyası sipariş eden çok yüksek bir günlük dosya cironuz yoksa, bu durumda 9 999 999 'da satır kapma işlemi için kater' e gerek yoktur. Bu durumda, herhangi bir günlük dosyasını RECLOG değerinden küçük bir sayıyla arşivleyebilir ve RECLOG ve MEDIA (ORTAM) değerlerinden daha az sayıda günlük kütüğü silebilirsiniz.

Ancak, çok yüksek bir günlük dosyası cirosu varsa veya dięer bir şekilde genel vakayla başa çıkmaktan emin olmak istiyorsanız, aşağıdaki algoritma genellikle şu şekilde kullanılabilir:

D == yeniden başlatma günlük dosyası numarası
(QMSTATUS RECLOG GÖRÜNTÜSÜNDEN)

Let M == ortam kurtarma günlüğü dosya numarası
(QMSTATUS MEDIALOG ' ı GÖRÜNTÜLE)

Silme ya da arşivleme için hak kazanma hakkı olan bir günlük dosyası numarasına izin ver
Bunun belirlenmesi gerekiyor.

```
işlev minlog (a, b) {  
  if (abs (a-b) < 5000000)  
    return min (a, b); # Not wrapped.  
  else  
    dönüş maks (a, b); # Wrapped. }
```

Bir günlük dosyası L silinebilir ise
(L! = S & & L! = M & & minlog (L, minlog (S, M)) == L).

L günlük dosyası L arşivlenirse arşivlenebilir
(L! = S & & minlog (L, S) == L).

Günlük dosyası yeri

When choosing a location for your log files, remember that operation is severely affected if WebSphere MQ fails to format a new log because of lack of disk space.

Dairesel bir günlük kullanıyorsanız, en az yapılandırılmış birincil günlük dosyaları için sürücüde yeterli yer olduğundan emin olun. Ayrıca, günlük büyümek zorunda bırakılırsa, en az bir ikincil günlük dosyası için yer bırakın.

Doğrusal bir günlük kullanıyorsanız, çok daha fazla alana izin verin; veriler günlüğe kaydedildikçe, günlük tarafından tüketilen alan sürekli olarak artar.

İdeal olarak, günlük dosyalarını, kuyruk yöneticisi verilerinden ayrı bir disk sürücüsüne yerleştirin. Bunun performans açısından avantajları vardır. Günlük dosyalarının birden çok disk sürücüsüne ikizlenen bir düzenleme içinde yerleştirilmesi de mümkün olabilir. Bu, günlüğün bulunduğu sürücünün hatasına karşı korur. İkizleme olmadan, WebSphere MQ sisteminizin son yedeğine geri dönmek zorunda kalırdınız.

Kurtarma için günlüğü kullanma

Başarısızlıklardan kurtulmak için günlüklerin kullanılması.

Verilerinizin zarar görebilmesi için birkaç yol vardır. WebSphere MQ , aşağıdakileri kurtarmanıza yardımcı olur:

- Zarar görmüş bir veri nesnesi
- Sistemdeki güç kaybı
- Bir iletişim hatası

Bu kısım, günlüklerin bu sorunları çözmek için nasıl kullanılırca kullanıldığını gösteriyor.

Güç kaybı ya da iletişim hatalarından kurtarma

WebSphere MQ , hem iletişim hatalarından, hem de güç kaybından kurtarılabilir. Buna ek olarak, bazen bir dosyanın yanlışlıkla silinmesi gibi diğer sorun tiplerinden de kurtarılabilir.

İletişim başarısızlığı durumunda, iletiler bir alma uygulaması tarafından kaldırılincaya kadar kuyruklarda kalır. İletim iletiliyorsa, ileti başarıyla iletilinceye kadar iletim kuyruğunda kalır. Bir iletişim hatasından kurtulmak için, başarısız olan bağlantıyı kullanarak kanalları yeniden başlatabilirsiniz.

Gücü kaybedersen, kuyruk yöneticisi yeniden başlatıldığında WebSphere MQ kuyrukları, hata sırasında kesinleştirilen durumlarına geri yükler. Bu, kalıcı iletilerin kaybolmamasını sağlar. Kalıcı olmayan iletiler atılır; WebSphere MQ aniden durduğunda, bunlar hayatta kalmazlar.

Hasarlı nesnelere kurtarılıyor

Örneğin, bir IBM WebSphere MQ nesnesinin kullanılmadığı yollar vardır; örneğin, yanlışlıkla zarar görmemesi nedeniyle. Daha sonra, tüm sisteminizi ya da bir kısmını kurtarmanız gerekir. Gereken işlem, hasar saptandığında, seçilen günlük yönteminin ortam kurtarmasını destekleyip desteklemediği ve hangilerinin zarar gördüğünün saptanması gerekir.

Ortam kurtarma

Ortam kurtarma işlemi, doğrusal bir günlüğe kaydedilen bilgilerden nesnelere yeniden oluşturur. Örneğin, bir nesne dosyası yanlışlıkla silinirse ya da başka bir nedenle kullanılamaz duruma gelirse, ortam kurtarma işlemi bunu yeniden yaratabilir. Bir nesneye ilişkin ortam kurtarma işlemi için gerekli olan günlükteki bilgiler *ortam görüntüsü* olarak adlandırılır.

Ortam görüntüsü, nesnenin kendisinin yeniden yaratılacağı nesnenin resmini içeren bir günlük kaydı dizidir.

Bir nesneyi yeniden yaratmak için gereken ilk günlük kaydı, *ortam kurtarma kaydı* olarak bilinir; bu, nesnenin en son ortam görüntüsünün başlangıcıdır. Her nesnenin ortam kurtarma kaydı, denetim noktası sırasında kaydedilen bilgi parçalarından biridir.

Bir nesne ortam görüntüsünden yeniden yaratıldığında, son görüntünün alındığı andan itibaren nesnede gerçekleştirilen güncellemeleri tanımlayan günlük kayıtlarını yeniden yürütmek de gerekir.

Örneğin, kalıcı bir iletinin kuyruğa konmasından önce, kuyruk nesnesinin görüntüsünü içeren bir yerel kuyruk gibi düşünün. Nesnenin en son görüntüsünü yeniden yaratmak için, görüntünün kendisini yeniden oynamanın yanı sıra, iletinin yerleştirmesini kuyruğa koyarak günlüğe kaydetme girişlerinin yeniden yürütülmesi gereklidir.

Bir nesne yaratıldığında, yazılan günlük kayıtları nesneyi tamamen yeniden yaratmak için yeterli bilgi içerir. Bu kayıtlar, nesnenin ilk ortam görüntüsünü oluşturan kayıtlardan birini içerir. Ardından, her bir kapanma sırasında, kuyruk yöneticisi ortam görüntülerini otomatik olarak şu şekilde kaydeder:

- Tüm süreç nesnelere görüntüleri ve yerel olmayan kuyruklar
- Boş yerel kuyruklara ilişkin görüntüler

Ortam görüntüleri, `rcdmqimg` içinde açıklanan `rcdmqimg` komutu kullanılarak el ile de kaydedilebilir. Bu komut, IBM WebSphere MQ nesnesine ilişkin bir ortam görüntüsünü yazar. Bir ortam görüntüsü yazıldığında, yalnızca ortam görüntüsünü tutan günlükler ve bu zamandan sonra oluşturulan tüm günlükler zarar görmüş nesnelere yeniden oluşturmak için gereklidir. Ortam görüntülerini yaratmanın yararı, kullanılabilir serbest bellek miktarı ve günlük dosyalarının oluşturulma hızı gibi etkenlere bağlıdır.

Ortam görüntülerinin kurtarılıyor

Kuyruk yöneticisi, kuyruk yöneticisinin başlatılması sırasında bazı nesnelere ortam görüntülerinden otomatik olarak kurtarır. Kuyruk yöneticisi son kapanırken eksik olan herhangi bir harekette yer alırsa ve yeniden başlatma işlemi sırasında bozuk ya da zarar görmüş bir harekette yer alırsa otomatik olarak bir kuyruğu kurtarır.

IBM WebSphere MQ nesnesini yeniden yaratmak için günlükteki kayıtları yeniden oynayan **rcrmqobj** komutunu kullanarak diğer nesnelere el ile kurtarmanız gerekir. Nesne, görüntünün saklandığı süre ile yeniden yaratma komutunun yayınlandığı zaman arasındaki tüm geçerli günlük olaylarıyla birlikte, günlükte bulunan en son görüntüden yeniden yaratılır. Bir IBM WebSphere MQ nesnesi zarar görürse, gerçekleştirilebilecek tek geçerli işlemler ya silme işlemi ya da bu yöntemle yeniden yaratılabilir. Kalıcı olmayan iletiler bu şekilde kurtarılamaz.

rcrmqobj komutunun daha ayrıntılı bilgileri için bkz. [rcrmqobj](#) .

Ortam kurtarma kaydını içeren günlük dosyası ve sonraki tüm günlük dosyaları, bir nesnenin ortam kurtarmayı denerken günlük dosyası dizininde kullanılabilir olmalıdır. Gerekli bir dosya bulunamazsa, işletmen iletisi AMQ6767 yayınlanır ve ortam kurtarma işlemi başarısız olur. Yeniden yaratmak istediğiniz nesnelere düzenli ortam görüntülerini almazseniz, bir nesneyi yeniden yaratmak için gereken tüm günlük kütüklerini tutmak için disk alanınız yetersiz olabilir.

Başlatma sırasında hasarlı nesnelere kurtarılıyor

Kuyruk yöneticisi başlatma sırasında zarar görmüş bir nesneyi keşfederse, gereken işlem nesne tipine ve kuyruk yöneticisinin ortam kurtarma işlemini destekleyecek şekilde yapılandırılıp yapılandırılmadığına bağlıdır.

Kuyruk yöneticisi nesnesi zarar görmüşse, nesne kurtarılamadıkça, kuyruk yöneticisi başlayamaz. Kuyruk yöneticisi doğrusal bir günlükle yapılandırıldıysa ve ortam kurtarma işlevini destekliyorsa, IBM WebSphere MQ otomatik olarak kuyruk yöneticisi nesnesini ortam görüntülerinden yeniden yaratmayı dener. Seçilen günlük yöntemi ortam kurtarmayı desteklemiyorsa, kuyruk yöneticisi yedeğini geri yükleyebilir ya da kuyruk yöneticisini silebilirsiniz.

Kuyruk yöneticisi durduğunda herhangi bir işlem etkinleştirilmişse, bu işlemlerin içinde kalıcı, kesinleştirilmemiş iletileri içeren ya da var olan yerel kuyruklar kuyruk yöneticisini başarıyla başlatmak için de gereklidir. Bu yerel kuyruklardan herhangi birinin zarar göreceği bulunursa ve kuyruk yöneticisi ortam kurtarmayı destekliyorsa, otomatik olarak bu kuyrukların ortam görüntülerinden yeniden yaratılmasını sağlar. Kuyruklardan herhangi biri kurtarılamazsa, IBM WebSphere MQ başlayamaz.

Ortam kurtarmayı desteklemeyen bir kuyruk yöneticisinde başlatma işlemi sırasında kesinleştirilmemiş iletiler içeren zarar görmüş herhangi bir yerel kuyruk saptanırsa, kuyruklar zarar görmüş nesnelere olarak işaretlenir ve bunlar üzerinde kesinleştirilmemiş iletiler yoksaılır. Bu durum, bu tür bir kuyruk yöneticisinde zarar görmüş nesnelere ortam kurtarılması gerçekleştirilmenin mümkün olmadığı ve geriye kalan tek işlem bunların silinmesine neden olmasıdır. Herhangi bir hasarı bildirmek için AMQ7472 iletisi yayınlandı.

Zarar görmüş nesnelere diğer zamanlarında kurtarılıyor

Nesnelere ortam kurtarma işlemi yalnızca başlatma sırasında otomatik olur. At other times, when object damage is detected, operator message AMQ7472 is issued and most operations using the object fail. Kuyruk yöneticisi başlatıldıktan sonra kuyruk yöneticisi nesnesi herhangi bir zaman zarar görmüşse, kuyruk yöneticisi önceden bir kapatma işlemi gerçekleştirir. Bir nesne zarar gördüğünde bunu silebilirsiniz ya da kuyruk yöneticisi doğrusal bir günlük kullanıyorsa, **rcrmqobj** komutunu kullanarak bu nesneyi ortam görüntüden kurtarmayı deneyin (ek ayrıntılar için [rcrmqobj](#) başlıklı konuya bakın).

IBM WebSphere MQ günlük dosyalarını koruma

Kuyruk yöneticisi çalışırken günlük dosyalarına dokunmayın, kurtarma işlemi olanaksız olabilir. Günlük dosyalarını yanlışlıkla değiştirilmeye karşı korumak için süper kullanıcı ya da mqm yetkisi kullanın.

Bir IBM WebSphere MQ kuyruk yöneticisi çalışırken etkin günlük kütüklerini el ile kaldırmayın. If a user inadvertently deletes the log files that a queue manager needs to restart, IBM WebSphere MQ **bunu yapmaz** issue any errors and continues to process data *kalıcı iletiler dahil*. Kuyruk yöneticisi olağan bir şekilde kapatılır, ancak yeniden başlatılamayabilir. İletilerin kurtarılması imkansız hale gelir.

Etkin bir kuyruk yöneticisi tarafından kullanılan günlükleri kaldırma yetkisine sahip olan kullanıcılar, diğer önemli kuyruk yöneticisi kaynaklarını (kuyruk dosyaları, nesne kataloğu ve IBM WebSphere MQ yürütülebilir dosyaları gibi) silme yetkisine de sahiptir. Bu nedenle, IBM WebSphere MQ ' un kendisini koruyamayacağı bir şekilde çalışan ya da atıl durumda bir kuyruk yöneticisinde zarar görebilir, ancak bu nedenle zarar görebilir.

Süper kullanıcı ya da mqm yetkisi verilirken dikkatli olun.

dmpmqlog komutunu kullanarak günlüğün içeriğini boşaltma

Kuyruk yöneticisi günlüğünün içindekileri atmak için dmpmqlog komutunu nasıl kullanacağını.

Kuyruk yöneticisi günlüğünün içindekileri atmak için dmpmqlog komutunu kullanın. Varsayılan olarak tüm etkin günlük kayıtları atılır; yani, komut günlüğün başından başlayarak dökümü başlatır (genellikle tamamlanan son denetim noktasının başlangıçlarıdır).

Günlük genellikle, kuyruk yöneticisi çalışmadığında atılabilir. Kuyruk yöneticisi sona erdirme sırasında bir denetim noktası devraldığı için, günlüğün etkin bölümü genellikle küçük bir sayıda günlük kaydı içerir. Ancak, dökümün başlangıç konumunu değiştirmek için, aşağıdaki seçeneklerden birini kullanarak daha fazla günlük kaydı dökümünü almak için dmpmqlog komutunu kullanabilirsiniz.

- Günlüğün *tabanından* dökümü başlatın. Günlüğün tabanı, günlüğün başını içeren günlük kütüğündeki ilk günlük kayıttır. Bu durumda atılmış olan ek veri miktarı, günlüğün başındaki dosyanın günlük dosyasına nereye yerleştirileceği bağlıdır. Günlük dosyasının başlangıcınıysa, yalnızca küçük bir miktar ek veri atılır. Baş, günlük dosyasının sonuna yakınsa, önemli ölçüde daha fazla veri atılır.
- Dökümün başlangıç konumunu tek bir günlük kaydı olarak belirtin. Her bir günlük kaydı, benzersiz bir *günlük sıra numarası (LSN)* ile tanımlanır. Döngüsel günlüğe kaydetme durumunda, bu başlangıç günlük kaydı, günlüğün tabanından önce olamaz; bu kısıtlama, doğrusal günlükler için geçerli değildir. Komutu çalıştırmadan önce etkin olmayan günlük dosyalarını yeniden yürürlüğe almanız gerekebilir. Başlangıç konumu olarak, önceki dmpmqlog çıkışından alınan geçerli bir LSN belirtmeniz gerekir.

Örneğin, doğrusal günlüğe kaydetme ile, son dmpmqlog çıkışınızdan nextlsn ' ı belirtebilirsiniz. nextlsn , Log File Header içinde görünür ve yazılacak sonraki günlük kaydının LSN ' lerini gösterir. Günlüğün son atılmasından bu yana yazılan tüm günlük kayıtlarını biçimlemek için bu seçeneği bir başlangıç konumu olarak kullanın.

- **Yalnızca doğrusal günlükler için**, dmpmqlog ' a belirli bir günlük dosyası kapsamında günlük kayıtlarını biçimlendirme işlemi başlatmasını yönebilirsiniz. Bu durumda dmpmqlog , etkin günlük dosyalarıyla aynı dizinde bulunan bu günlük dosyasını ve birbirini izleyen her birini bulmayı bekler. Bu seçenek dairesel günlükler için geçerli değildir; burada dmpmqlog , günlüğün tabanından önceki günlük kayıtlarına erişemez.

The output from the dmpmqlog command is the Log File Header and a series of formatted log records. Kuyruk yöneticisi, verilerde yapılan değişiklikleri kaydetmek için birkaç günlük kaydı kullanır.

Biçimlendirilen bilgilerin bazıları yalnızca dahili olarak kullanılır. Aşağıdaki listede en yararlı günlük kayıtları yer alır:

Günlük Dosyası Üstbilgisi

Her bir günlük, her zaman dmpmqlog komutu tarafından biçimlendirilen ilk iş dosyası üstbilgisine sahiptir. Aşağıdaki alanları içerir:

<i>oturum açma işlevi</i>	Birincil günlük kapsamlarının sayısı.
<i>oturum açma</i>	İkincil günlük kapsamlarının sayısı.
<i>logsize</i>	Kapsam başına 4 KB ' lik sayfa sayısı.
<i>baselsn</i>	Günlük kapsamının bulunduğu günlük kapsamındaki ilk LSN ' dir.

<i>nextlsn</i>	Yazılacak sonraki günlük kaydının LSN ' si.
<i>headlsn</i>	Günlüğün başındaki günlük kaydının LSN ' si.
<i>terzn</i>	Günlüğün kuyruk konumunu tanımlayan LSN ' dir.
<i>hflag1</i>	Günlüğün DAIRESEL ya da LOG RETAIN (doğrusal) olup olmadığı.
<i>HeadExtentID</i>	Günlüğün başını içeren günlük kaptandır.

Günlük Kaydı Üstbilgisi

Günlük içindeki her bir günlük kaydının, aşağıdaki bilgileri içeren bir sabit üstbilgisi vardır:

<i>LSN</i>	Günlük sıra numarası.
<i>LogRecdTipi</i>	Günlük kaydının tipi.
<i>XTranid</i>	Bu günlük kaydıyla ilişkili işlem tanıtıcısı (varsa). Bir MQI <i>TranType</i> , yalnızca bir WebSphere MQişlemini belirtir. XA 'nın bir <i>TranType</i> ' i diğer kaynak yöneticileriyle birlikte. Aynı iş birimi içinde yer alan güncellemeler aynı <i>XTranid</i> ' a sahiptir.
<i>QueueName</i>	Bu günlük kaydıyla ilişkili kuyruk (varsa).
<i>Qid</i>	Kuyruğa ilişkin benzersiz iç tanıtıcı.
<i>PrevLSN</i>	Aynı hareket içinde önceki günlük kaydının LSN 'si (varsa).

Kuyruk Yöneticisini Başlat

Bu günlük, kuyruk yöneticisinin başlattığı günlüğe kaydeder.

<i>StartDate</i>	Kuyruk yöneticisinin başlatılmış olduğu tarih.
<i>StartTime</i>	Kuyruk yöneticisinin başlatılmış olduğu saat.

Kuyruk Yöneticisini Durdur

Kuyruk yöneticisinin durduğu bu günlükler.

<i>StopDate</i>	Kuyruk yöneticisinin durduğu tarih.
<i>StopTime</i>	Kuyruk yöneticisinin durduğu saat.
<i>ForceFlag</i>	Kullanılan sona erdirme tipi.

Denetim Noktası Başlat

Bu, kuyruk yöneticisi denetim noktasının başlangıcını gösterir.

Denetim Denetim Noktası

Bu, kuyruk yöneticisi denetim noktasının sonunu belirtir.

<i>ChkPtLSN</i>	Bu denetim noktasını başlatan günlük kaydının LSN ' si.
-----------------	---

İleti koy

Bu, kalıcı bir iletinin kuyruğa konmasını günlüğe kaydeder. İleti eşitleme noktası altına konduysa, günlük kaydı üstbilgisi boş değerli olmayan bir *XTranid* içerir. Kaydın geri kalan kısmı aşağıdakileri içerir:

<i>MapIndex</i>	Kuyruktaki iletiye ilişkin tanıtıcı. Bu ileti, kuyruktan bu iletiyi almak için kullanılan karşılık gelen MQGET ' a eşleştirmek için kullanılabilir. Bu durumda, aynı QueueName ve MapIndex' yi içeren sonraki bir <i>Get Message</i> günlük kaydı bulunabilmektedir. Bu noktada, MapIndex tanıtıcısı o kuyruğa sonraki bir ileti konulması için yeniden kullanılabilir.
<i>Veri</i>	Bu günlük kaydına ilişkin onaltılı dökümden, ileti tanımlayıcısı (eyecatcher MD) ve ileti verilerinin kendisini izleyen çeşitli iç veriler bulunur.

Bölüm Koy

Tek bir günlük kaydı için çok büyük olan kalıcı iletiler, birden çok *Put Part* günlük kaydı ve ardından tek bir *Put Message* kaydı olarak günlüğe kaydedilir. *Put Part* kayıtları varsa, *PrevLSN* alanı, *Put Part* kayıtlarını ve son *Put Message* kaydını birlikte zincirleyecek.

Veri Önceki günlük kaydının kapalı olduğu ileti verilerini devam ettirir.

İleti Al

Yalnızca kalıcı iletilerin kaydedildiği günlüğe kaydedilir. İleti eşitleme noktası altındaysa, günlük kaydı üstbilgisi boş değerli olmayan bir *XTranid* içerir. Kaydın geri kalan kısmı aşağıdakileri içerir:

MapIndex Kuyruktan alınan iletiyi tanımlar. Aynı *QueueName* ve *MapIndex* değerini içeren en son *Put Message* günlük kaydı, alınan iletiyi tanımlar.

QPriority Kuyruktan alınan iletinin önceliği.

İşlemi Başlat

Yeni bir işlemin başlangıcını belirtir. MQI ' nin TranType değeri, yalnızca bir WebSphere MQ işlemi olduğunu belirtir. XA 'nın TranType tipi, diğer kaynak yöneticilerini içeren bir XA' ya işaret eder. Bu işlem tarafından yapılan tüm güncellemeler aynı *XTranid* ' e sahip olacaktır.

İşlemi Hazırla

Kuyruk yöneticisinin belirtilen *XTranid* ile ilişkili güncellemeleri kesinleştirmeye hazır olduğunu gösterir. Bu günlük kaydı, diğer kaynak yöneticilerini içeren iki aşamalı kesinleştirmenin bir parçası olarak yazılır.

İşlemi Kesinleştir

Kuyruk yöneticisinin bir hareket tarafından yapılan tüm güncellemeleri kesinleştirdiğini gösterir.

İşlemi Geri Al

Bu, kuyruk yöneticisinin bir işlemi geri alma niyetini belirtir.

İşlemi Sona Son

Bu, geriye işlenen bir hareketin sonunu belirtir.

Hareket Tablosu

Bu kayıt, eşitleme noktası sırasında yazılır. Kalıcı güncellemeleri yapan her işlemin durumunu kaydeder. Aşağıdaki bilgilerin kaydedildiği her işlem için:

<i>XTranid</i>	İşlem tanıtıcısı.
<i>FirstLSN</i>	İşlemlle ilişkili ilk günlük kaydının LSN ' si.
<i>LastLSN</i>	İşlemlle ilişkili son günlük kaydının LSN ' si.

İşlem Katılımcıları

Bu günlük kaydı, kuyruk yöneticisinin XA Transaction Manager bileşeni tarafından yazılır. İşlemlere katılan dış kaynak yöneticilerini kaydeder. Aşağıdaki her katılımcı için aşağıdakilerin kaydedildiği kaydedilir:

<i>RMName</i>	Kaynak yöneticisinin adı.
<i>RMID</i>	Kaynak yöneticisi tanıtıcısı. Bu, kaynak yöneticisinin katıldığı genel işlemleri kaydeden sonraki <i>Transaction Prepared</i> günlük kayıtlarında da günlüğe kaydedilir.
<i>SwitchFile</i>	Bu kaynak yöneticisine ilişkin anahtar yükleme dosyası.
<i>XAOpenString</i>	Bu kaynak yöneticisine ilişkin XA açık dizgisi.
<i>XACloseString</i>	Bu kaynak yöneticisine ilişkin XA kapatma dizgisi.

Hareket Hazırlandı

Bu günlük kaydı, kuyruk yöneticisinin XA Transaction Manager bileşeni tarafından yazılır. Bu, belirtilen genel hareketin başarıyla hazırlandığını gösterir. Katılımcı kaynak yöneticilerinin her birine kesinleştirme talimatı verilecektir. Hazırlanan her kaynak yöneticisinin *RMID* kaydı, günlük kaydına kaydedilir. If the queue manager itself is participating in the transaction a *Participant Entry* with an *RMID* of zero will be present.

İşlem Unut

Bu günlük kaydı, kuyruk yöneticisinin XA Transaction Manager bileşeni tarafından yazılır. Kesinleştirme kararı her katılımcıya teslim edildiğinde *Transaction Prepared* günlük kaydını izler.

Kuyruğu Temizle

Bu, bir kuyruktaki tüm iletilerin temizlenmesini (örneğin, MQSC komutu CLEAR QUEUE) kullanmanın bir neden olduğunu kaydeder.

Kuyruk Öznitelikleri

Bu, bir kuyruğun özniteliklerinin kullanıma hazırlamayı ya da değiştirilmesini günlüğe kaydeder.

Nesne Oluştur

Bu, bir WebSphere MQ nesnesi yaratılmasını günlüğe kaydeder.

<i>ObjName</i>	Yaratılan nesnenin adı.
<i>UserId</i>	Yaratma işlemini gerçekleştiren kullanıcı kimliği.

Nesneyi Sil

Bu, bir WebSphere MQ nesnesinin silinmesini günlüğe kaydeder.

<i>ObjName</i>	Silinen nesnenin adı.
----------------	-----------------------

IBM WebSphere MQ kuyruk yöneticisi verilerinin yedeklenmesi ve geri yüklenmesi

Kuyruk yöneticileri ve kuyruk yöneticisi verileri yedekleniyor.

Düzenli aralıklarla, kuyruk yöneticilerini donanım arızalarının neden olduğu olası bozulmaya karşı korumak için önlemler alabilirsiniz. Kuyruk yöneticisini korumanın üç yolu vardır:

Kuyruk yöneticisi verilerini yedekle

Donanım başarısız olursa, kuyruk yöneticisi durdurulmaya zorlanabilir. Donanım hatası nedeniyle kuyruk yöneticisi günlüğü verileri kaybolursa, kuyruk yöneticisi yeniden başlatılamayabilir. Kuyruk yöneticisi verilerini yedeklediğinizde, kaybedilen kuyruk yöneticisi verilerinin bazılarını ya da tümünü kurtarmanız mümkün olabilir.

Genellikle kuyruk yöneticisi verilerini yedeklediğinizde, donanım hatası durumunda kaybedildiğiniz veriler, kurtarma günlüğünün bütünlüğünün kaybolmasına neden olur.

Kuyruk yöneticisi verilerini yedeklemek için kuyruk yöneticisinin çalışmaması gerekir.

Kuyruk yöneticisi verilerini yedeklemek ve geri yüklemek için aşağıdaki bilgileri kullanın:

- [“Kuyruk yöneticisi verileri yedekleniyor” sayfa 406.](#)
- [“Kuyruk yöneticisi verileri geri yükleniyor” sayfa 406.](#)

Yedek kuyruk yöneticisi kullan

Donanım hatası ciddi bir hata ise, bir kuyruk yöneticisi kurtarılamaz. Bu durumda, kurtarılamayan kuyruk yöneticisinin özel olarak ayrılmış bir yedek kuyruk yöneticisi varsa, yedek kuyruk yöneticisi kurtarılamaz kuyruk yöneticisi yerine etkinleştirilebilir. Düzenli olarak güncellendiyse, yedek kuyruk yöneticisi günlüğü, kurtarılamayan kuyruk yöneticisinden tamamlanan son günlük verilerini içeren günlük verilerini içerebilir.

Varolan kuyruk yöneticisi çalışmaya devam ederken, yedek kuyruk yöneticisi güncellenebilir.

Yedek kuyruk yöneticisini yaratmak ve etkinleştirmek için aşağıdaki başlıkları kullanın:

- [“Yedek kuyruk yöneticisi yaratılması” sayfa 408.](#)

- [“Yedek Kuyruk Yöneticisinin Başlatılması” sayfa 409.](#)

Yalnızca kuyruk yöneticisi yapılanışını yedekle

Donanım başarısız olursa, kuyruk yöneticisi durdurulmaya zorlanabilir. Donanım hatası nedeniyle hem kuyruk yöneticisi yapılanışı hem de günlük verileri kaybolursa, kuyruk yöneticisi yeniden başlatamayacak ya da günlüğünden kurtarılamaz. Kuyruk yöneticisi yapılanışını yedeklemeniz durumunda, kuyruk yöneticisini ve tüm nesnelere saklanmış tanımlamalardan yeniden yaratabilirsiniz.

Kuyruk yöneticisi yapılanışını yedeklemek için kuyruk yöneticisinin çalışır durumda olması gerekir.

Kuyruk yöneticisi yapılanışını yedeklemek ve geri yüklemek için aşağıdaki başlara bakın:

- [“Kuyruk yöneticisi yapılanışı yedekleniyor” sayfa 409](#)
- [“Kuyruk yöneticisi yapılanışı geri yükleniyor” sayfa 410](#)

Kuyruk yöneticisi verileri yedekleniyor

Kuyruk yöneticisi verilerinin yedeklenmesi, donanım hatalarından kaynaklanan olası veri kaybına karşı koruma göstermenize yardımcı olabilir.

Başlamadan önce

Kuyruk yöneticisinin çalışmadığından emin olun. Çalışmakta olan bir kuyruk yöneticisinin yedeğini almaya çalışırsanız, dosyalar kopyalanırken devam etmekte olan güncellemeler nedeniyle yedekleme tutarlı olmayabilir. Olanaklıysa, `endmqm -w` komutunu çalıştırarak kuyruk yöneticisini durdurun (bekleme kapanması), ancak bu hata başarısız olursa, `endmqm -i` komutunu kullanın (hemen sona erdirin).

Bu görev hakkında

Kuyruk yöneticisi verilerinin yedek kopyasını almak için aşağıdaki görevleri tamamlayın:

1. Yapılanış kütüklerindeki bilgileri kullanarak, kuyruk yöneticisinin verilerini ve günlük kütüklerini yerleştirdiği dizinleri arayın. Daha fazla bilgi için [“IBM WebSphere MQ ve kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi” sayfa 410](#) başlıklı konuya bakın.

Not: Dizinde görünen adları anlamakta güçlük çekebilirsiniz. The names are transformed to ensure that they are compatible with the platform on which you are using WebSphere MQ. Ad dönüştürmeleriyle ilgili daha fazla bilgi için bkz. [Understanding WebSphere MQ file names](#).

2. Tüm alt dizinler dahil olmak üzere, kuyruk yöneticisinin tüm veri ve günlük dosyası dizinlerinin kopyalarını alın.

Make sure that you do not miss any files, especially the log control file, as described in [“Günlüklerin görünüşü” sayfa 391](#), and the configuration files as described in [“Başlatma ve yapılandırma dosyaları” sayfa 69](#). Dizinlerden bazıları boş olabilir, ancak yedeklemeyi daha sonraki bir tarihte geri yüklemek için hepsine ihtiyacınız vardır.

3. Dosyaların sahiplik gemilerini koru. WebSphere MQ for UNIX and Linux sistemleri için bunu `tar` komutu ile yapabilirsiniz. (2 GB ' den büyük kuyrukunuz varsa, `tar` komutunu kullanamazsınız. Ek bilgi için [Büyük kuyrukların etkinleştirilmesi](#) başlıklı konuya bakın.

Not: WebSphere MQ Sürüm 7.5 ' e ve sonrasına yükselttiğinizde, **.ini** dosyasının ve kayıt girişlerinin yedeğini aldığınızdan emin olun. Kuyruk yöneticisi bilgileri **.ini** dosyasında depolanır ve önceki bir WebSphere MQ sürümüne dönmek için kullanılabilir.

Kuyruk yöneticisi verileri geri yükleniyor

Kuyruk yöneticisi verilerinin yedeğini geri yüklemek için aşağıdaki adımları izleyin.

Başlamadan önce

Kuyruk yöneticisinin çalışmadığından emin olun.

Bu görev hakkında

Kuyruk yöneticisi verilerinin yedeğini geri yüklemek için:

1. Yapılanış kütüklerindeki bilgileri kullanarak, kuyruk yöneticisinin verilerini ve günlük kütüklerini yerleştirdiği dizinleri bulun.
2. Yedeklenen verileri yerleştirecek dizinlerin içine boşalt.
3. Yedeklenen kuyruk yöneticisi verilerini ve günlük kütüklerini doğru yerlere kopyalayın.
4. Yapılandırma bilgileri dosyalarını güncelleyin.

Tüm gerekli dizinlere sahip olduğundan emin olmak için sonuç dizini yapısını denetleyin.

IBM WebSphere MQ dizinleri ve alt dizinleriyle ilgili daha fazla bilgi için bkz. [Windows sistemlerinde dizin yapısı ve UNIX and Linux sistemlerindeki dizin içeriği](#) .

Günlük dosyalarının yanı sıra bir günlük denetim dosyanız olduğundan emin olun. Also check that the IBM WebSphere MQ and queue manager configuration files are consistent so that WebSphere MQ can look for the restored data in the correct places.

Döngüsel günlük kaydı için, kuyruk yöneticisi verilerini ve günlük dosyası dizinlerini aynı anda yedeklemenizi ve böylece, tutarlı bir kuyruk yöneticisi verilerini ve günlüklerini geri yükleyebilmenizi sağlar.

Doğrusal günlük kaydı için, kuyruk yöneticisi verilerini ve günlük dosyası dizinlerini aynı anda yedeklemektedir. Yalnızca, eksiksiz bir günlük dosyası dizisi varsa, kuyruk yöneticisi veri dosyalarını geri yüklemek mümkündür.

Not: WebSphere MQ Sürüm 7.5 ' e ve sonrasına yükselttiğinizde, **.ini** dosyasının ve kayıt girişlerinin yedeğini aldığınızdan emin olun. Kuyruk yöneticisi bilgileri **.ini** dosyasında depolanır ve önceki bir WebSphere MQsürümüne dönmek için kullanılabilir.

Sonuçlar

Veriler doğru şekilde yedeklenip geri yüklendiyse, kuyruk yöneticisi şimdi başlatılacaktır.

Yedek kuyruk yöneticisinin kullanılması

Var olan bir kuyruk yöneticisinin özel olarak ayrılmış bir yedek kuyruk yöneticisi olabilir.

Yedek kuyruk yöneticisi, varolan kuyruk yöneticisinin etkin olmayan bir kopyasıdır. Var olan kuyruk yöneticisi, önemli donanım hatası nedeniyle kurtarılamaz duruma gelirse, kurtarılamayan kuyruk yöneticisinin yerine yedek kuyruk yöneticisi çevrimiçi duruma getirilebilir.

Yedek kuyruk yöneticisinin olağanüstü durumdan kurtarma için etkin bir yöntem olmaya devam etmesi için, varolan kuyruk yöneticisi günlük dosyalarının düzenli olarak yedek kuyruk yöneticisine kopyalanması gerekir. Var olan kuyruk yöneticisinin kopyalanacak günlük dosyaları için durdurulması gerekmez; ancak, kuyruk yöneticisi yazma işlemini bitirdiyse yalnızca bir günlük dosyasını kopyalamalısınız. Var olan kuyruk yöneticisi günlüğü sürekli olarak güncellendiğinden, var olan kuyruk yöneticisi günlüğü ile yedek kuyruk yöneticisi günlüğüne kopyalanan günlük verileri arasında her zaman küçük bir tutarsızlık var. Yedek kuyruk yöneticisine yapılan düzenli güncellemeler, iki günlükle arasındaki uyumsuzluğu en aza indirir.

Çevrimiçi olarak bir yedek kuyruk yöneticisinin getirilmesi gerekiyorsa, bu yönetici etkinleştirilmeli ve başlatılmalıdır. Bir yedekleme kuyruk yöneticisini başlatılmadan önce etkinleştirme gereksinmesi, bir yedek kuyruk yöneticisine karşı yanlışlıkla başlatılmakta olan bir koruma ölçüsüne sahip olur. Bir yedekleme kuyruk yöneticisi etkinleştirildikten sonra artık güncellenemez.

Yedekleme kuyruğu yöneticisi yaratılmasına, güncellenmesine ve başlatılmasına ilişkin bilgi edinmek için aşağıdaki konulara bakın:

- [“Yedek kuyruk yöneticisi yaratılması” sayfa 408](#)
- [“Yedek Kuyruk Yöneticinin Güncellenmesi” sayfa 408](#)
- [“Yedek Kuyruk Yöneticisinin Başlatılması” sayfa 409](#)

Yedek kuyruk yöneticisi yaratılması

Bir yedekleme kuyruk yöneticisini yalnızca doğrusal günlük kaydı kullanırken kullanabilirsiniz.

Varolan bir kuyruk yöneticisi için yedek kuyruk yöneticisi yaratmak için aşağıdaki işlemi gerçekleştirin:

1. Create a backup queue manager for the existing queue manager using the control command `strmqm`. Yedek kuyruk yöneticisi aşağıdakileri gerektirir:
 - Varolan kuyruk yöneticisiyle aynı özniteliklere sahip olmak için, örneğin, kuyruk yöneticisi adı, günlük kaydı tipi ve günlük dosyası büyüklüğü.
 - Var olan kuyruk yöneticisiyle aynı platformda yer almak için.
 - Var olan kuyruk yöneticisinden eşit ya da daha yüksek bir kod düzeyinde olmalıdır.
2. Tüm alt izinler de içinde olmak üzere, var olan kuyruk yöneticisinin tüm veri ve günlük dosyası izinlerinin kopyalarını "[Kuyruk yöneticisi verileri yedekleniyor](#)" sayfa 406 içinde açıklandığı şekilde alın.
3. Var olan kuyruk yöneticisinden alınan kopyalarla birlikte, yedek kuyruk yöneticisinin verilerinin ve tüm alt izinlerin de içinde bulunduğu günlük dosyası izinlerinin üzerine yazılır.
4. Yedek kuyruk yöneticisinde aşağıdaki denetim komutunu yürütün:

```
strmqm -r BackupQMName
```

Bu, kuyruk yöneticisini WebSphere MQ içinde bir yedek kuyruk yöneticisi olarak işaretler ve yedek kuyruk yöneticisini var olan kuyruk yöneticisiyle adımına getirmek için kopyalanan tüm günlük kapsamlarını yeniden yürütür.

Yedek Kuyruk Yöneticinin Güncellenmesi

Bir yedek kuyruk yöneticisinin olağanüstü durumdan kurtarma için etkin bir yöntem olmaya devam ettiğini doğrulamak için düzenli olarak güncellenmesi gerekir.

Düzenli güncelleme, yedek kuyruk yöneticisi günlüğü ile yürürlükteki kuyruk yöneticisi günlüğü arasındaki tutarsızlığı azaltmaya devam eder. Kuyruk yöneticisinin yedeklenmesini durdurmanın gerekmesi gerekmez.

Bir yedek kuyruk yöneticisini güncellemek için aşağıdaki işlemi gerçekleştirin:

1. Yedeklenecek kuyruk yöneticisine ilişkin aşağıdaki Script (MQSC) komutunu verin:

```
RESET QMGR TYPE(ADVANCELOG)
```

Bu, geçerli günlüğe yazma işlemi durdurur ve daha sonra kuyruk yöneticisini günlüğe kaydetme işlemi sonraki günlüğe kaydetmeye devam eder. Bu işlem, yürürlükteki saate kadar günlüğe kaydedilen tüm bilgileri yedeklemenizi sağlar.

2. Yedeklenecek kuyruk yöneticisine ilişkin şu Script (MQSC) komutunu vererek (yeni) yürürlükteki etkin günlük kaplam numarasını alın:

```
DIS QMSTATUS CURRLOG
```

3. Güncellenen günlük kaplam dosyalarını geçerli kuyruk yöneticisi günlüğü dizininden yedek kuyruk yöneticisi günlüğüne kopyalayın; son güncelleştirmeden bu yana tüm günlük kapsamlarını kopyalayın ve 2. adımda belirtilen şekilde (ancak dahil değil) tüm günlük kapsamlarını kopyalayın. Yalnızca günlük kaplam dosyalarını kopyala, "S. ." ile başlayan dosyaları.
4. Yedek kuyruk yöneticisinde aşağıdaki denetim komutunu verin:

```
strmqm -r BackupQMName
```


Bu işlem, kopyalanan tüm günlük kapsamlarını yeniden yürütür ve yedek kuyruk yöneticisini kuyruk yöneticisiyle birlikte adım adım olarak getirir. Yeniden yürütme işlemi tamamlandığında, kurtarma işlemini yeniden başlatmak için gereken tüm günlük kapsamlarını ve ortam kurtarma için gereken tüm günlük kapsamlarını tanımlayan bir ileti alırsınız.

Uyarı: Bir non-contiguous günlük kümesini yedek kuyruk yöneticisi günlüğü dizinine kopyaladıysanız, yalnızca ilk eksik günlüğün bulunduğu noktaya kadar olan günlükler yeniden oynatılır.

Yedek Kuyruk Yöneticisinin Başlatılması

Kurtarılamayan bir kuyruk yöneticisi için yedek kuyruk yöneticisinin yerine birini kullanabilirsiniz.

Bunu yapmak için aşağıdaki adımları gerçekleştirin:

1. Yedek kuyruk yöneticisini etkinleştirmek için aşağıdaki denetim komutunu yürütün:

```
stxmqm -a BackupQMName
```

Yedek kuyruk yöneticisi etkinleştirildi. Şimdi etkin olan yedek kuyruk yöneticisi artık güncellenemez.

2. Yedek kuyruk yöneticisini başlatmak için aşağıdaki denetim komutunu yürütün:

```
stxmqm BackupQMName
```

WebSphere MQ, bunu kurtarma işlemini yeniden başlatma olarak kabul eder ve yedek kuyruk yöneticisinden günlük olarak kullanılır. Yedek kuyruk yöneticisini yeniden yürütme işlemi sırasında son güncelleme gerçekleştirildiği için, yalnızca son kaydedilen denetim noktasından gelen etkin hareketler geriye işlenir.

Kurtarılamayan bir kuyruk yöneticisinin yerine, kurtarılamaz bir kuyruk yöneticisi yerine koyma kuyruk yöneticisi yerine konduğunda, kurtarılamaz kuyruk yöneticisinden bazı kuyruk yöneticisi verileri kaybedilebilir. Kaybedilen veri miktarı, yedekleme kuyruğu yöneticisinin en son ne zaman güncellendiği üzerine bağlıdır. En son güncelleme ne kadar yakın zamanda, kuyruk yöneticisi veri kaybı daha az olur.

3. Tüm kanalları yeniden başlatın.

Tüm gerekli dizinlere sahip olduğundan emin olmak için sonuç dizini yapısını denetleyin.

WebSphere MQ dizinlerine ve alt dizinlerine ilişkin ek bilgi edinmek için [Planning file system support](#) başlıklı konuya bakın.

Günlük dosyalarının yanı sıra bir günlük denetim dosyanız olduğundan emin olun. Also check that the WebSphere MQ and queue manager configuration files are consistent so that WebSphere MQ can look in the correct places for the restored data.

Veriler doğru şekilde yedeklenip geri yüklendiyse, kuyruk yöneticisi şimdi başlatılacaktır.

Not: Kuyruk yöneticisi verileri ve günlük dosyaları farklı dizinlerde tutulsa da, dizinleri aynı anda yedeklemeli ve geri yükleyebilirsiniz. Kuyruk yöneticisi verileri ve günlük dosyaları farklı yaşlara sahip olursa, kuyruk yöneticisi geçerli bir durumda değildir ve büyük olasılıkla başlatılamayacaktır. Başlatılırsa, verilerinizin bozuk olması olasılığı yüksektir.

Kuyruk yöneticisi yapılandırma yedekleniyor

Kuyruk yöneticisi yapılandırmasının yedeklenmesi, kuyruk yöneticisini tanımlamalarından yeniden oluşturmanıza yardımcı olabilir.

Kuyruk yöneticisi yapılandırmasının yedek kopyasını almak için:

1. Kuyruk yöneticisinin çalıştığını doğrulayın.
2. a. AIX, HP-UX, Linux, Solaris ya da Windows üzerinde: Dump MQ Configuration komutunu (dmpmqcfg) varsayılan biçimlendirme seçeneğini (-f mqsc) kullanarak yürütün (-f mqsc) MQSC ve tüm

öznitelikler (-a), tanımlamayı bir dosyaya saklamak için standart çıkış yeniden yönlendirmesini kullanın, örneğin:

```
dmpmqcfig -m MYQMGR -a > /mq/backups/MYQMGR.mqsc
```

Kuyruk yöneticisi yapılandırması geri yükleniyor

Kuyruk yöneticisi yapılandırmasının yedeğini geri yüklemek için aşağıdaki adımları izleyin.

Kuyruk yöneticisi yapılandırmasının yedeğini geri yüklemek için:

1. Kuyruk yöneticisinin çalıştığını doğrulayın. Veri ve günlüklerin zarar görmesiyle kurtarılamaz olduğunda kuyruk yöneticisinin yeniden yaratıldığını unutmayın.
2. Altyapınıza bağlı olarak, aşağıdaki komutlardan birini yürütün:
 - a. On AIX, HP-UX, Linux, Solaris, or Pencereler: Execute runmqsc against the queue manager, use standard input redirection to restore the definitions from a script file generated by the Dump MQ Configuration (dmpmqcfig) command, for example:

```
runmqsc MYQMGR < /mq/backups/MYQMGR.mqsc
```

İlgili başvurular

[dmpmqcfig](#)

IBM WebSphere MQ ve kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi

Change the behavior of IBM WebSphere MQ or an individual queue manager to suit the needs of your installation.

You can change IBM WebSphere MQ configuration information by changing the values specified on a set of configuration attributes (or parameters) that govern IBM WebSphere MQ.

Öznitelik bilgilerini, IBM WebSphere MQ yapılandırma dosyalarını düzenleyerek değiştirin. Windows ve Linux için IBM WebSphere MQ üzerinde (x86 ve x86-64 platformları), IBM WebSphere MQ yapılandırma dosyaları IBM WebSphere MQ Explorer kullanılarak düzenlenebilir.

On Windows systems you can also use amqmdain to change configuration information, as described in [amqmdain](#)

Platformunuz için IBM WebSphere MQ ve kuyruk yöneticilerini yapılandırma hakkında daha fazla bilgi edinmek için aşağıdaki alt konulara bakın:

İlgili kavramlar

[“Yapılandırılıyor” sayfa 5](#)

Bir ya da daha çok bilgisayarda bir ya da daha çok kuyruk yöneticisi yaratın ve bunları geliştirme, test etme ve üretim sistemlerinizde, iş verilerinizi içeren iletileri işlemek için yapılandırın.

İlgili görevler

[Planlama](#)

[WebSphere MQ' un denetlenmesi](#)

UNIX, Linux, and Windows sistemlerindeki yapılandırma bilgilerinin değiştirilmesi

Yapılandırma öznitelikleri yapılandırma kütüklerinde, düğüm düzeyinde ve kuyruk yöneticisinin altında tutulur.

Windows, UNIX and Linux platformlarında, aşağıdakiler içinde IBM WebSphere MQ yapılandırma özniteliklerini değiştirebilirsiniz:

- Düğümdeki IBM WebSphere MQ değişikliklerini bir bütün olarak yürürlüğe almak için bir IBM WebSphere MQ yapılandırma dosyası (**mqqs.ini**). Her düğüm için bir mqqs.ini dosyası vardır.

- Belirli kuyruk yöneticilerine ilişkin değişiklikleri yürürlüğe almak için bir kuyruk yöneticisi yapılanış kütüğü (**qm.ini**). Düğümdeki her kuyruk yöneticisi için bir qm.ini dosyası vardır.

İstemci yapılandırma seçenekleri, istemci yapılandırma dosyasında ayrı ayrı tutulsun.

Bir yapılandırma dosyası (ya da **stanza** dosyası) bir ya da daha çok stanzas içerir. Bu dosya, bir birlikte ortak bir işleve sahip olan .ini dosyasındaki satır gruplarıdır ya da günlük işlevleri, kanal işlevleri ve kurulabilir hizmetler gibi bir sistemin bir parçasını tanımlar.

Kuyruk yöneticileriyle ilişkili verileri bulmak için IBM WebSphere MQ yapılandırma dosyası kullanıldığından, var olmayan ya da yanlış bir yapılandırma dosyası bazı ya da tüm MQSC komutlarının başarısız olmasına neden olabilir. Ayrıca, uygulamalar, IBM WebSphere MQ yapılandırma dosyasında tanımlı olmayan bir kuyruk yöneticisine bağlanamaz.

Genellikle, bir konfigürasyon dosyasında yaptığınız değişiklikler, kuyruk yöneticisi bir sonraki başlatılıncaya kadar yürürlüğe girmez.

Windows ve Linux (x86 ve x86-64 platformları) sistemlerinde, yapılandırma bilgilerini IBM WebSphere MQ Explorerolanağından düzenleyebilirsiniz.

Windows sistemlerinde, yapılandırma dosyalarını düzenlemek için amqmda.in komutunu da kullanabilirsiniz.

Windows, UNIX and Linux sistemlerine ilişkin yapılandırma seçenekleri hakkında daha fazla bilgi için aşağıdaki alt konulara bakın:

İlgili kavramlar

[“Yapılandırılıyor” sayfa 5](#)

Bir ya da daha çok bilgisayarda bir ya da daha çok kuyruk yöneticisi yaratın ve bunları geliştirme, test etme ve üretim sistemlerinizde, iş verilerinizi içeren iletileri işlemek için yapılandırın.

[“IBM WebSphere MQ ve kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi” sayfa 410](#)

Change the behavior of IBM WebSphere MQ or an individual queue manager to suit the needs of your installation.

İlgili görevler

[Planlama](#)

[WebSphere MQ' un denetlenmesi](#)

İlgili başvurular

[“IBM WebSphere MQ yapılandırma bilgilerini değiştirmeye ilişkin öznitelikler” sayfa 416](#)

On IBM WebSphere MQ for Windows systems and on IBM WebSphere MQ for Linux (x86 and x86-64 platforms) systems, modify configuration information using the IBM WebSphere MQ Explorer. On other systems, modify the information by editing the mqsc.ini configuration file.

[“Kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi” sayfa 422](#)

Burada açıklanan öznitelikler, tek bir kuyruk yöneticisinin yapılanışını değiştirir. Bunlar, WebSphere MQ için herhangi bir ayarı geçersiz kılar.

Yapılandırma dosyalarının düzenlenmesi

Komutları ya da standart bir metin düzenleyiciyi kullanarak yapılanış dosyalarını düzenleyin.

Bir yapılandırma dosyasını düzenlemeden önce, bir kopyaya sahip olmak için gereksinim duyarsa geri dönebileceğiniz bir dosyayı yedeklemeniz gerekir.

Yapılandırma dosyalarını da düzenleyebilirsiniz:

- Düğümdeki kuyruk yöneticilerinin yapılandırmasını değiştiren komutları otomatik olarak kullanma
- Standart bir metin düzenleyiciyi kullanarak el ile

Kuruluştan sonra WebSphere MQ yapılanış kütüklerinde varsayılan değerleri düzenleyebilirsiniz.

Bir yapılandırma dosyası özniteliğe yanlış bir değer ayarladıysanız, değer yoksayılr ve sorunu belirtmek için bir işletmen iletileri yayınlanır. (Etki, özniteliği tamamen eksik olarak görmektedir.)

Yeni bir kuyruk yöneticisi yarattığınızda:

- WebSphere MQ yapılandırma dosyasını yedekle
- Yeni kuyruk yöneticisi yapılandırma dosyasını yedekle

Açıklamalar, açıklama metninden önce bir ";" ya da "#" karakteri ekleyerek yapılandırma dosyalarına dahil edilebilir. Bir açıklamayı temsil etmeden ";" ya da "#" karakterini kullanmak istiyorsanız, karakterin başına bir "\" karakteri ile önek ekleyebilir ve bu karakter yapılandırma verilerinin bir parçası olarak kullanılır.

Bir yapılandırma dosyasını ne zaman düzenlemeniz gerekiyor?

Yedeklemeden kurtarmak için bir yapılandırma dosyasını düzenleyin, bir kuyruk yöneticisini taşıyın, varsayılan kuyruk yöneticisini değiştirin ya da IBM desteğine yardımcı olun.

Örneğin, bir yapılandırma dosyasını düzenlemeniz gerekebilir:

- Bir yapılandırma dosyasını kaybederseniz. (Yedeklenebilirdeniz, yedekten kurtarın.)
- Bir ya da daha çok kuyruk yöneticisini yeni bir dizine taşımanız gerekir.
- Varsayılan kuyruk yöneticisini değiştirmeniz gerekir; var olan kuyruk yöneticisini yanlışlıkla silerseniz bu durum oluşabilir.
- Bunu IBM Support Center (Destek Merkezi) aracılığıyla yapmanız önerilir.

Yapılandırma dosyası öncelikleri

Bir önceliğin değeri birden çok yerde tanımlanır. Komutlarda ayarlanan öncelikler, yapılandırma dosyalarındaki önceliklere göre önceliklidir.

Bir yapılandırma dosyasının öncelik değerleri, aşağıdaki önceliklere göre ayarlanır:

- Komut satırında girilen parametreler, yapılandırma dosyalarında tanımlanan değerlere göre öncelikli olarak uygulanır
- qm.ini dosyalarında tanımlanan değerler, mqs.ini dosyasında tanımlanan değerlere göre öncelik kazanır.

IBM WebSphere MQ yapılandırma dosyası, mqs.ini

IBM WebSphere MQ yapılandırma dosyası (mqs.ini), düğümdeki tüm kuyruk yöneticilerine ilişkin bilgileri içerir. Kuruluş sırasında otomatik olarak yaratılır.

IBM WebSphere MQ için, UNIX and Linux ürünleri için, veri dizini ve günlük dizini her zaman sırasıyla /var/mqm ve /var/mqm/Log olur.

On Windows systems, the location of the data directory mqs . ini, and the location of the log directory, are stored in the registry, as their location can vary.

In addition, on Windows systems, the installation configuration information (contained in mqinst . ini on IBM WebSphere MQ for UNIX and Linux systems) is in the registry, as there is no mqinst . ini file on Windows.

Windows sistemlerine ilişkin mqs.ini dosyası, HKLM\SOFTWARE\IBM\WepSphere MQ anahtarında belirtilen WorkPath tarafından verilir. Şunları içerir:

- Kuyruk yöneticilerinin adları
- Varsayılan kuyruk yöneticisinin adı
- Her biri ile ilişkilendirilmiş dosyaların konumu

Yeni bir IBM WebSphere MQ kuruluşu için sağlanan LogDefault stanzas, öncelikler için belirttik değerler içermiyor. Bir önceliğin olmaması, bu değer için varsayılan değerlerin yeni bir kuyruk yöneticisi yaratılmasıyla birlikte kullanıldığı anlamına gelir. Varsayılan değerler, Şekil 71 sayfa 413'teki LogDefault stanzas için gösterilir. LogBufferPages önceliği için sıfır değeri 512 anlamına gelir.

Varsayılan olmayan bir değer gerektiriyorsa, bu değeri LogDefault stanzas içinde açıkça belirtmeniz gerekir.

```

#*****#
#* Module Name: mq5.ini                                     **#
#* Type       : WebSphere MQ Machine-wide Configuration File **#
#* Function   : Define WebSphere MQ resources for an entire machine **#
#*****#
#* Notes     :                                             **#
#* 1) This is the installation time default configuration **#
#*          **#
#*****#
AllQueueManagers:
#*****#
#* The path to the qmgrs directory, below which queue manager data **#
#* is stored **#
#*****#
DefaultPrefix=/var/mqm

LogDefaults:
  LogPrimaryFiles=3
  LogSecondaryFiles=2
  LogFilePages=4096
  LogType=CIRCULAR
  LogBufferPages=0
  LogDefaultPath=/var/mqm/log

QueueManager:
  Name=saturn.queue.manager
  Prefix=/var/mqm
  Directory=saturn!queue!manager
  InstallationName=Installation1

QueueManager:
  Name=pluto.queue.manager
  Prefix=/var/mqm
  Directory=pluto!queue!manager
  InstallationName=Installation2

DefaultQueueManager:
  Name=saturn.queue.manager

ApiExitTemplate:
  Name=OurPayrollQueueAuditor
  Sequence=2
  Function=EntryPoint
  Module=/usr/ABC/auditor
  Data=123

ApiExitCommon:
  Name=MQPoliceman
  Sequence=1
  Function=EntryPoint
  Module=/usr/MQPolice/tmqp
  Data=CheckEverything

```

Şekil 71. UNIX sistemleri için bir IBM WebSphere MQ yapılandırma dosyası örneği

Kuyruk yöneticisi yapılanış kütükleri, qm.ini

Bir kuyruk yöneticisi yapılanış kütüğü (qm.ini), belirli bir kuyruk yöneticisiyle ilgili bilgileri içerir.

Her kuyruk yöneticisi için bir kuyruk yöneticisi yapılanış kütüğü vardır. The qm.ini file is automatically created when the queue manager with which it is associated is created.

V 7.5.0.9 From IBM WebSphere MQ Version 7.5.0, Düzeltme Paketi 9, the **strmqm** command checks the syntax of the CHANNELS and SSL stanzas in the qm . ini file before starting the queue manager fully, which makes it much easier to see what is wrong, and correct it quickly if **strmqm** finds that the qm . ini file contains any errors. Ek bilgi için bkz. [strmqm](#).

qm . ini dosyalarının konumu

UNIX Linux Windows

UNIX and Linux sistemlerinde, kuyruk yöneticisi tarafından meşgul edilen dizin ağacının kökünde bir qm.ini dosyası tutulur. Örneğin, QMNAME adı verilen bir kuyruk yöneticisine ilişkin bir yapılanış kütüğünün yolu ve adı:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Windows sistemlerinde, qm.ini dosyasının konumu, HKLM\SOFTWARE\IBM\WebSphere MQ anahtarında belirtilen WorkPath tarafından verilir. Örneğin, QMNAME adı verilen bir kuyruk yöneticisine ilişkin bir yapılanış kütüğünün yolu ve adı:

```
C:\Program Files\IBM\WebSphere MQ\qmgrs\QMNAME\qm.ini
```

Kuyruk yöneticisi adı en çok 48 karakter uzunluğunda olabilir. Ancak bu, adın geçerli ya da benzersiz olduğunu garanti etmez. Bu nedenle, kuyruk yöneticisi adına dayalı olarak bir dizin adı oluşturulur. Bu süreç *ad dönüşümü* olarak bilinir. Bir açıklama için bkz. [Understanding WebSphere MQ file names](#).

Örnek qm.ini dosyası



The following example shows how groups of attributes might be arranged in a queue manager configuration file in IBM WebSphere MQ for UNIX and Linux systems.

```
##* Module Name: qm.ini                                     ##*
##* Type       : WebSphere MQ queue manager configuration file ##*
## Function    : Define the configuration of a single queue manager ##*
##*           ##*
##*****##*
##* Notes      :                                           ##*
##* 1) This file defines the configuration of the queue manager ##*
##*           ##*
##*****##*

ExitPath:
  ExitsDefaultPath=/var/mqm/exits
  ExitsDefaultPath64=/var/mqm/exits64

Service:
  Name=AuthorizationService
  EntryPoints=13

ServiceComponent:
  Service=AuthorizationService
  Name=MQSeries.UNIX.auth.service
  Module=opt/mqm/bin/amqzfu
  ComponentDataSize=0

Log:
  LogPrimaryFiles=3
  LogSecondaryFiles=2
  LogFilePages=4096
  LogType=CIRCULAR
  LogBufferPages=01
  LogPath=/var/mqm/log/saturn!queue!manager/

AccessMode:
  SecurityGroup=wmq\wmq

XAResourceManager:
  Name=DB2 Resource Manager Bank
  SwitchFile=/usr/bin/db2swit
  XAOpenString=MQBankDB
  XACloseString=
  ThreadOfControl=THREAD

Channels: 2
  MaxChannels=200
  MaxActiveChannels=100
  MQIBindType=STANDARD

AccessMode:
```

```
SecurityGroup=wmq\wmq
TCP:
  KeepAlive = Yes
  SvrSndBuffSize=32768
  SvrRcvBuffSize=32768
  Connect_Timeout=0

QMErrorLog:
  ErrorLogSize=262144
  ExcludeMessage=7234
  SuppressMessage=9001,9002,9202
  SuppressInterval=30

ApiExitLocal:
  Name=ClientApplicationAPIchecker
  Sequence=3
  Function=EntryPoint
  Module=/usr/Dev/ClientAppChecker
  Data=9.20.176.20
```

Notlar:

1. LogBufferPages için sıfır değeri, 512 değerini verir.
2. Kanal stanza hakkında daha fazla bilgi için bkz. [“Başlatma ve yapılandırma dosyaları” sayfa 69.](#)
3. XAResourceManager kısmı sayısı üst sınırı 255 ile sınırlanır. Ancak, hareket performansının bozulmasını önlemek için yalnızca az sayıda stanzalar kullanmanız gerekir.

Unix 'teWebSphere MQ , .ini dosyası uzantısı olan yapılandırma dosyalarını kullanır; örneğin, qm.ini. There are some utilities in WebSphereMQ, such as **setmqm**, that will make a temporary backup copy of the files. For example, the file qm.ini will create a backup copy named qm.ini.bak. Bir yardımcı program qm.ini dosyasını değiştirir, güncellenen dosyayı saklar ve qm.ini.bak dosyasını siler. If the utility cannot store the qm.ini file, then it restores the contents of qm.ini from the backup file qm.ini.bak, and then deletes the qm.ini.bak file.

Var olan bir qm.ini.bak dosyası varsa, yardımcı program qm.ini dosyasını qm.ini.bak dosyasının içeriğiyle geri çevirir ve qm.ini.bak dosyasını siler. Bu nedenle, bu yedek dosyaları WebSphere MQ yardımcı programları tarafından silinebileceğinden, .bak dosyasının uzantısını kullanarak *.ini dosyalarının yedek kopyalarını yaratmamalısınız.

Değişikliklerinizin ne zaman yürürlüğe girmesine ilişkin bilgi için bkz. [“UNIX, Linux, and Windows sistemlerindeki yapılandırma bilgilerinin değiştirilmesi” sayfa 410 .](#)

Kuruluş yapılandırma dosyası, mqinst.ini

UNIX and Linux sistemleri

The installation configuration file, mqinst.ini, contains information about all the IBM WebSphere MQ installations on a UNIX or Linux system.

mqinst.ini dosyası, UNIX and Linux sistemlerindeki /etc/opt/mqm dizininde yer alıyor. Her kurulumla ilişkin aşağıdaki bilgilerin yanı sıra, varsa, birincil kurulumla ilişkin bilgileri de içerir:

- Kuruluş adı
- Kuruluş tanımlaması
- Kuruluş tanıtıcısı
- Kuruluş yolu

Biçimi düzeltilmediği ve değiştirilebileceği için, bu dosyanın doğrudan düzenlenmemesi ya da bu dosyaya gönderme yapılmaması gerekir. Bunun yerine, mqinst.ini kütüğündeki değerleri yaratmak, silmek, sorgulamak ve değiştirmek için aşağıdaki komutları kullanın:

```
Girdiler oluşturmak için crtmqinst .
Girdileri silmek için dlmqinst .
Girdileri görüntülemek için dspmqinst .
Girdileri ayarlamak için setmqinst .
```

Yalnızca iç kullanım için kuruluş tanıtıcısı otomatik olarak ayarlıdır ve değiştirilmemelidir.

Windows sistemleri

Kuruluş yapılandırma bilgileri, Windows sistemlerinde aşağıdaki anahtarda tutulur:

```
HKLM\SOFTWARE\IBM\WebSphere MQ\Installation\
```

Bu anahtarın biçimi düzeltilmediği ve değiştirilebileceği için, bu anahtarın düzenlenmemesi ya da doğrudan gönderme yapılmaması gerekir. Bunun yerine, kayıttaki değerleri sorgulamak ve değiştirmek için aşağıdaki komutları kullanın:

Girdileri görüntülemek için `dsqmqlinst`.

Girdileri ayarlamak için `setmqinst`.

Windows üzerinde, `crtmqinst` ve `dltmqinst` komutları kullanılamaz. Kuruluş ve kaldırma işlemleri, gerekli kayıt girişlerinin yaratılması ve silinmesine ilişkin işlem sağlar.

IBM WebSphere MQ yapılandırma bilgilerini değiştirmeye ilişkin öznitelikler

On IBM WebSphere MQ for Windows systems and on IBM WebSphere MQ for Linux (x86 and x86-64 platforms) systems, modify configuration information using the IBM WebSphere MQ Explorer. On other systems, modify the information by editing the mqsc.ini configuration file.

Belirli bileşenlere ilişkin öznitelikler için aşağıdaki alt başlıklara bakın:

İlgili kavramlar

[“Yapılandırılıyor” sayfa 5](#)

Bir ya da daha çok bilgisayarda bir ya da daha çok kuyruk yöneticisi yaratın ve bunları geliştirme, test etme ve üretim sistemlerinizde, iş verilerinizi içeren iletileri işlemek için yapılandırın.

[“IBM WebSphere MQ ve kuyruk yöneticisi yapısını değiştirilmesi” sayfa 410](#)

Change the behavior of IBM WebSphere MQ or an individual queue manager to suit the needs of your installation.

İlgili görevler

[Planlama](#)

[WebSphere MQ' un denetlenmesi](#)

İlgili başvurular

[“Kuyruk yöneticisi yapısını değiştirilmesi” sayfa 422](#)

Burada açıklanan öznitelikler, tek bir kuyruk yöneticisinin yapısını değiştirir. Bunlar, WebSphere MQ için herhangi bir ayarı geçersiz kılar.

Tüm kuyruk yöneticileri

Tüm kuyruk yöneticilerine ilişkin aşağıdaki bilgileri belirtmek için, IBM WebSphere MQ Gezgini 'nden General ve Extended WebSphere MQ özellikleri sayfasını ya da mqsc.ini dosyasındaki AllQueueManagers stanza olanağını kullanın.

DefaultPrefix=directory_name

Bu öznitelik, kuyruk yöneticisi verilerinin içinde saklandığı qmgrs dizininin yolunu belirtir.

Kuyruk yöneticisi için varsayılan öneki değiştirdiğinizde, kuruluş sırasında yaratılan dizin yapısını eşleyin.

Özellikle, qmgrs yapısını oluşturmalısınız. Varsayılan öneki değiştirmeden önce WebSphere MQ 'u durdurun ve yapıları yeni konuma taşıdıktan sonra WebSphere MQ ' u yeniden başlatın ve varsayılan öneki değiştirdiniz.

Not: UNIX and Linux sistemlerinde /var/mqm/errors dizinini ya da Windows sistemlerindeki \errors dizinini silmeyin.

Varsayılan öneki değiřtirmenin alternatifi olarak, `crtmqm` komutu için `DefaultPrefix` komutunu geçersiz kılmak için `MQSPREFIX` ortam değışkenini kullanabilirsiniz.

İřletim sistemi kısıtlamaları nedeniyle, belirtilen yolu yol uzunluęunun toplamı ve kuyruk yöneticisi adı en çok 70 karakter uzunluęunda olacak řekilde, saęlanan yolu yeterince kısa tutun.

ConvEBCDICNewline= NL_TO_LF | TABLE | ISO

EBCDIC kod sayfaları ASCII kod sayfaları tarafından desteklenmeyen bir newline (NL) karakteri içerir (ASCII 'nin bazı ISO değışkenleri eřdeęer içerse de).

WebSphere MQ 'un EBCDIC NL karakterini ASCII biçimine nasıl dönüřtüreceęini belirtmek için `ConvEBCDICNewline` öznitelięini kullanın.

NL_TO_LF

EBCDIC NL karakterini (X'15 ') ASCII satır besleme karakterine (LF) (X'0A'), tüm EBCDIC 'ler için ASCII dönüřtürmeleri için dönüřtürün.

Varsayılan değeri `NL_TO_LF` 'dir.

Çizelge

EBCDIC NL karakterini, altyapınızda kullanılan dönüřtürme çizelgelerine göre, tüm EBCDIC 'den ASCII 'ye dönüřtürmeye dönüřtürür.

Bu dönüřtürme tipinin etkisi altyapıdan platforma ve dilden dile kadar değışebilir; aynı platformda bile, farklı CCSID 'ler kullanıyorsanız davranıř değışebilir.

ISO

Dönüřtür:

- TABLE yöntemini kullanan ISO CCSID 'ler
- NL_TO_CF yöntemini kullanan dięer tüm CCSID 'ler

Olası ISO CCSID 'ler [Çizelge 32 sayfa 417](#) içinde gösterilmektedir.

<i>Çizelge 32. Olası ISO CCSID 'lerin listesi</i>	
CCSID	Kod Kümesi
819	ISO8859-1
912	ISO8859-2
915	ISO8859-5
1089	ISO8859-6
813	ISO8859-7
916	ISO8859-8
920	ISO8859-9
1051	roman8

ASCII CCSID değeri ISO altkümesi değilse, `ConvEBCDICNewline` varsayılan değeri olarak `NL_TO_LF` olarak ayarlanır.

Varsayılan kuyruk yöneticisi

Varsayılan kuyruk yöneticisini belirtmek için, IBM WebSphere MQ Gezgini 'nden `General` WebSphere MQ özellikler sayfasını ya da `mqsc.ini` dosyasındaki `DefaultQueueManager` stanza olanaęını kullanın.

Ad =default_queue_manager

Varsayılan kuyruk yöneticisi, bir kuyruk yöneticisi adının açık bir řekilde belirtilmemesi için gereken komutları iřler. Yeni bir varsayılan kuyruk yöneticisi oluřturursanız, `DefaultQueueManager` öznitelięi otomatik olarak güncelleřtirilir. Yanlıřlıkla yeni bir varsayılan kuyruk yöneticisi yarattıktan sonra, özgün olarak geri dönmek istiyorsanız, `DefaultQueueManager` öznitelięini el ile değıştirin.

Çıkış özellikleri

Kuyruk yöneticisi çıkış programları tarafından kullanılan yapılandırma seçeneklerini belirlemek için, IBM WebSphere MQ Gezgin'i 'nden Extended IBM WebSphere MQ özellikler sayfasını ya da mqsc.ini dosyasındaki ExitProperties stanza olanağını kullanın.

CLWLMode=SAFE|CAST (CAST)

Küme iş yükü (CLWL) çıkışı, bir MQI çağrısına yanıt olarak hangi küme kuyruğunun açılacağı (örneğin, MQOPEN, MQPUT) hangi küme kuyruğunu açabildiğinizi belirtmenize olanak tanır. CLWL çıkışı, CLWLMode özniteisinde belirttiğiniz değere bağlı olarak, FAST ya da SAFE kipinde çalıştırılır. CLWLMode öznitelikliğini atlarsanız, küme iş yükü çıkışı SAFE kipinde çalıştırılır.

SAFE

CLWL çıkışını kuyruk yöneticisinden ayrı bir işlemde çalıştırın. Bu varsayılandır.

SAFE kipinde çalışırken kullanıcı tarafından yazılan CLWL çıkışıyla ilgili bir sorun ortaya çıksa, aşağıdakiler gerçekleşir:

- CLWL sunucu işlemi (amqzlwao) başarısız olur.
- Kuyruk yöneticisi, CLWL sunucu işlemini yeniden başlatır.
- Hata, hata günlüğüne raporlanır. Bir MQI çağrısı devam ederse, bildirim formunu bir dönüş kodu biçiminde alırsınız.

Kuyruk yöneticisinin bütünlüğü korunur.

Not: CLWL çıkışının ayrı bir süreçte çalıştırılması başarıyı etkileyebilir.

Hızlı

Kuyruk yöneticisi işleminde küme çıkışını iç olarak çalıştırın.

Bu seçeneğin belirlenmesi, SAFE kipindeki çalışmayla ilişkili süreç değiştirme maliyetlerinden kaçınarak performansı iyileştirir, ancak bunu kuyruk yöneticisi bütünlüğü pahasına yapar. You should only run the CLWL exit in FAST mode if you are convinced that there are **hayır** problems with your CLWL exit, and you are particularly concerned about performance.

CLWL çıkışı FAST kipinde çalışırken bir sorun ortaya çıkmışsa, kuyruk yöneticisi başarısız olur ve kuyruk yöneticisinin bütünlüğünün tehlikeye atılma riskini siz çalıştırıyorsunuz demektir.

IBM WebSphere MQ için günlük varsayılanları

Tüm kuyruk yöneticilerine ilişkin günlük varsayılanlarına ilişkin bilgileri belirlemek için IBM WebSphere MQ Explorer' den Default log settings IBM WebSphere MQ özellikler sayfasını ya da mqsc.ini dosyasındaki LogDefaults stanza sayfasını kullanın.

Stanza yoksa, MQ varsayılan değerleri kullanılır. Bir kuyruk yöneticisi yarattığınızda günlük öznitelikleri varsayılan değerler olarak kullanılır; ancak, crtmqm komutundaki günlük özniteliklerini belirttiğinizde geçersiz kılınabilir. Bu komutun ayrıntıları için bkz. [crtmqm](#).

Bir kuyruk yöneticisi yaratıldıktan sonra, o kuyruk yöneticisine ilişkin günlük öznitelikleri, "[Kuyruk yöneticisi günlükleri](#)" sayfa 426 içinde açıklanan ayarlardan alınır.

Belirli kuyruk yöneticisi ("[Kuyruk yöneticisi günlükleri](#)" sayfa 426 içinde belirtilir) için belirtilen varsayılan örnek ("[Tüm kuyruk yöneticileri](#)" sayfa 416 içinde belirtilir) ve günlük yolu, kuyruk yöneticisinin ve günlüğünün farklı fiziksel sürücülerde olmasını sağlar. Bu yöntem, varsayılan değer olarak aynı sürücüde yer alsa da, önerilen yöntemdir.

Günlük boyutlarının hesaplanması hakkında bilgi için bkz. "[Günlüğün büyüklüğünün hesaplanması](#)" sayfa 395.

Not: Aşağıdaki parametre listesinde verilen sınırlar, WebSphere MQ tarafından ayarlanan sınırlardır. İşletim sistemi sınırları, olası günlük büyüklüğü üst sınırını azaltabilir.

LogPrimaryFiles=3|2-254 (Windows) |2-510 (UNIX and Linux sistemleri)

Kuyruk yöneticisi yaratıldığında ayrılmış olan günlük dosyaları.

Sahip olduğunuz birincil günlük dosyası sayısı alt sınırı 2 'dir ve üst sınır değeri Windows' ta 254 ya da UNIX and Linux sistemlerindeki 510 'tır. Varsayılan 3'tür.

The total number of primary and secondary log files must not exceed 255 on Pencereler, or 511 on UNIX and Linux systems, and must not be less than 3.

Bu değer, kuyruk yöneticisi yaratıldığında ya da başlatıldığında incelenir. Bunu, kuyruk yöneticisi yaratıldıktan sonra değiştirebilirsiniz. Ancak, değer içindeki bir değişiklik kuyruk yöneticisi yeniden başlatılincaya kadar yürürlüğe girmez ve etki hemen hemen olmayabilir.

LogSecondaryFiles=2|1-253 (Windows) |1-509 (UNIX and Linux sistemleri)

Birincil dosyalar tükendiğinde ayrılmış olan günlük dosyaları.

The minimum number of secondary log files is 1 and the maximum is 253 on Pencereler, or 509 on UNIX and Linux systems. Varsayılan değer 2 'dir.

The total number of primary and secondary log files must not exceed 255 on Pencereler, or 511 on UNIX and Linux systems, and must not be less than 3.

Değer, kuyruk yöneticisi başlatıldığında incelenir. Bu değeri değiştirebilirsiniz, ancak kuyruk yöneticisi yeniden başlatılincaya kadar değişiklikler yürürlüğe girmez ve etki hemen hemen olmayabilir.

LogFilePages =sayı

Günlük verileri, günlük dosyaları adı verilen bir dizi dosya içinde tutulur. Günlük dosyası büyüklüğü 4 KB ' lik sayfa birimlerinde belirtilir.

Günlük dosyası sayfalarının varsayılan sayısı 4096, günlük dosyası büyüklüğü 16 MB 'dir.

UNIX and Linux sistemlerinde günlük dosyası sayfası sayısı alt sınırı 64 'tür ve Windows ' da günlük dosyası sayfası sayısı alt sınırı 32 'dir; her iki durumda da üst sınır 65 535 'dir.

Not: Kuyruk yöneticisi yaratma işlemi sırasında belirtilen günlük kütüklerinin büyüklüğü, kuyruk yöneticisi için değiştirilemez.

LogType=CIRCULAR| LINEAR

Kullanılacak günlüğün tipi. Varsayılan değer DAIRESEL 'dir.

CIRCULAR

Sistem durduğunda, devam etmekte olan işlemleri geri almak için günlüğü kullanarak kurtarma işlemi yeniden başlatın.

Dairesel günlüğe kaydetmenin tam açıklaması için bkz. "[Günlüğe kaydetme tipleri](#)" sayfa 392 .

doğrusal

Yeniden başlatma kurtarma ve ortam ya da ileriye işleyerek kurtarma için (günlüğün içeriğini yeniden oynayarak kaybedilen ya da zarar görmüş veriler yaratılması).

Doğrusal günlüğe kaydetmenin tam açıklaması için bkz. "[Günlüğe kaydetme tipleri](#)" sayfa 392 .

Varsayılan değeri değiştirmek isterseniz, LogType özneteliğini düzenleyebilir ya da cırtmqm komutunu kullanarak doğrusal kaydetmeyi belirtebilirsiniz. Bir kuyruk yöneticisi yaratıldıktan sonra günlüğe kaydetme yöntemini değiştiremezsiniz.

LogBufferPages=0|0-4096

4 KB ' lik sayfa birimlerindeki arabelleklerin büyüklüğünü belirleyen, yazma için arabellek kayıtlarına ayrılan bellek miktarı.

Arabellek sayfa sayısı alt sınırı 18 ve üst sınır 4096 'tır. Daha büyük arabellekler, özellikle daha büyük iletiler için daha yüksek verimi elde eder.

0 değerini belirtirseniz (varsayılan değer), kuyruk yöneticisi büyüklüğü seçer. WebSphere MQ Sürüm 7.0 ' da bu değer 512 'dir (2048 KB).

1 ile 17 aralığında bir sayı belirtirseniz, kuyruk yöneticisi varsayılan olarak 18 (72 KB) değerine ayarlanır. 18 ve 4096 aralığında bir sayı belirtirseniz, kuyruk yöneticisi ayrılan belleği ayarlamak için belirtilen sayıyı kullanır.

LogDefaultYolu =directory_name

Bir kuyruk yöneticisine ilişkin günlük kütüklerinin bulunduğu dizin. Dizin, kuyruk yöneticisinin yazabileceği ve tercihen ileti kuyruklarından farklı bir sürücüde bulunan yerel bir aygıtta bulunur. Farklı bir sürücü belirlenmesi, sistem arızası durumunda ek koruma sağlar.

Varsayılan:

- <DefaultPrefix>\log for WebSphere MQ for Windows ; burada <DefaultPrefix> , All Queue Managers WebSphere MQ özellikler sayfasındaki DefaultPrefix öznitelide belirtilen değerdir. Bu değer kuruluş sırasında ayarlanır.
- /var/mqm/log for WebSphere MQ for UNIX and Linux systems

Diğer bir seçenek olarak, -ld işaretini kullanarak crtqm komutundaki bir dizinin adını da belirtebilirsiniz. Bir kuyruk yöneticisi yaratıldığında, kuyruk yöneticisi dizini altında da bir dizin yaratılır ve bu dizin günlük kütüklerini tutmak için kullanılır. Bu dizinin adı kuyruk yöneticisi adını temel alır. Bu, günlük dosyası yolunun benzersiz olmasını ve dizin adı uzunlukları üzerinde herhangi bir sınırlamaya uymasını sağlar.

crtqm komutunda -ld seçeneğini belirlemezseniz, mq.s.ini dosyasındaki LogDefaultPath özniteliğinin değeri kullanılır.

Birden çok kuyruk yöneticisinin farklı günlük dizinleri kullandığından emin olmak için, kuyruk yöneticisi adı dizin adının sonuna eklenir.

Kuyruk yöneticisi yaratıldığında, yapılanış bilgilerindeki günlük özniteliklerinde bir LogPath değeri yaratılır ve kuyruk yöneticisinin günlüğü için tam dizin adı verilmesine neden olur. Bu değer, kuyruk yöneticisi başlatıldığında ya da silindiğinde günlüğün yerini belirlemek için kullanılır.

LogWriteIntegrity =SingleWrite|DoubleWrite| TripleWrite

Günlüğe kaydedicinin, günlük kayıtlarını güvenilir olarak yazmak için kullandığı yöntem.

TripleWrite

Bu varsayılan yöntemdir.

Not, **DoubleWrite** seçeneğini belirleyebileceğinden, ancak bunu yapabiliyorsa, sistem bunu **TripleWrite** olarak yorumluyor.

SingleWrite

You should use **SingleWrite**, only if the file-system or device hosting the WebSphere MQ recovery log explicitly guarantees the atomicity of 4KB writes.

Yani, bir 4KB sayfasının yazma işlemi herhangi bir nedenle başarısız olduğunda, olası tek iki durum, önceki görüntüden ya da görüntünün sonrasına sahip olabilir. Hiçbir ara durum mümkün değil.

ACPI (Gelişmiş Yapılandırma ve Güç Arabirimi)

Sistem bir askıya alma isteği aldığı anda, WebSphere MQ ' nın nasıl davranacağını belirtmek için IBM WebSphere MQ Explorer ' dan ACPI WebSphere MQ özellikler sayfasını kullanın.

Windows , Advanced Configuration and Power Interface (ACPI) standardını destekler. Bu, sistem girdiğinde ve askıya alma kipinden devam ederken, ACPI etkin donanıma sahip Pencereler kullanıcılarının kanalları durdurmasını ve yeniden başlatmalarını sağlar.

ACPI WebSphere MQ özellikler sayfasında belirtilen ayarların yalnızca Uyarı İzleyici çalışırken uygulandığını göz önünde bulundurun. Uyarı İzleme Programı çalışıyorsa, Uyarı İzleyici simgesi görev çubuğunda bulunur.

DoDialog=Y | N

Bir askıya alma isteği sırasında iletişim kutusunu görüntüler.

DenySuspend= Y | N

Askıya alma isteğini reddeder. This is used if DoDialog=N, or if DoDialog=Y and a dialog cannot be displayed, for example, because your notebook lid is closed.

CheckChannelsRunning=Y | N

Çalışan kanalların olup olmadığını denetler. Sonuç, diğer ayarların sonucunu belirleyebilir.

Aşağıdaki çizelgede bu parametrelerin her birleşiminin etkisi sıralanıyor:

DoDialog	DenySuspend	CheckChannels Çalışıyor	İşlem
N	N	N	Askıya alma isteğini kabul edin.
N	N	Y	Askıya alma isteğini kabul edin.
N	Y	N	Askıya alma isteğini reddedin.
N	Y	Y	Çalışmakta olan kanallar varsa, askıya alma isteğini reddedin; bu isteği kabul etmezseniz.
Y	N	N	İletişim kutusunu görüntüler (bkz. Not ; askıya alma isteğini kabul edin). Bu varsayılandır.
Y	N	Y	Çalışmakta olan hiçbir kanal, askıya alma isteğini kabul etmiyorsa; iletişim kutusunu görüntüleniyorsa (bkz. Not ; isteği kabul edin).
Y	Y	N	İletişim kutusunu görüntüle (Not ; askıya alma isteğini reddetme).
Y	Y	Y	Hiçbir kanal, askıya alma isteğini kabul etmiyorsa; iletişim kutusunu görüntülense (Not ; isteği reddetme).

Not: İletişim kutusunun görüntüleneceği durumlarda, iletişim kutusu görüntülenemezse (örneğin, not defteri kapınız kapalı olduğundan), askıya alma isteğinin kabul edilip edilmediğini belirlemek için DenySuspend seçeneği kullanılır.

API çıkışları

API çıkışlarına ilişkin girdileri değiştirmek için IBM WebSphere MQ Explorer ya da amqmdain komutunu kullanın.

Tüm kuyruk yöneticilerine ilişkin API çıkış yordamlarını tanımlamak için IBM WebSphere MQ Explorer'a da mqs.ini dosyasındaki ApiExitTemplate ve ApiExitCommon stanza özelliklerini kullanarak Exits IBM WebSphere MQ özellikler sayfasını kullanın. Windows sistemlerinde, API çıkışlarına ilişkin girdileri değiştirmek için amqmdain komutunu da kullanabilirsiniz. (To identify API exit routines for individual queue managers, you use the ApiExitLocal stanza, as described in “API çıkışları” sayfa 435.)

Bu stanzalara ilişkin özniteliklere ilişkin eksiksiz açıklamalar için [API çıkışlarının yapılandırılması](#) başlıklı konuya bakın.

Kuyruk yöneticileri

Her kuyruk yöneticisi için bir QueueManager stanza vardır. Kuyruk yöneticisi dizininin yerini belirtmek için stanza 'yı kullanın.

Windows, UNIX and Linux sistemlerinde, her kuyruk yöneticisi için bir QueueManager stanza vardır. Bu öznitelikler, kuyruk yöneticisi adını ve o kuyruk yöneticisiyle ilişkilendirilmiş dosyaları içeren dizinin adını belirtir. Dizinin adı kuyruk yöneticisi adını temel alır, ancak kuyruk yöneticisi adı geçerli bir dosya adı değilse dönüştürüledir. Ad dönüşümüyle ilgili daha fazla bilgi için bkz. [Understanding WebSphere MQ file names](#).

Ad =queue_manager_name

Kuyruk yöneticisinin adı.

Önek =önek

Kuyruk yöneticisi kütüklerinin saklandığı yer. Varsayılan olarak bu değer, Tüm Kuyruk Yöneticileri bilgilerinin DefaultPrefix özniteliğinde belirtilen değerle aynıdır.

Dizin =ad

Kuyruk yöneticisi kütüklerinin saklandığı <prefix>\QMGRS dizini altındaki alt dizinin adı. Bu ad kuyruk yöneticisi adını temel alır, ancak yinelenen bir ad varsa ya da kuyruk yöneticisi adı geçerli bir dosya adı değilse dönüştürülebilirler.

DataPath=yol

Kuyruk yöneticisi yaratıldığında sağlanan belirtik bir veri yolu, kuyruk yöneticisi verilerinin yolu olarak Prefix ve Directory değerini geçersiz kılar.

InstallationName=ad

Bu kuyruk yöneticisiyle ilişkilendirilmiş WebSphere MQ kuruluşu adı. Bu kuyruk yöneticisiyle etkileşimde bulunulduğunda, bu kuruluşun komutlarının kullanılması gerekir. If no InstallationName value is present, the queue manager is associated with an installation of WebSphere MQ earlier than version 7.1.

İlgili kavramlar

“Kuyruk yöneticisinin bir kuruluşla ilişkilendirilmesi” sayfa 16

Bir kuyruk yöneticisi yarattığınızda, bu yönetici otomatik olarak **crtmqm** komutunu veren kuruluşla ilişkilendirilir. On UNIX, Linux, and Windows, you can change the installation associated with a queue manager using the **setmqm** command.

Güvenlik

Nesne Yetkisi Yöneticisi 'ne (OAM) ilişkin seçenekleri belirlemek için qm.ini dosyasındaki Security Stanza 'yı kullanın.

ClusterQueueAccessControl=RQMName | Xmitq

Küme kuyruklarının erişim denetimini ya da küme kuyruğu yöneticilerindeki tam olarak nitelenmiş kuyrukları denetlemek için bu özniteliği ayarlayın.

RQMAdı

Uzaktan barındırılan kuyrukların erişim denetimi için denetlenen tanıtlar, adlandırılmış kuyruklar ya da adlandırılmış kuyruk yöneticisi profilleridir.

Xmitq

Uzaktan barındırılan kuyrukların erişim denetimi için denetlenen profiller, SYSTEM.CLUSTER.TRANSMIT.QUEUE' a çözülür.

Varsayılan değer, 'Xmitq' değeridir.

GroupModel=GlobalGroups

Bu öznitelik, OAM ' in Windowsüzzerindeki bir kullanıcının grup üyeliğini belirlerken genel grupları denetleyip denetleyip denetmeyeceğini belirler.

Varsayılan değer, genel grupları işaretlemek değildir.

GlobalGroups

OAM genel grupları denetler.

GlobalGroups kümesi, yetkilendirme komutları, **setmqaut**, **dspmqaute** ve **dmpmqaut** genel grup adlarını kabul eder; **setmqaut** -g parametresine bakın.

Not: Setting the ClusterQueueAccessControl=RQMName and having a custom implementation of the Authorization Service at less than MQZAS_VERSION_6 results in the queue manager not starting. Bu örnekte, ClusterQueueAccessControl=Xmitq seçeneğini ayarlayın ya da özel Yetkilendirme Hizmetini MQZAS_VERSION_6 ' e ya da daha büyük bir sürüme yükseltin.

Kuyruk yöneticisi yapılanış bilgilerinin değiştirilmesi

Burada açıklanan öznitelikler, tek bir kuyruk yöneticisinin yapılanışını değiştirir. Bunlar, WebSphere MQ için herhangi bir ayarı geçersiz kılar.

UNIX and Linux sistemlerinde, kuyruk yöneticisi yapılandırma bilgilerini qm.ini yapılandırma kütüğünü düzenleyerek değiştirmenize yardımcı olur. qm.ini içinde bir stanza tanımlıyorsanız, her öğeyi yeni bir satırda başlatmanız gerekmez. Bir açıklamayı belirtmek için noktalı virgül (;) ya da bir HASH karakteri (#) kullanabilirsiniz.

Windows ve Linux (x86 ve x86-64 platformları) sistemlerinde, bazı yapılandırma bilgilerini IBM WebSphere MQ Explorer komutunu kullanarak değiştirebilirsiniz. Ancak, kurulabilir hizmetlerin ve bunların bileşenlerini değiştirmenin önemli etkileri olduğu için, kurulabilir hizmetler IBM WebSphere MQ Explorer' de salt okunur olur. You must therefore make any changes to installable services by using **regedit** on Windows, and by editing the qm.ini file on UNIX and Linux.

Kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesine ilişkin ayrıntılar için aşağıdaki alt başlıklara bakın:

İlgili kavramlar

[“Yapılandırılıyor” sayfa 5](#)

Bir ya da daha çok bilgisayarda bir ya da daha çok kuyruk yöneticisi yaratın ve bunları geliştirme, test etme ve üretim sistemlerinizde, iş verilerinizi içeren iletileri işlemek için yapılandırın.

[“IBM WebSphere MQ ve kuyruk yöneticisi yapılandırma bilgilerinin değiştirilmesi” sayfa 410](#)

Change the behavior of IBM WebSphere MQ or an individual queue manager to suit the needs of your installation.

İlgili görevler

[Planlama](#)

[WebSphere MQ' un denetlenmesi](#)

İlgili başvurular

[“IBM WebSphere MQ yapılandırma bilgilerini değiştirmeye ilişkin öznitelikler” sayfa 416](#)

On IBM WebSphere MQ for Windows systems and on IBM WebSphere MQ for Linux (x86 and x86-64 platforms) systems, modify configuration information using the IBM WebSphere MQ Explorer. On other systems, modify the information by editing the mqsc.ini configuration file.

Erişim Kipi

Access Mode yalnızca Windows sunucuları için geçerlidir. AccessMode kısmı, **crtmqm** komutundaki -a [r] seçeneği tarafından ayarlanır. Kuyruk yöneticisi yaratıldıktan sonra AccessMode Stanza 'yı değiştirmeyin.

Erişim grubunu kullan (-a [r]) Bir Windows güvenlik grubu belirtmek için **crtmqm** komutunun seçeneği, tüm kuyruk yöneticisi veri dosyalarına tam erişim izni verilecek olan üyeler. Kullanılan sözdizimine bağlı olarak, grup yerel ya da genel bir grup olabilir. Grup adı için geçerli sözdizimi aşağıdaki gibidir:

LocalGroup

Etki alanı adı \ GlobalGroup adı

GlobalGroup adı@Etki alanı adı

crtmqm komutunu -a [r] seçeneğiyle çalıştırmadan önce, ek erişim grubunu tanımlamanız gerekir.

Grubu, -ayerine -ar kullanarak belirtirseniz, yerel mqm grubuna kuyruk yöneticisi veri dosyalarına erişim izni verilmez. Bu seçeneği, kuyruk yöneticisi veri dosyalarını bulunduran dosya sistemi yerel olarak tanımlı gruplar için erişim denetimi girişlerini desteklemiyorsa kullanın.

Grup tipik olarak, paylaşılan bir kuyruk yöneticisi verilerine ve günlük dosyasına erişimi olan çok eşgörünümli kuyruk yöneticileri sağlamak için kullanılan genel bir güvenlik grubudur. Klasör üzerinde okuma ve yazma izinlerini ayarlamak ya da kuyruk yöneticisi verilerini ve günlük dosyalarını içeren paylaşımı paylaşmak için ek güvenlik erişim grubunu kullanın.

Ek güvenlik erişim grubu, kuyruk yöneticisi verilerini ve günlüklerini içeren klasöre ilişkin izinleri ayarlamak için mqm adlı yerel grubu kullanmaya alternatif bir seçenektir. Unlike the local group mqm, you can make the additional security access group a local or a global group. Çok eşgörünümli kuyruk yöneticileri tarafından kullanılan verileri ve günlük dosyalarını içeren paylaşılan klasörler üzerinde izinleri ayarlamak için bu genel bir grup olmalıdır.

Windows işletim sistemi, kuyruk yöneticisi verilerini ve günlük dosyalarını okumak ve yazmak için erişim izinlerini denetler. Kuyruk yöneticisi işlemleri çalıştıran kullanıcı kimliğinin izinlerini denetler. Denetlenen kullanıcı kimliği, kuyruk yöneticisini bir hizmet olarak mı başlattığınız ya da etkileşimli olarak başlattığınız için değişir. If you started the queue manager as a service, the user ID checked by the Windows system is the user ID you configured with the **Hazırla** IBM WebSphere MQ wizard. If you started the queue manager interactively, the user ID checked by the Windows system is the user ID that ran the **strmqm** command.

Kuyruk yöneticisini başlatmak için, kullanıcı kimliğinin yerel mqm grubunun bir üyesi olması gerekir. Kullanıcı kimliği, ek güvenlik erişim grubunun bir üyesiye, kuyruk yöneticisi, grubu kullanarak izinleri verilen dosyaları okuyabilir ve yazabilir.

Sınırlama: Yalnızca Windows işletim sisteminde bir ek güvenlik erişim grubu belirleyebilirsiniz. Diğer işletim sistemlerinde bir ek güvenlik erişim grubu belirtirseniz, **crtmqm** komutu bir hata döndürür.

İlgili kavramlar

[“Secure unshared queue manager data and log directories and files on Windows” sayfa 368](#)

[“Paylaşılan kuyruk yöneticisi verilerini ve günlük dizinlerini ve dosyalarını Windows üzerinde güvenli” sayfa 365](#)

İlgili görevler

[“Etki alanı iş istasyonlarında ya da sunucularda çok eşgörünümü kuyruk yöneticisi yarat” sayfa 341](#)

İlgili başvurular

[crtmqm](#)

Kurulabilir hizmetler

You change installable services on Windows by using **regedit**, and on UNIX and Linux by using the Service stanza in the `qm.ini` file.

Not: Kurulabilir hizmetleri ve bunların bileşenlerini değiştirmenin önemli etkileri vardır. Bu nedenle, kurulabilir hizmetler WebSphere MQ Explorer 'da salt okunur olur.

Windows sistemlerinde kurulabilir hizmetleri değiştirmek için, `regedit`ya da UNIX and Linux sistemlerinde `qm.ini` dosyasındaki Service stanza olanağını kullanın. Bir hizmet içindeki her bileşen için, o bileşene ilişkin kodu içeren modülün adını ve yolunu da belirtmeniz gerekir. UNIX and Linux sistemlerinde, bunun için ServiceComponent Stanza 'yı kullanın.

Ad =AuthorizationService|NameService

Gerekli hizmetin adı.

AuthorizationService

WebSphere MQ için, Authorization Service bileşeni nesne yetkili yöneticisi ya da OAM olarak bilinir.

Kuyruk yöneticisi oluşturulduğunda, AuthorizationService stanza ve ilişkili ServiceComponent stanza otomatik olarak eklenir. Diğer ServiceComponent stanzalarını el ile ekleyin.

NameService

Varsayılan değer olarak bir ad hizmeti sağlanmaz. Bir ad hizmetine gereksinim duyarsanız, NameService Stanza 'yı el ile eklemeniz gerekir.

EntryPoints=giriş sayısı

Hizmet için tanımlanan giriş noktalarının sayısı. Bu, kullanıma hazırlama ve sonlandırma giriş noktalarını içerir.

SecurityPolicy=Default|NTSIDsRequired (yalnızca Windows için WebSphere MQ)

SecurityPolicy özniteliği, belirtilen hizmet varsayılan yetkilendirme hizmetiysa, yani OAM ise geçerlidir. SecurityPolicy özniteliği, her kuyruk yöneticisi için güvenlik ilkesini belirtmenizi sağlar. Olası değerler şunlardır:

Default

Geçerli olmak için varsayılan güvenlik ilkesini kullanın. Bir Windows güvenlik tanıtıcısı (NT SID) belirli bir kullanıcı kimliği için OAM 'a geçirilmezse, ilgili güvenlik veritabanlarında arama yaparak uygun SID ' yi elde etmek için bir girişimde bulunmanız gerekir.

NTSIDsRequired

Güvenlik denetimleri gerçekleştirilirken bir NT SID 'yi OAM' ye geçirin.

Ek bilgi için [Windows güvenlik tanıtıcıları \(SID\) konusuna bakın.](#)

SharedBindingsUserId=kullanıcı-tipi

SharedBindingsUserId özniteliği, belirtilen hizmet varsayılan yetkilendirme hizmetiysa, yani OAM ise geçerlidir. SharedBindingsUserId özniteliği yalnızca paylaşılan bağ tanımlarıyla ilişkili olarak kullanılır. Bu değer, *IdentityContext* yapısındaki *UserIdentifier* alanının, MQZ_AUTHENTICATE_USER işlevinden, etkin kullanıcı kimliği mi, yoksa gerçek kullanıcı kimliği mi olduğunu belirtmenizi sağlar. MQZ_AUTHENTICATE_USER işleviyle ilgili bilgi için [MQZ_AUTHENTICATE_USER-Authenticate user](#) başlıklı konuya bakın. Olası değerler şunlardır:

Default

UserIdentifier alanının değeri, gerçek kullanıcı kimliği olarak ayarlanır.

Real

UserIdentifier alanının değeri, gerçek kullanıcı kimliği olarak ayarlanır.

yürürlükteki

The value of the *UserIdentifier* field is set as the effective user Id.

FastpathBindingsUserId=kullanıcı-tipi

FastpathBindingsUserId özniteliği, belirtilen hizmet varsayılan yetkilendirme hizmetiysa, yani OAM ise geçerlidir. FastpathBindingsUserId özniteliği, yalnızca fastpath bağ tanımlarıyla ilişkili olarak kullanılır. Bu değer, *IdentityContext* yapısındaki *UserIdentifier* alanının, MQZ_AUTHENTICATE_USER işlevinden, etkin kullanıcı kimliği mi, yoksa gerçek kullanıcı kimliği mi olduğunu belirtmenizi sağlar. MQZ_AUTHENTICATE_USER işleviyle ilgili bilgi için [MQZ_AUTHENTICATE_USER-Authenticate user](#) başlıklı konuya bakın. Olası değerler şunlardır:

Default

UserIdentifier alanının değeri, gerçek kullanıcı kimliği olarak ayarlanır.

Real

UserIdentifier alanının değeri, gerçek kullanıcı kimliği olarak ayarlanır.

yürürlükteki

The value of the *UserIdentifier* field is set as the effective user Id.

IsolatedBindingsUserId =kullanıcı-tipi

IsolatedBindingsUserId özniteliği, belirtilen hizmet varsayılan yetkilendirme hizmetiysa, yani OAM ise geçerlidir. IsolatedBindingsUserId özniteliği yalnızca yalıtılmış bağ tanımlarıyla ilişkili olarak kullanılır. Bu değer, *IdentityContext* yapısındaki *UserIdentifier* alanının, MQZ_AUTHENTICATE_USER işlevinden, etkin kullanıcı kimliği mi, yoksa gerçek kullanıcı kimliği mi olduğunu belirtmenizi sağlar. MQZ_AUTHENTICATE_USER işleviyle ilgili bilgi için [MQZ_AUTHENTICATE_USER-Authenticate user](#) başlıklı konuya bakın. Olası değerler şunlardır:

Default

The value of the *UserIdentifier* field is set as the effective user Id.

Real

UserIdentifier alanının değeri, gerçek kullanıcı kimliği olarak ayarlanır.

yürürlükteki

The value of the *UserIdentifier* field is set as the effective user Id.

Kurulabilir hizmetler ve bileşenler hakkında daha fazla bilgi için bkz. [Unix services and components for UNIX, Linux and Windows.](#)

Genel olarak güvenlik hizmetlerine ilişkin daha fazla bilgi için bkz. [Pencereler, UNIX and Linux sistemleri üzerinde güvenliğin ayarlanması.](#)

İlgili başvurular

[Kurulabilir hizmetler için başvuru bilgileri](#)

Hizmet bileşenleri

Yeni bir kurulabilir hizmet eklerken hizmet bileşeni bilgilerini belirtmeniz gerekir. Windows sistemlerinde `regedit` kullanın ve UNIX and Linux sistemlerinde `qm.ini` dosyasındaki `ServiceComponent` stanza olanağını kullanın. Varsayılan olarak yetkilendirme hizmeti kısmı var ve ilişkili bileşen (OAM) etkin.

Hizmet bileşenlerini aşağıdaki gibi belirtin:

Hizmet =*hizmet_adı*

Gerekli hizmetin adı. Bu, Hizmet yapılandırma bilgilerinin Name özniteliğinde belirtilen değerle eşleşmelidir.

Ad =*component_name*

Hizmet bileşeninin açıklayıcı adı. Bu benzersiz olmalı ve yalnızca WebSphere MQ nesnelere (örneğin, kuyruk adları) adları için geçerli olan karakterleri içermelidir. Bu ad, hizmet tarafından oluşturulan işletmen iletilerinde ortaya çıkar. Bu adın, bir şirket ticari markası ya da benzeri ayırt edici bir dizgiyle başladığını öneriyoruz.

Modül =*module_name*

Bu bileşene ilişkin kodu içerecek modülün adı. Bu tam yol adı olmalıdır.

ComponentDataBüyükülüğü =*boyut*

Her çağrışımında bileşene aktarılan bileşen verileri alanının bayt cinsinden büyükülüğü. Bileşen verisi gerekmiyorsa sıfır değerini belirtin.

Kurulabilir hizmetler ve bileşenler hakkında daha fazla bilgi için bkz. [Unix services and components for UNIX, Linux and Windows](#).

Kuyruk yöneticisi günlükleri

Bir kuyruk yöneticisinde oturum açmaya ilişkin bilgileri belirlemek için IBM WebSphere MQ Explorer ya da `qm.ini` dosyasındaki Log stanza adlı Log kuyruk yöneticisi özellikleri sayfasını kullanın.

Varsayılan olarak, bu ayarlar kuyruk yöneticisine ilişkin varsayılan günlük ayarları için belirtilen ayarlardan edinilir ("[IBM WebSphere MQ için günlük varsayılanları](#)" sayfa 418' ta açıklanmıştır). Bu ayarları yalnızca bu kuyruk yöneticisini farklı bir şekilde yapılandırmak istiyorsanız değiştirin.

Günlük boyutlarının hesaplanması hakkında bilgi için bkz. "[Günlüğün büyükülüğünün hesaplanması](#)" sayfa 395.

Not: Aşağıdaki parametre listesinde verilen sınırlar, WebSphere MQ tarafından ayarlanır. İşletim sistemi sınırları, olası günlük büyükülüğü üst sınırını azaltabilir.

LogPrimaryFiles =3 |2-254 (Windows) |2-510 (UNIX and Linux sistemleri)

Kuyruk yöneticisi yaratıldığında ayrılmış olan günlük dosyaları.

Sahip olduğunuz birincil günlük dosyası sayısı alt sınırı 2 'dir ve üst sınır değeri Windows' ta 254 ya da UNIX and Linux sistemlerindeki 510 'tır. Varsayılan 3'tür.

The total number of primary and secondary log files must not exceed 255 on Pencereler, or 511 on UNIX and Linux systems, and must not be less than 3.

Bu değer, kuyruk yöneticisi yaratıldığında ya da başlatıldığında incelenir. Bunu, kuyruk yöneticisi yaratıldıktan sonra değiştirebilirsiniz. Ancak, değer içindeki bir değişiklik kuyruk yöneticisi yeniden başlatılınca kadar yürürlüğe girmez ve etki hemen hemen olmayabilir.

LogSecondaryFiles =2 |1-253 (Windows) |1-509 (UNIX and Linux sistemleri)

Birincil dosyalar tükendiğinde ayrılmış olan günlük dosyaları.

The minimum number of secondary log files is 1 and the maximum is 253 on Pencereler, or 509 on UNIX and Linux systems. Varsayılan değer 2 'dir.

The total number of primary and secondary log files must not exceed 255 on Pencereler, or 511 on UNIX and Linux systems, and must not be less than 3.

Değer, kuyruk yöneticisi başlatıldığında incelenir. Bu değeri değiştirebilirsiniz, ancak kuyruk yöneticisi yeniden başlatılınca kadar değişiklikler yürürlüğe girmez ve etki hemen hemen olmayabilir.

LogFilePages =sayı

Günlük verileri, günlük dosyaları adı verilen bir dizi dosya içinde tutulur. Günlük dosyası büyüklüğü 4 KB ' lik sayfa birimlerinde belirtilir.

Günlük dosyası sayfalarının varsayılan sayısı 4096, günlük dosyası büyüklüğü 16 MB 'dir.

UNIX and Linux sistemlerinde günlük dosyası sayfası sayısı alt sınırı 64 'tür ve Windows ' da günlük dosyası sayfası sayısı alt sınırı 32 'dir; her iki durumda da üst sınır 65 535 'dir.

Not: Kuyruk yöneticisi yaratma işlemi sırasında belirtilen günlük kütüklerinin büyüklüğü, kuyruk yöneticisi için değiştirilemez.

LogType=CIRCULAR | LINEAR

Kuyruk yöneticisi tarafından kullanılacak günlüğe kaydetme tipi. Kuyruk yöneticisi yaratıldıktan sonra, kullanılacak günlüğe kaydetme tipini değiştiremezsiniz. Gereksinim duyduğunuz günlük kaydı tipine sahip bir kuyruk yöneticisi yaratılmasına ilişkin bilgi için, [“IBM WebSphere MQ için günlük varsayılanları” sayfa 418](#) içindeki LogType özneliğinin açıklamasına bakın.

CIRCULAR

Sistem durduğunda, devam etmekte olan işlemleri geri almak için günlüğü kullanarak kurtarma işlemi yeniden başlatın.

Dairesel günlüğe kaydetmenin tam açıklaması için bkz. [“Günlüğe kaydetme tipleri” sayfa 392](#) .

doğrusal

Yeniden başlatma kurtarma ve ortam ya da ileriye işleyerek kurtarma için (günlüğün içeriğini yeniden oynayarak kaybedilen ya da zarar görmüş veriler yaratılması).

Doğrusal günlüğe kaydetmenin tam açıklaması için bkz. [“Günlüğe kaydetme tipleri” sayfa 392](#) .

LogBufferPages =0 |0-4096

4 KB ' lik sayfa birimlerindeki arabelleklerin büyüklüğünü belirleyen, yazma için arabellek kayıtlarına ayrılan bellek miktarı.

Arabellek sayfa sayısı alt sınırı 18 ve üst sınır 4096 'tır. Daha büyük arabellekler, özellikle daha büyük iletiler için daha yüksek verimi elde eder.

0 değerini belirtirseniz (varsayılan değer), kuyruk yöneticisi büyüklüğü seçer. WebSphere MQ Sürüm 7.0 ' da bu değer 512 'dir (2048 KB).

1 ile 17 aralığında bir sayı belirtirseniz, kuyruk yöneticisi varsayılan olarak 18 (72 KB) değerine ayarlanır. 18 ile 4096 aralığında bir sayı belirtirseniz, kuyruk yöneticisi ayrılmış belleği ayarlamak için belirtilen sayıyı kullanır.

Değer, kuyruk yöneticisi başlatıldığında incelenir. Değer, belirtilen sınırlar içinde artırılabilir ya da azaltılabilir. Ancak, değer değişikliği, kuyruk yöneticisi bir sonraki başlatılıncaya kadar yürürlüğe girmez.

LogPath=directory_name

Bir kuyruk yöneticisine ilişkin günlük kütüklerinin bulunduğu dizin. Bu, kuyruk yöneticisinin, ileti kuyruklarından farklı bir sürücüde yazabileceği ve tercihen yazabileceği yerel bir aygıtta var olmalıdır. Farklı bir sürücü belirlenmesi, sistem arızası durumunda ek koruma sağlar.

Varsayılan:

- C:\Program Files\IBM\WebSphere MQ\log in WebSphere MQ for Pencereler .
- UNIX and Linux sistemleri için WebSphere MQ ' da/var/mqm/log .

crtmqm komutundaki bir dizinin adını -ld işaretini kullanarak belirleyebilirsiniz. Bir kuyruk yöneticisi yaratıldığında, kuyruk yöneticisi dizini altında da bir dizin yaratılır ve bu dizin günlük kütüklerini tutmak için kullanılır. Bu dizinin adı kuyruk yöneticisi adını temel alır. Bu, günlük dosyası yolunun benzersiz olmasını ve dizin adı uzunlukları üzerinde herhangi bir sınırlamaya uymasını sağlar.

crtmqm komutunda -ld seçeneğini belirlemezseniz, LogDefaultPath özneliğinin değeri kullanılır.

WebSphere MQ for UNIX and Linux sistemlerinde, kullanıcı kimliği mqm ve group mqm, günlük kütüklerine tam yetkiler olmalıdır. Bu dosyaların konumlarını değiştirirseniz, bu yetkileri kendiniz de

vermelisiniz. Günlük dosyaları ürünle birlikte sağlanan varsayılan konumlarda yer almıyorsa, bu gerekli değildir.

LogWriteIntegrity =SingleWrite|DoubleWrite| TripleWrite

Günlüğe kaydedicinin, günlük kayıtlarını güvenilir olarak yazmak için kullandığı yöntem.

TripleWrite

Bu varsayılan yöntemdir.

Not, **DoubleWrite** seçeneğini belirleyebileceğinden, ancak bunu yapabiliyorsa, sistem bunu **TripleWrite** olarak yorumluyor.

SingleWrite

You should use **SingleWrite**, only if the file-system or device hosting the WebSphere MQ recovery log explicitly guarantees the atomicity of 4KB writes.

Yani, bir 4KB sayfasının yazma işlemi herhangi bir nedenle başarısız olduğunda, olası tek iki durum, önceki görüntüden ya da görüntünün sonrasına sahip olabilir. Hiçbir ara durum mümkün değil.

Kısıtlı kip

Bu seçenek yalnızca UNIX and Linux sistemleri için geçerlidir. RestrictedMode stanza, **crtmqm** komutundaki -g seçeneği ile ayarlanır. Kuyruk yöneticisi yaratıldıktan sonra bu kısmı değiştirmeyin. If you do not use the -g option, the stanza is not created in the qm.ini file.

There are some directories under which IBM WebSphere MQ applications create files while they are connected to the queue manager within the queue manager data directory. Uygulamaların bu dizinlerde dosya yaratması için, bu dosyalara dünya yazma erişimi verilir:

- /var/mqm/sockets/QMgrName/@ipcc/ssem/hostname/
- /var/mqm/sockets/QMgrName/@app/ssem/hostname/
- /var/mqm/sockets/QMgrName/zsocketapp/hostname/

Burada <QMGRNAME> , kuyruk yöneticisinin adıdır; <hostname> ise anasistem adıdır.

Bazı sistemlerde, tüm kullanıcılara bu dizinlere yazma erişimi vermek kabul edilemez. Örneğin, kuyruk yöneticisine erişmeye gerek kalmayan kullanıcılar. Kısıtlı kip, kuyruk yöneticisi verilerini saklayan dizinlerin izinlerini değiştirir. Daha sonra, dizinlere yalnızca belirtilen uygulama grubunun üyeleri erişebilir. Kuyruk yöneticisiyle iletişim kurmak için kullanılan System V IPC paylaşılan belleğindeki izinler de aynı şekilde değiştirilir.

Uygulama grubu, aşağıdaki bilgileri yapma iznine sahip üyelerle birlikte grubun adıdır:

- MQI uygulamalarını çalıştır
- tüm IPCC kaynaklarını güncelle
- bazı kuyruk yöneticisi dizinlerinin içeriğini değiştirme

Kuyruk yöneticisi için kısıtlı kipi kullanmak üzere:

- Kuyruk yöneticisinin yaratıcısı, mqm grubunda ve uygulama grubunda yer almalıdır.
- mqm kullanıcı kimliği, uygulama grubunda yer almalıdır.
- Kuyruk yöneticisini yönetmek isteyen tüm kullanıcıların mqm grubunda ve uygulama grubunda yer almaları gerekir.
- IBM WebSphere MQ uygulamalarını çalıştırmak isteyen tüm kullanıcıların uygulama grubu içinde olması gerekir.

Uygulama grubunda olmayan bir kullanıcı tarafından yayınlanan herhangi bir MQCONN ya da MQCONNX çağırısı MQRC_Q_MGR_NOT_AVAM neden koduyla başarısız oldu.

Sınırlı kip, IBM WebSphere MQ yetkilendirme hizmetiyle çalışır. Bu nedenle, kullanıcılara IBM WebSphere MQ ' a bağlanma yetkisi de vermeniz ve IBM WebSphere MQ yetki hizmetini kullanarak gereken kaynaklara erişmelerini de vermeniz gerekir.

UNIX Linux Windows IBM WebSphere MQ yetki hizmetinin yapılandırılmasıyla ilgili ek bilgi [Pencereler, UNIX and Linux sistemleri üzerinde güvenliğin ayarlanması](#) içinde bulunabilir.

Yetkilendirme hizmeti tarafından sağlanan denetim, kuyruk yöneticisi kaynakları için yeterli yalıtım sağlamadığında yalnızca IBM WebSphere MQ kısıtlamalı kipi kullanın.

XA kaynak yöneticileri

Kuyruk yöneticisi tarafından eşgüdümlü genel iş birimlerinde yer alan kaynak yöneticilerine ilişkin aşağıdaki bilgileri belirtmek için IBM WebSphere MQ Gezgini 'nden ya da qm.ini dosyasındaki XAResourceManager ' dan XA resource manager kuyruk yöneticisi özellikler sayfasını kullanın.

Genel iş birimlerine katılan bir kaynak yöneticisinin her bir eşgörünümü için XA kaynak yöneticisi yapılanış bilgilerini el ile ekleyin; varsayılan değer sağlanmaz.

Kaynak yöneticisi özniteliklerine ilişkin ek bilgi için [Veritabanı eşgüdümü](#) konusuna bakın.

Ad =ad (zorunlu)

Bu öznitelik, kaynak yöneticisi yönetim ortamını tanıtır.

Name değeri en çok 31 karakter uzunluğunda olabilir. Kaynak yöneticisi adını XA-anahtar yapısında tanımlandığı şekilde kullanabilirsiniz. Ancak, aynı kaynak yöneticisinin birden çok eşgörünümü kullanıyorsanız, her yönetim ortamı için benzersiz bir ad oluşturmalsınız. You can ensure uniqueness by including the name of the database in the Name string, for example.

WebSphere MQ uses the Name value in messages and in output from the dspmqtrn command.

Bir kaynak yöneticisi yönetim ortamının adını değiştirmeyin ya da ilişkili kuyruk yöneticisi başlatıldıktan ve kaynak yöneticisi adı yürürlükte olduğunda, bu yönetim ortamının girişini yapılanış bilgilerinden silmeyin.

SwitchFile=ad (zorunlu)

Kaynak yöneticisinin XA anahtar yapısını içeren yükleme dosyasının tam olarak nitelenmiş adı.

32 bit kullanan uygulamalar içeren 64 bit kuyruk yöneticisi kullanıyorsanız, name değeri yalnızca kaynak yöneticisinin XA anahtar yapısını içeren yükleme dosyasının temel adını içermelidir.

32 bitlik dosya, uygulamaya ExitsDefaultPath tarafından belirtilen yoldan yüklenir.

The 64-bit file will be loaded into the queue manager from the path specified by ExitsDefaultPath64.

XAOpenString=dizgi (isteğe bağlı)

Kaynak yöneticisinin xa_open giriş noktasına geçirilecek veri dizgisi. Dizilimin içeriği, kaynak yöneticisinin kendisine bağlıdır. Örneğin, dizgi, kaynak yöneticisinin bu yönetim ortamının erişeceği veritabanını tanımlamayı sağlar. Bu özniteliği tanımlamaya ilişkin ek bilgi için aşağıdaki başlara bakın:

- [DB2 için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)
- [Oracle için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)
- [Sybase için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)
- [Informix için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)

ve uygun dizgi için kaynak yöneticisi belgelerinize bakın.

XACloseString=dizgi (isteğe bağlı)

Kaynak yöneticisinin xa_close giriş noktasına geçirilecek veri dizgisi. Dizilimin içeriği, kaynak yöneticisinin kendisine bağlıdır. Bu özniteliği tanımlamaya ilişkin ek bilgi için aşağıdaki başlara bakın:

- [DB2 için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)
- [Oracle için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)
- [Sybase için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)
- [Informix için kaynak yöneticisi yapılanış bilgileri eklenmesi](#)

ve ilgili dizgiye ilişkin veritabanı belgelerinize bakın.

ThreadOfControl=THREAD | PROCESS

Bu öznitelik, WebSphere MQ for Pencereleri için zorunludur. Kuyruk yöneticisi bu değeri, kendi çok iş parçacıklı süreçlerinden birinin kaynak yöneticisini araması gerektiğinde diziselleştirme için kullanır.

Başlık

Kaynak yöneticisi tam olarak *iş parçacığı farkındadır*. Çok iş parçacıklı bir WebSphere MQ işleminde, aynı anda birden çok iş parçacığının dış kaynak yöneticisine XA işlev çağrılarını yapabilir.

PROCESS

Kaynak yöneticisi *iş parçacığı güvenlideğil*. Çok iş parçacıklı bir WebSphere MQ işleminde, kaynak yöneticisinde aynı anda yalnızca bir XA işlev çağrısı yapılabilir.

The ThreadOfControl entry does not apply to XA function calls issued by the queue manager in a multithreaded application process. Genel olarak, farklı iş parçacıklarına ilişkin eşzamanlı çalışma birimlerine sahip bir uygulama, bu işlem kipinin kaynak yöneticilerinin her biri tarafından desteklenmesini gerektirir.

Kanalların öznitelikleri

Bu öznitelikler bir kanalın yapılandırmasını belirler.

Bu bilgiler, z/OS platformu için WebSphere MQ için geçerli değildir.

Kanallara ilişkin bilgileri belirtmek için, WebSphere MQ Gezgini 'nden ya da qm.ini dosyasındaki CHANNELS stanza olanağından Channels kuyruk yöneticisi özellikler sayfasını kullanın.

MaxChannels=100 |sayı

İzin verilen *geçerli* kanal sayısı üst sınırı.

Değer 1-65535 aralığında olmalıdır. Varsayılan değer 100 değeridir.

MaxActiveKanalları =MaxChannels_value

Herhangi bir zamanda *etkin* olması için izin verilen kanal sayısı üst sınırı. Varsayılan değer, MaxChannels özniteliği için belirtilen değerdir.

MaxInitiators=3 |sayı

Başlatıcı sayısı üst sınırı. Varsayılan değer ve üst sınır değeri 3 'tür.

MQIBindType= FASTPATH | STANDARD

Uygulamalar için bağ tanımı:

FastPath

Kanallar MQCONNX FASTPATH kullanılarak bağlantı sağlar; aracı işlemi yoktur.

Standart

Kanallar STANDARD kullanarak bağlantı sağlar.

PipeLineUzunluk =1 |sayı

Bir kanalın kullanacağı koşut zamanlı iş parçacığı sayısı üst sınırı. Varsayılan değer 1'dir. 1 'den büyük herhangi bir değer 2 olarak kabul edilir.

Pipelining (pipeting) kullandığınızda, kanalın her iki ucundaki kuyruk yöneticilerini, 1 'den büyük bir *PipeLineLength* değerine sahip olmak için yapılandırın.

Not: Pipelining yalnızca TCP/IP kanalları için geçerlidir.

AdoptNewMCA=N0| SVR | SDR | RCVR | CLUSRCVR | ALL | FASTPATH

WebSphere MQ bir kanalı başlatmak için bir istek alırsa, ancak kanala ilişkin bir yönetim ortamının zaten çalıştığını fark ederse, yeni bir kanal yönetim ortamının yeni başlatılmadan önce var olan kanal yönetim ortamının durdurulması gerekir. AdoptNewMCA özniteliği, bu şekilde sona erdirilebilecek kanal tiplerini denetleyebilmenizi sağlar.

Belirli bir kanal tipi için AdoptNewMCA özniteliğini belirtirseniz, ancak eşleşen bir kanal yönetim ortamı zaten çalıştığı için yeni kanal başlatılamazsa:

1. Yeni kanal, bitmesini isteyerek öncekini durdurmayı deniyor.
2. Önceki kanal sunucusu, AdoptNewMCATimeout bekleme süresinin sona ermesiyle bu isteğe yanıt vermezse, önceki kanal sunucusuna ilişkin iş parçacığı ya da işlem sona erdirilir.

3. Önceki kanal sunucusu 2. adımdan sonra sona ermediyse ve AdoptNewMCATimeout bekleme süresi ikinci kez sona erdikten sonra, WebSphere MQ kanalı CHANNEL IN USE hatası ile sona erdirir.

AdoptNewMCA işlevselliği sunucu, gönderen, alıcı ve küme alıcı kanallarına uygulanır. Gönderici ya da sunucu kanalı olması durumunda, alan kuyruk yöneticisinde belirli bir adı taşıyan bir kanaldan yalnızca bir yönetim ortamı çalıştırılabilir. Bir günlük nesnesi ya da küme alıcı kanalisında, alan kuyruk yöneticisinde belirli bir adı taşıyan birden çok kanal eşgörünümü çalışıyor olabilir, ancak belirli bir uzak kuyruk yöneticisinden herhangi bir zamanda tek bir eşgörünüm çalıştırılabilir.

Not: AdoptNewMCA, requester ya da sunucu bağlantısı kanallarında desteklenmez.

Aşağıdaki listeden, virgülle ya da boşluklarla ayırarak bir ya da daha çok değer belirtin:

HAYIR

AdoptNewMCA özelliği gerekli değildir. Bu varsayılandır.

SVR

Sunucu kanallarını kabul edin.

DR

Gönderici kanalları evlat edinir

RVR

Alıcı kanallarını kabul et.

CLUSRCVR

Küme alıcı kanallarını kabul edin.

TÜMÜ

FESTPATH kanalları dışında tüm kanal tiplerini kabul edin.

FastPath

Kanal, bir FASTPATH kanalıysa, kanalı benimseyir. Bu yalnızca uygun kanal tipi de belirtilirse oluşur, örneğin: AdoptNewMCA=RCVR , SVR , FASTPATH.

Dikkat!: AdoptNewMCA özniteliği, FASTPATH kanallarıyla öngörülemez bir şekilde işlem görülebilirler. FASTPATH kanallarına ilişkin AdoptNewMCA özniteliğini etkinleştirirken çok dikkatli olun.

AdoptNewMCATimeout=60 | 1-3600

Yeni kanal örneğinin eski kanal yönetim ortamının sona ermesi için bekleyeceği süre (saniye). 1-3600 aralığında bir değer belirtin. Varsayılan değer 60'tır.

AdoptNewMCACheck = QM | ADDRESS | NAME | ALL

AdoptNewMCA özniteliğini etkinleştirirken gerekli olan denetmenin tipi. Mümkünse, kanallarınızı kapatmak, istemeden ya da kötü bir şekilde korumak için tam denetme işlemi gerçekleştirin. En azından, kanal adlarının eşleşip eşleşmediğinden emin olun.

Aşağıdaki değerlerden birini ya da birkaçını, QM, ADya da Tüm durumunda virgülle ya da boşluklarla ayırarak belirtin:

QM

Kuyruk yöneticisi adlarının eşleşip eşleşmediğinden emin olun.

Kuyruk yöneticisi adının kendisinin QMID değil, eşleştirildiğine dikkat edin.

ADRES

İletişim kaynağı IP adresini denetleyin. Örneğin, TCP/IP adresi.

Not: Virgülle ayrılmış CONNAME değerleri, hedef adresler için geçerlidir ve bu nedenle bu seçenekle ilgili değildir.

Çok eşgörünümlü bir kuyruk yöneticisinin hosta 'tan hostb' a kadar başarısız olması durumunda, o kuyruk yöneticisinden giden tüm giden kanallar hostbkaynak IP adresini kullanır. Bu hosta' dan farklıysa, AdoptNewMCACheck=ADDRESS eşleşmemesi başarısız olur.

Bir saldırganın var olan bir çalışan kanalı kesintiye uğratmasını önlemek için SSL ya da TLS ' yi karşılıklı kimlik doğrulamasıyla kullanabilirsiniz. Diğer bir seçenek olarak, çok eşgörünümlü kuyruk

yöneticileri yerine IP devralma işlemi içeren bir HACMP tipi bir çözüm kullanın ya da kaynak IP adresini maskeleyerek için bir ağ yükleme dengeleyici kullanın.

AD

Kanal adlarının eşleşip eşleşmediğine bakın.

TÜMÜ

Eşleşen kuyruk yöneticisi adları, iletişim adresi ve eşleşen kanal adları olup olmadığını denetleyin.

Varsayılan değer AdoptNewMCACheck=NAME , ADDRESS , QM' dir.

İlgili kavramlar

“Kanal durumları” sayfa 54

Bir kanal, her an birçok eyaletten birinde olabilir. Bazı eyaletlerde alt devletler de vardır. Belirli bir durumdan bir kanal başka eyaletler arasında geçiş yapabilir.

TCP, LU62, NETBIOS ve SPX

Ağ iletişim kuralı yapılandırma parametrelerini belirlemek için bu kuyruk yöneticisi özellikleri sayfalarını ya da qm.ini dosyasındaki stanzaları kullanın. Kanallar için varsayılan öznitelikleri geçersiz kılar.

TCP

İletim Denetimi İletişim Kuralı/Internet Protocol (TCP/IP) yapılandırma parametrelerini belirlemek için IBM WebSphere MQ Explorer ya da qm.ini dosyasındaki TCP stanza içindeki TCP kuyruk yöneticisi özellikleri sayfasını kullanın.

Kapı =1414|kapı_numarası

TCP/IP oturumları için varsayılan kapı numarası (ondalık gösterim). WebSphere MQ için *bilinen* kapı numarası 1414 'tür.

Library1 =DLLName1 (yalnızca Windows için WebSphere MQ)

TCP/IP yuva DLL ' inin adı.

Varsayılan değer WSOCK32' dir.

KeepAlive=NO|YES

KeepAlive işlevini açık ya da kapalı olarak değiştirin. KeepAlive=YES, TCP/IP ' nin bağlantının diğer ucunun hala kullanılabilir durumda olduğunu düzenli aralıklarla denetmesine neden olur. Değilse, kanal kapatılır.

ListenerBacklog= sayı

TCP/IP dinleyicisine ilişkin varsayılan bekleyen istek sayısını geçersiz kılar.

TCP/IP ' de giriş yaparken, bekleyen bağlantı isteği sayısı üst sınırı belirlendi. Bu, dinleyicinin isteği kabul etmesi için TCP/IP kapısında bekleyen isteklerin bir *birikim günlüğü* olarak düşünülebilmektedir. Varsayılan dinleyici birikim günlüğü değerleri [Çizelge 33 sayfa 432](#) içinde gösterilir.

Çizelge 33. Varsayılan bekleyen bağlantı istekleri (TCP)	
Altyapı	Varsayılan ListenerBacklog değeri
Windows Server	100
Windows İş İstasyonu	5
Linux	100
Solaris	100
HP-UX	20
AIX V4.2 ya da sonraki sürümü	100
AIX V4.1 ya da önceki yayın düzeyi	10

Not: Bazı işletim sistemleri, gösterilen varsayılan değerden daha büyük bir değeri destekler. Bağlantı sınırına ulaşmamak için bunu kullanın.

Tersi durumda, bazı işletim sistemleri TCP birikim birikimi büyüklüğünü sınırlayabilir, bu nedenle etkin TCP arka günlüğü burada istenenden daha küçük olabilir.

Birikim, Çizelge 33 sayfa 432 içinde gösterilen değerlere ulaşırsa, TCP/IP bağlantısı reddedilir ve kanal başlatılamaz. İleti kanalları için, kanaldaki bu sonuçlar bir RETRY durumuna geçiyor ve bağlantıyı daha sonra yeniden deniyor. İstemci bağlantıları için, istemci MQCONN komutundan bir MQRC_Q_MGR_NOT_AVALABILIR neden kodu alır ve bağlantıyı daha sonra yeniden dener.

SvrSndBuffSize=32768|number

İstemci-bağlantı sunucusu bağlantı kanalının sunucu ucu tarafından kullanılan TCP/IP gönderme arabelleğindeki bayt cinsinden büyüklük.

SvrRcvBuffSize=32768|number

İstemci-bağlantı sunucusu bağlantı kanalının sunucu ucu tarafından kullanılan TCP/IP alma arabelleğindeki bayt cinsinden büyüklük.

Connect_Timeout=0|number

Yuva zamanlarını dışarı bağlama girişiminden önce geçecek süre (saniye). Varsayılan değer olarak varsayılan değer, bağlantı zaman aşımı olmadığını belirtir.

LU62 (yalnızca Pencereleler için WebSphere MQ)

SNA LU 6.2 iletişim kuralı yapılandırma değiştirgelerini belirtmek için, IBM WebSphere MQ Explorer 'dan ya da qm.ini kütüğündeki LU62 stanza ögesini kullanarak LU6.2 kuyruk yöneticisi özellikleri sayfasını kullanın.

TPN Adı

Uzak yerde başlatılacak TP adı.

Kitaplık1 =DLLName 1

APPC DLL 'in adı.

Varsayılan değer WCPIC32' dir.

Kitaplık2 =DLLName2

Kod iki ayrı kitaplıkta saklandıysa, Library1 ile aynıdır.

Varsayılan değer WCPIC32' dir.

NETBIOS (yalnızca Windows için WebSphere MQ)

NetBIOS iletişim kuralı yapılandırma parametrelerini belirlemek için IBM WebSphere MQ Explorer ya da qm.ini dosyasındaki NETBIOS stanza ile Netbios kuyruk yöneticisi özellikler sayfasını kullanın.

LocalName=ad

Bu makinenin LAN üzerinde bilindiği ad.

AdapterNum=0|adapter_number

LAN bağdaştırıcısının numarası. Varsayılan değer, bağdaştırıcı 0 'tır.

NumSess=1|number_of_sessions

Ayrılacak oturumların sayısı. Varsayılan değer 1'dir.

NumCmds=1|number_of_commands

Ayrılacak komutların sayısı. Varsayılan değer 1'dir.

NumNames=1|number_of_names

Ayrılacak ad sayısı. Varsayılan değer 1'dir.

Kitaplık1 =DLLName1

NetBIOS DLL 'inin adı.

Varsayılan değer NETAPI32' dir.

SPX (yalnızca Windows için WebSphere MQ)

SPX iletişim kuralı yapılandırma parametrelerini belirlemek için IBM WebSphere MQ Explorer ya da qm.ini dosyasındaki SPX stanza olanağından SPX kuyruk yöneticisi özellikler sayfasını kullanın.

Yuva =5E86|socket_number

Onaltılı gösterimde SPX yuvası sayısı. Varsayılan değer X'5E86' dir.

BoardNum=0|bağdaştırıcı_numarası

LAN bağdaştırıcısı numarası. Varsayılan değer, bağdaştırıcı 0 'tır.

KeepAlive= NO | YES (EVET)

KeepAlive işlevini açık ya da kapalı olarak değiştirin.

KeepAlive=YES, SPX 'in, bağlantının diğer ucunun hala kullanılabilir durumda olduğunu düzenli aralıklarla denetmesine neden olur. Değilse, kanal kapatılır.

Kitaplık1 =DLLName1

SPX DLL ' inin adı.

Varsayılan değer WSOCK32.DLL(DLL)

Kitaplık2 =DLLName2

Kod iki ayrı kitaplıkta saklandıysa, LibraryName1 ile aynı.

Varsayılan değer WSOCK32.DLL(DLL)

ListenerBacklog= sayı

SPX dinleyicisine ilişkin varsayılan bekleyen istek sayısını geçersiz kılar.

SPX 'te alınırken, bekleyen bağlantı isteği sayısı üst sınırı belirlendi. Bu, dinleyicinin isteği kabul etmesi için SPX yuvasında bekleyen isteklerin bir *birikim günlüğü* olarak düşünülebilmektedir. Varsayılan dinleyici birikim günlüğü değerleri Çizelge 34 sayfa 434 içinde gösterilir.

Çizelge 34. Varsayılan bekleyen bağlantı istekleri (SPX)	
Altyapı	Varsayılan ListenerBacklog değeri
Windows Server	100
Windows İş İstasyonu	5

Not: Bazı işletim sistemleri, gösterilen varsayılan değerden daha büyük bir değeri destekler. Bağlantı sınırına ulaşmamak için bunu kullanın.

Tersi durumda, bazı işletim sistemleri SPX birikim günlüğünün büyüklüğünü sınırlayabilir, bu nedenle etkin SPX birikim günlüğü burada istenenden daha küçük olabilir.

Birikim, Çizelge 34 sayfa 434 içinde gösterilen değerlere ulaşırsa, SPX bağlantısı reddedilir ve kanal başlatılamaz. İleti kanalları için, kanaldaki bu sonuçlar bir RETRY durumuna geçiyor ve bağlantıyı daha sonra yeniden deniyor. İstemci bağlantıları için, istemci MQCONN komutundan bir MQRC_Q_MGR_NOT_AVAM neden kodu alır ve daha sonra bağlantıyı yeniden denemiyor.

Çıkış yolu

Kuyruk yöneticisi sistemindeki kullanıcı çıkış programlarına ilişkin yolu belirtmek için IBM WebSphere MQ Gezgini 'nden Exits kuyruk yöneticisi özellikleri sayfasını ya da qm.ini dosyasındaki ExitPath stanza sayfasını kullanın.

ExitsDefaultYolu =dizgi

ExitsDefaultYolu özniteliği, aşağıdaki konumun yerini belirtir:

- İstemciler için 32 bitlik kanal çıkışları
- Sunucular için 32 bitlik kanal çıkışları ve veri dönüştürme çıkışları
- Nitelenmemiş XA anahtarı yükleme dosyaları

ExitsDefaultPath64 =dizgi

ExitsDefaultPath64 özniteliği, aşağıdakilerin yerini belirtir:

- İstemciler için 64 bitlik kanal çıkışları
- Sunucular için 64 bitlik kanal çıkışları ve veri dönüştürme çıkışları

- Nitelenmemiş XA anahtarı yükleme dosyaları

API çıkışları

For a server, use the Exits queue manager properties page from the IBM WebSphere MQ Explorer, or the ApiExitLocal stanza in the qm.ini file to identify API exit routines for a queue manager. Bir istemci için, bir kuyruk yöneticisine ilişkin API çıkış yordamlarını tanımlamak için mqclient.ini dosyasında ApiExitLocal stanza 'yı değiştirin.

Windows sistemlerinde, API çıkışlarına ilişkin girdileri değiştirmek için amqmdain komutunu da kullanabilirsiniz. (To identify API exit routines for all queue managers, you use the ApiExitCommon and ApiExitTemplate stanzas, as described in “API çıkışları” sayfa 421.)

API çıkışı doğru şekilde çalışacağı için, sunucudan gelen iletinin istemciye dönüştürülemeyen bir ileti gönderilmesi gerekir. API çıkışı iletiyi işledikten sonra, ileti istemciye dönüştürülmelidir. Bu nedenle, istemciye tüm dönüştürme çıkışlarını kurmanızı gerektirir.

Bu stanzalara ilişkin özniteliklere ilişkin eksiksiz açıklamalar için [API çıkışlarının yapılandırılması](#) başlıklı konuya bakın.

UNIX, Linux, and Windows üzerindeki QMErrorLog Stanza

Use the Extended queue manager properties page from the WebSphere MQ Explorer, or the QMErrorLog stanza in the qm.ini file to tailor the operation and contents of queue manager error logs.



Uyarı: You can use WebSphere MQ Explorer to make the changes, only if you are using a local queue manager on the Windows platform.

ErrorLogBoyutu =maxsize

Yedekleme işlemi için kopyalandığı kuyruk yöneticisi hata günlüğünün büyüklüğünü belirler. maxsize , 32768 ile 2147483648 byte arasında bir değer olmalıdır. ErrorLogSize belirtilmemişse, varsayılan değer olan 2097152 bayt (2 MB) kullanılır.

ExcludeMessage=msgIds

Kuyruk yöneticisi hata günlüğüne yazılmamış iletileri belirtir. If your WebSphere MQ system is heavily used, with many channels stopping and starting, a large number of information messages are sent to the z/OS console and hardcopy log. The WebSphere MQ-IMS bridge and buffer manager might also produce a large number of information messages, so excluding messages prevents you from receiving a large number of messages if you require it. msgIds , aşağıda belirtilen ileti tanıtıcılarının virgülle ayrılmış bir listesini içerir:

- 5211-Özellik adı uzunluğu üst sınırı aşıldı.
- 5973-Distributed publish/subscreen aboneliği engellenmiş
- 5974-Distributed publish/subscreen yayını engellenmiş
- 6254-Sistem paylaşılan kitaplığı devingen olarak yükleyemedi
- 7234-Yüksel ileti sayısı
- 9001-Kanal programı olağan şekilde sona erdi
- 9002-Kanal programı başlatıldı
- 9202-Uzak anasistem kullanılamıyor
- 9208-Anasistemden alma hatası
- 9209-Bağlantı kapatıldı
- 9228-Kanal yanıtlayıcısı başlatılamıyor
- 9489-SVRCONN eşgörünüm sayısı üst sınırı aşıldı
- 9490-SVRCONN istemci sınırı başına eşgörünüm sayısı üst sınırı aşıldı
- 9508-Kuyruk yöneticisiyle bağlantı kurulamıyor
- 9524-Uzak kuyruk yöneticisi kullanılamıyor
- 9528-Kullanıcı, kanal kapatma isteğinde bulundu
- 9558-Uzak Kanal kullanılamıyor
- 9637-Kanalda sertifika yok
- 9776-Kanal kullanıcı kimliği tarafından engellendi

9777-Kanal, NOCESS eşlemi tarafından engellendi
9782-Bağlantı adresi engellendi
9999-kanal programı olağan dışı sona erdi

SuppressMessage=msgIds

Kuyruk yöneticisi hata günlüğüne yalnızca belirli bir zaman aralığında bir kez yazılan iletileri belirtir. If your WebSphere MQ system is heavily used, with many channels stopping and starting, a large number of information messages are sent to the z/OS console and hardcopy log. WebSphere MQ-IMS köprüsü ve arabellek yöneticisi çok sayıda bilgi ileti de üretebilir; bu nedenle, iletilerin gizleniyor olması gerekiyorsa, yinelenen iletilerin sayısını almanıza engel olur. Zaman aralığı SuppressInterval tarafından belirtilir. *msgIds* , aşağıda belirtilen ileti tanıtıcılarının virgülle ayrılmış bir listesini içerir:

5211-Özellik adı uzunluğu üst sınırı aşıldı.
5973-Distributed publish/subscreen aboneliği engellenmiş
5974-Distributed publish/subscreen yayını engellenmiş
6254-Sistem paylaşılan kitaplığı devingen olarak yükleyemedi
7234-Yüksel ileti sayısı
9001-Kanal programı olağan şekilde sona erdi
9002-Kanal programı başlatıldı
9202-Uzak anasistem kullanılamıyor
9208-Anasistemden alma hatası
9209-Bağlantı kapatıldı
9228-Kanal yanıtlayıcısı başlatılamıyor
9489-SVRCONN eşgörünüm sayısı üst sınırı aşıldı
9490-SVRCONN istemci sınırı başına eşgörünüm sayısı üst sınırı aşıldı
9508-Kuyruk yöneticisiyle bağlantı kurulamıyor
9524-Uzak kuyruk yöneticisi kullanılamıyor
9528-Kullanıcı, kanal kapatma isteğinde bulundu
9558-Uzak Kanal kullanılamıyor
9637-Kanalda sertifika yok
9776-Kanal kullanıcı kimliği tarafından engellendi
9777-Kanal, NOCESS eşlemi tarafından engellendi
9782-Bağlantı adresi engellendi
9999-kanal programı olağan dışı sona erdi

Hem SuppressMessage hem de ExcludeMessage' da aynı ileti tanıtıcısı belirtilirse, ileti dışlanır.

SuppressInterval=uzunluk

SuppressMessage içinde belirtilen iletilerin yalnızca bir kez kuyruk yöneticisi hata günlüğüne yazıldığı saniye cinsinden zaman aralığını belirtir. *uzunluk* , 1 ile 86400 arasında bir değer aralığında olmalıdır. SuppressInterval belirtilmemişse, varsayılan değer olan 30 saniye kullanılır.

Kuyruk yöneticisi varsayılan bağ tanımlama tipi

Varsayılan bağ tanımlama tipini belirtmek için IBM WebSphere MQ Gezgini 'nden Extended kuyruk yöneticisi özellikleri sayfasını ya da qm.ini dosyasındaki Connection stanza ögesini kullanın.

DefaultBindTip =SHARED|ISOLATED

DefaultBindTipi ISOLATED olarak ayarlandıysa, uygulamalar ve kuyruk yöneticisi ayrı süreçlerde çalıştırılır ve bunlar arasında hiçbir kaynak paylaşımıyorsa.

DefaultBindtipi SHARED ise, uygulamalar ve kuyruk yöneticisi ayrı süreçlerde çalıştırılır, ancak aralarında bazı kaynaklar paylaşılır.

Varsayılan değer SHARED ' dir.

Kuyruk yöneticisi yapılandırma dosyasının SSL ve TLS kısmı

Kuyruk yöneticinizin SSL ya da TLS kanallarını yapılandırmak için kuyruk yöneticisi yapılandırma kütüğünün SSL kısmına bakın.

OCSP (Çevrimiçi Sertifika Durumu Protokolü)

Bir sertifika, bir AuthorityInfoerişimi uzantısını içerebilir. Bu uzanti, OCSP (Online Certificate Status Protocol; Çevrimiçi Sertifika Durumu Protokolü) aracılığıyla bağlantı kurulacak bir sunucuyu belirtir. Kuyruk yöneticinizdeki SSL ya da TLS kanallarına AuthorityInfoAccess uzantılarını kullanmak üzere izin vermek için, bunların içinde belirtilen OCSP sunucusunun kullanılabilir durumda olduğundan, doğru yapılandırıldığından ve ağ üzerinden erişilebilir olduğundan emin olun. Ek bilgi için [İptal edilen sertifikalarla çalışmabaşlıklı konuya](#) bakın.

CrlDistributionNoktası (CDP)

Bir sertifika, bir CrlDistributionNokta uzantısını içerebilir. Bu uzanti, bir sertifika iptal listesi (CRL) yüklemek için kullanılan protokolü ve iletişim kurulacak sunucuyu tanıtan bir URL içerir.

Kuyruk yöneticinizin SSL ya da TLS kanallarına CrlDistributionNokta uzantılarını kullanmasını sağlamak istiyorsanız, bunların içinde adı geçen CDP sunucusunun kullanılabilir, doğru olarak yapılandırılmış ve ağ üzerinden erişilebilir olduğundan emin olun.

SSL Stanza

Kuyruk yöneticinizdeki SSL ya da TLS kanallarının aşağıdaki olanakları nasıl kullanmayı denediğini ve bunlar kullanılırken sorunlar ortaya çıktığında nasıl tepki vereceklerini yapılandırmak için `qm.ini` dosyasındaki SSL stanza olanağını kullanın.

Aşağıdaki durumların her birinde, sağlanan değer listelenen geçerli değerlerden biri değilse, varsayılan değer alınır. Geçersiz bir değer belirtildiğinden söz edilen bir hata iletisi yazılmadı.

CDPCheckExtensions=YES|NO

CDPCheckExtensions , bu kuyruk yöneticisinde SSL ya da TLS kanallarının, CrlDistributionPoint sertifika uzantılarında adı geçen CDP sunucularını denetleyip denetmeyeceğini belirtir.

- YES: SSL ya da TLS kanalları, bir dijital sertifikana geri verilip verilmediğini belirlemek için CDP sunucularını denetleyip denetmeye çalışır.
- NO: SSL ya da TLS kanalları, CDP sunucularını denetlemeyi denemez. Bu, varsayılan değerdir.

OCSPAAuthentication=REQUIRED|WARN|OPTIONAL

OCSPAAuthentication, bir OCSP sunucusundan iptal durumu saptanamadığında yapılacak işlemi belirtir.

OCSP denetimi etkinleştirilmişse, bir SSL ya da TLS kanalı programı bir OCSP sunucusuyla bağlantı kurma girişiminde bulunur.

Kanal programı herhangi bir OCSP sunucusuyla iletişim kuramıyorsa ya da hiçbir sunucu sertifikana ilişkin iptal durumunu sunamazsa, OCSPAAuthentication parametresinin değeri kullanılır.

- REQUIREMS: İptal durumunun saptanması, bağlantının bir hata ile kapatılmasına neden olur. Bu, varsayılan değerdir.
- WARN: İptal durumunun saptanması, bir uyarı iletisinin kuyruk yöneticisi hata günlüğüne yazılmasına neden olur, ancak bağlantının devam etmesi için izin verilir.
- İSTEĞE BAĞLI: İptal durumunun saptanması, bağlantının sessiz bir şekilde devam etmesine olanak sağlar. Uyarı ya da hata verilmedi.

OCSPCheckExtensions=YES|NO

OCSPCheckExtensions , bu kuyruk yöneticisinde SSL ve TLS kanallarının, AuthorityInfoerişim sertifikası uzantılarında adı geçen OCSP sunucularını denetleyip denetmeyeceğini belirtir.

- YES: SSL ve TLS kanalları, bir dijital sertifikana geri verilip verilmediğini belirlemek için OCSP sunucularını denetlimeyi dener. Bu, varsayılan değerdir.
- NO(NO): SSL ve TLS kanalları, OCSP sunucularını denetleme girişiminde yoktur.

SSLHTTPProxyName=dizgi

Bu dizgi, GSKit for OCSP denetimlerini kullanarak kullanılacak HTTP yetkili sunucusunun anasistem adı ya da ağ adresidir. Bu adres, isteğe bağlı bir kapı numarası ve parantez içine alınmış olarak izlenebilir. Kapı numarasını belirtmezseniz, varsayılan HTTP kapısı 80 olarak kullanılır. HP-UX PA-RISC ve Sun Solaris SPARC altyapılarında ve AIX üzerindeki 32 bit istemciler için, ağ adresi yalnızca bir IPv4 adresi olabilir; diğer altyapılarda IPv4 ya da IPv6 adresi olabilir.

Örneğin, bir güvenlik duvarı OCSP yanıtlayıcıya erişimi engelliyorsa, bu öznetelik gerekli olabilir.

Çıkış özellikleri

Bir kuyruk yöneticisinde çıkış özelliklerine ilişkin bilgileri belirlemek için IBM WebSphere MQ Explorer'a da qm.ini dosyasındaki ExitPropertiesLocal stanza adlı küme kuyruk yöneticisi özellikleri sayfasını kullanın. Diğer bir seçenek olarak, bunu **amqmdain** komutunu kullanarak da ayarlayabilirsiniz.

Varsayılan olarak, bu ayar, makine genelindeki yapılandırmanın ExitProperties kısmında bulunan CLWLMode özneteliğinden devralınır ("Çıkış özellikleri" sayfa 418 içinde açıklanmıştır). Bu ayarı, yalnızca bu kuyruk yöneticisini farklı bir şekilde yapılandırmak istiyorsanız değiştirin. Bu değer, Küme kuyruk yöneticisi özellikler sayfasındaki küme iş yükü kipi özneteliği kullanılarak tek tek kuyruk yöneticileri için geçersiz kılınabilir.

CLWLMode=SAFE|CAST (CAST)

Küme iş yükü (CLWL) çıkışı, bir MQI çağrısına yanıt olarak hangi küme kuyruğunun açılacağı (örneğin, MQOPEN, MQPUT) hangi küme kuyruğunu açabildiğinizi belirtmenize olanak tanır. CLWL çıkışı, CLWLMode öznetisinde belirttiğiniz değere bağlı olarak, FAST ya da SAFE kipinde çalıştırılır. CLWLMode özneteliğini atlarsanız, küme iş yükü çıkışı SAFE kipinde çalıştırılır.

SAFE

CLWL çıkışını kuyruk yöneticisinden ayrı bir işlemde çalıştırın. Bu varsayılandır.

SAFE kipinde çalışırken kullanıcı tarafından yazılan CLWL çıkışıyla ilgili bir sorun ortaya çıksa, aşağıdakiler gerçekleşir:

- CLWL sunucu işlemi (amqzlw0) başarısız olur.
- Kuyruk yöneticisi, CLWL sunucu işlemini yeniden başlatır.
- Hata, hata günlüğüne raporlanır. Bir MQI çağrısı devam ederse, bildirim formunu bir dönüş kodu biçiminde alırsınız.

Kuyruk yöneticisinin bütünlüğü korunur.

Not: CLWL çıkışının ayrı bir süreçte çalıştırılması başarıyı etkileyebilir.

Hızlı

Kuyruk yöneticisi işleminde küme çıkışını iç olarak çalıştırın.

Bu seçeneğin belirlenmesi, SAFE kipindeki çalışmayla ilişkili süreç değiştirme maliyetlerinden kaçınarak performansı iyileştirir, ancak bunu kuyruk yöneticisi bütünlüğü pahasına yapar. You should only run the CLWL exit in FAST mode if you are convinced that there are **hayır** problems with your CLWL exit, and you are particularly concerned about performance.

CLWL çıkışı FAST kipinde çalışırken bir sorun ortaya çıkmışsa, kuyruk yöneticisi başarısız olur ve kuyruk yöneticisinin bütünlüğünün tehlikeye atılma riskini siz çalıştırıyorsunuz demektir.

Alt Havuz

Bu stanza, WebSphere MQ tarafından oluşturulur. Bunu değiştirmeyin.

The stanza Subpool, and attribute ShortSubpoolName within that stanza, are written automatically by WebSphere MQ when you create a queue manager. WebSphere MQ , ShortSubpooladı için bir değer seçer. Bu değeri değiştirmeyin.

The name corresponds to a directory and symbolic link created inside the /var/mqm/sockets directory, which WebSphere MQ uses for internal communications between its running processes.

yapılandırmaHP Integrity NonStop Server

HP Integrity NonStop Server kuruluđu için IBM WebSphere MQ istemcinizi yapılandırmanıza yardımcı olması için bu bilgileri kullanın.

Bir istemciyi yapılandırma dosyası kullanarak yapılandırma ile ilgili ayrıntılar için bkz. [“Yapılandırma dosyası kullanarak istemci yapılandırılması” sayfa 124.](#)

Bir istemcinin ortam deęişkenlerini kullanarak yapılandırılmasına ilişkin ayrıntılar için bkz. [“WebSphere MQ ortam deęişkenlerinin kullanılması” sayfa 142.](#)

TMF/Gateway altında HP Integrity NonStop Server işlemleri için IBM WebSphere MQ Client 'ı gerçekleştiriyorsanız, TMF/Gateway 'in nasıl yapılandırılacağı hakkında bilgi için alt başlıklara bakın. Gateway işlemine genel bir bakış eklenmiştir; Yol Geçidi altında çalışacak Ağ Geçidi 'ni yapılandıran ve istemci kullanıma hazırlama kütüğünü yapılandırmanın, HP Integrity NonStop Server için IBM WebSphere MQ istemcinizin TMF Ağ Geçidine ulaşmasını sağlayacak şekilde yapılandırılması.

This section also contains IBM WebSphere MQ client for HP Integrity NonStop Server specific information about granting permissions to channels.

Gateway sürecine genel bakış

HP NonStop Transaction Management Facility (TMF), bir ağ geçidi işleminin kaynak yöneticisi olarak kaydolması için hizmetler sağlar. The IBM WebSphere MQ provided TMF/Gateway process runs under Pathway.

IBM WebSphere MQ , TMF tarafından koordine edilen her kuyruk yöneticisi için tek bir ağ geçidi işlemi kaydeder; bu nedenle, eşgüdümlü iş birimlerine katılacak her kuyruk yöneticisi için ayrı bir TMF/Ağ Geçidi yapılandırmanız gerekir. Bu kayıt, her bir kuyruk yöneticisinin bağımsız bir kaynak yöneticisi olması ve her bir kuyruk yöneticisini HP NonStop TMF ile bir kez kaydettirmek, kolay anlaşılır bir eşlemeyle sonuçlansa da, bu kayıt yöneticisi tarafından yönetimle ilgili bir kayıt olarak kabul edilir.

Birden çok IBM WebSphere MQ kuruluşu için, her kuyruk yöneticisi için, TMF tarafından eşgüdümlü olması için, bu kuruluşlardan birinden tek bir ağ geçidi işlemi göstermeniz gerekir.

Ağ geçidi işlemine ilişkin arabirim, aynı sürümde ya da daha önceki sürümlerde herhangi bir istemciyi destekler.

Ağ geçidi sürecini yönetme hakkında daha fazla bilgi için bkz. [HP Integrity NonStop Server Yönetimi.](#)

Ağ geçidi, Yol altında çalışacak şekilde yapılandırılıyor

TMF/Gateway, TMF 'nin IBM WebSphere MQ işlemleri için hareket koordinatörü olmasını sağlayan HP NonStop Transaction Management Facility (TMF) ve IBM WebSphere MQ arasındaki arabirimdir.

IBM WebSphere MQ tarafından sağlanan TMF/Gateway, uzak kuyruk yöneticisiyle iletişim kurmak için TMF koordinasyonundan eXtended Architecture (XA) hareket koordinasyonuna yönelik hareketleri dönüştürür.

Eşgüdümü gerektiren bir kuyruk yöneticisi başına bir TMF/Gateway gereklidir ve istemcinin doğru Gateway 'e bağlanabilmesi için istemci yapılandırması gereklidir.

TMF/Gateway, bir kuyruk yöneticisiyle iletişim kurmak için istemciye sağlanan tüm mekanizmaları kullanabilir. TMF/Gateway ürününü, diğer uygulamalarınız için istediğiniz şekilde yapılandırın.

TMF/Gateway bir HP Integrity NonStop Server işlem çifti değildir ve Pathway ortamında çalışmak üzere tasarlanmıştır. TMF/Gateway, sonraki çalıştırmalarda yeniden kullandığı TMF içinde kalıcı kaynaklar yaratır; bu nedenle, TMF/Gateway her zaman aynı kullanıcı yetkisi altında çalıştırılmalıdır.

Sunucu sınıfının tanımlanması

TMF/Gateway, Pathway ortamı içinde bir serverclass olarak barındırılır. Serverclass 'ı tanımlamak için aşağıdaki sunucu özniteliklerini ayarlamamız gerekir:

PROCESSTYPE=OSS

Sunucu sınıfındaki sunucuların tipini belirtir. Ağ geçidi işlemi, çok iş parçacıklı bir OSS programıdır. Bu öznitelik zorunludur ve OSSolarak ayarlanmalıdır.

MAXSERVERS=1

Bu serverclass 'ta aynı anda çalışabilecek sunucu işlemi sayısı üst sınırını belirtir. Herhangi bir kuyruk yöneticisi için yalnızca tek bir Gateway işlemi olabilir. Bu öznitelik zorunludur ve 1olarak ayarlanmalıdır.

NUMSTATIC=1

Bu serverclass içindeki statik sunucu sayısı üst sınırını belirtir. Gateway işlemi bir statik sunucu olarak çalıştırılmalıdır. Bu öznitelik zorunludur ve 1olarak ayarlanmalıdır.

TMF=ON

Bu sunucu sınıfındaki sunucuların, TMF alt sistemi tarafından denetlenen veri dosyalarını kilitleyip güncelleyemeyeceğini belirler. Gateway işlemi, IBM WebSphere MQ istemci uygulamalarının TMF hareketlerine katılır; bu nedenle bu öznitelik ONolarak ayarlanmalıdır.

PROGRAM=<OSS installation path>/opt/mqm/bin/runmqtmf

IBM WebSphere MQ için IBM WebSphere MQ istemcisi için bu öznitelik runmqtmf olmalıdır. Bu öznitelik, mutlak OSS yol adı olmalıdır. Dava önemli.

ARGLIST=-m <QMGr adı >[,-c <kanal adı>] [-p <kapı>] [-h <anasistem adı>] [-n <üst Sınır threads>]

Bu öznitelikler, Gateway işlemine parametreler sağlar. Burada:

- QMGrName , bu Gateway işlemi için kuyruk yöneticisinin adıdır. Bir kuyruk paylaşım grubu (ya da başka bir kapı dağıtım teknolojisi) kullanıyorsanız, bu parametrenin belirli bir kuyruk yöneticisine hedeflenmesi gerekir. Bu parametre zorunludur.
- kanal adı , Gateway işlemi tarafından kullanılacak kuyruk yöneticilerindeki sunucu kanalının adıdır. Bu parametre isteğe bağlıdır.
- kapı , kuyruk yöneticisine ilişkin TCP/IP kapısıdır. Bu parametre isteğe bağlıdır.
- anasistem adı , kuyruk yöneticisine ilişkin anasistem adıdır. Bu parametre isteğe bağlıdır.
- iş parçacığı sayısı üst sınırı , Gateway işlemi tarafından yaratılan işçi iş parçacığı sayısı üst sınırıdır. Bu parametre, 10 ya da daha büyük bir değer olabilir. 10 'dan küçük bir değer belirtilse bile, kullanılan en düşük değer 10' dur. Herhangi bir değer sağlanmazsa, Gateway işlemi en çok 50 iş parçacığı yaratır.

Gateway 'e bağlantı bilgilerini sağlamak için alternatif bir yöntem olarak -c, -pve -h özniteliklerini "Ortam değişkenleri kullanılarak TMF/Gateway 'in yapılandırılması" sayfa 441' de tanımlanana ek olarak kullanın. Bir ya da daha çok -c, -pve -h özniteliklerini belirtmezseniz, varsayılan olarak belirtmediğiniz öznitelikler aşağıdaki değerlere göre belirlenir:

- kanal adı varsayılan olarak SYSTEM.DEF.SVRCONN değerine ayarlanır
- anasistem adı varsayılan değer olarak localhost değerine ayarlanır
- kapı varsayılan olarak 1414 değerine ayarlanır

Sağladığınız parametrelerden herhangi biri geçersizse, TMF/Gateway sorun tanılama iletisini AMQ5379 hata günlüğüne gönderir ve sonlanır.

OWNER=ID

Gateway 'in çalıştığı ve kuyruk yöneticisine bağlanma yetkisi verilmesi gereken kullanıcı kimliği.

SECURITY="value"

Specifies the users, in relation to the Owner attribute, who can access the Gateway from a IBM WebSphere MQ client application.

LINKDEPTH and MAXLINKS must be configured with values appropriate for the expected number of IBM WebSphere MQ client applications that might want to concurrently communicate with the Gateway. If these values are set too low, you might see occurrences of the error message [AMQ5399](#) issued from client applications.

Bu sunucu özniteliklerine ilişkin ek bilgi için *HP NonStop TS/MP 2.5 System Management Manual* adlı kılavuza bakın.

Ortam değişkenleri kullanılarak TMF/Gateway 'in yapılandırılması

TMF/Gateway 'i tanımlamak için en yaygın kullanılan yöntemlerden biri, MQSERVER ortam değişkenini ayarlamadır; örneğin:

```
ENV MQSERVER=<channel name>/<transport>/<host name>(<listener port>)
```

Komutun başlangıcındaki ENV, Pathway gösterimine sahip olur.

İstemci kullanıma hazırlama kütüğünün yapılandırılması

If you are using the HP NonStop Transaction Management Facility (TMF), you must have an IBM WebSphere MQ client initialization file to enable your IBM WebSphere MQ client for the HP Integrity NonStop Server to reach the TMF Gateway.

HP Integrity NonStop Server için IBM WebSphere MQ istemci kullanıma hazırlama kütüğü bir yerde tutulabilir; daha fazla bilgi için bkz. [“İstemci yapılandırma dosyasının konumu”](#) sayfa 125.

Yapılandırma dosyasının içeriğine bir örnekle birlikte ilişkin ayrıntılar için bkz. [“Yapılandırma dosyası kullanarak istemci yapılandırılması”](#) sayfa 124. TMF kuyruk yöneticisini ve sunucu ayrıntılarını belirtmek için TMF 'yi kullanın. Ek bilgi için [“TMF ve TMF/Gateway stanzaları”](#) sayfa 141' e bakın.

HP Integrity NonStop Server için IBM WebSphere MQ istemcisine ilişkin girişlere bir örnek:

```
TMF :
  PathMon=$PSD1P

TmfGateway:
  QManager=MQ5B
  Server=MQ-MQ5B

TmfGateway:
  QManager=MQ5C
  Server=MQ-MQ5C
```

Ortam değişkenlerini kullanarak bir istemcinin yapılandırılmasıyla ilgili daha fazla bilgi için bkz. [“WebSphere MQ ortam değişkenlerinin kullanılması”](#) sayfa 142.

Kanallara izin verilmesi

HP Integrity NonStop Server için IBM WebSphere MQ istemcisine yönelik kanallara izin verilmesi, diğer işletim sistemleriyle aynıdır, ancak ağ geçidinin altında çalıştığı iyenin kimliğini bilmeniz gerekir.

Daha sonra uygun izinleri vermek için ağ geçidinin sahibinin kimliğini kullanabilirsiniz. Önemli fark, kuyruk yöneticisi kanallarına izin verilmesinin herhangi bir uygulama yetkisi altında olmamasıdır.

Bir yetki vermek, bir yetki almak, bir IBM WebSphere MQ birincil kullanıcı ya da kullanıcı grubu izni vermek ve bir yetkiyi iptal etmek için [setmqaut](#) komutunu kullanın; yani, bir işlem gerçekleştirmek için izni kaldırın.

Özel notlar

Bu belge, ABD'de kullanıma sunulan ürünler ve hizmetler için hazırlanmıştır.

IBM, bu belgede sözü edilen ürün, hizmet ya da özellikleri diğer ülkelerde kullanıma sunmayabilir. Bulduğunuz yerde kullanıma sunulan ürün ve hizmetleri yerel IBM müşteri temsilcisinden ya da çözüm ortağınızdan öğrenebilirsiniz. Bir IBM ürün, program ya da hizmetine gönderme yapılması, açık ya da örtük olarak yalnızca o IBM ürünü, programı ya da hizmetinin kullanılabilirliğini göstermez. Aynı işlevi gören ve IBM'in fikri mülkiyet haklarına zarar vermeyen herhangi bir ürün, program ya da hizmet de kullanılabilir. Ancak, IBM dışı ürün, program ya da hizmetlerle gerçekleştirilen işlemlerin değerlendirilmesi ve doğrulanması kullanıcının sorumluluğundadır.

IBM'in, bu belgedeki konularla ilgili patentleri ya da patent başvuruları olabilir. Bu belgenin size verilmiş olması, patentlerin izinsiz kullanım hakkının da verildiği anlamına gelmez. Lisansla ilgili sorularınızı aşağıdaki adrese yazabilirsiniz:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Çift byte (DBCS) bilgilerle ilgili lisans soruları için, ülkenizdeki IBM'in Fikri Haklar (Intellectual Property) bölümüyle bağlantı kurun ya da sorularınızı aşağıda adrese yazın:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japonya

Aşağıdaki paragraf, İngiltere ya da bu tür hükümlerin yerel yasalarla uyumadığı diğer ülkelerde geçerli değildir: INTERNATIONAL BUSINESS MACHINES CORPORATION BU YAYINI, HAK İHLALİ YAPILMAYACAĞINA DAİR GARANTİLERLE TİCARİLİK VEYA BELİRLİ BİR AMACA UYGUNLUK İÇİN ZİMNİ GARANTİLER DE DAHİL OLMAK VE FAKS BUNLARLA SINIRLI OLMAMAK ÜZERE AÇIK YA DA ZİMNİ HİÇBİR GARANTİ VERMEKSİZİN "OLDUĞU GİBİ" ESASIYLA SAĞLAMAKTADIR. Bazı ülkeler bazı işlemlerde garantinin açık ya da örtük olarak reddedilmesine izin vermez; dolayısıyla, bu bildirim sizin için geçerli olmayabilir.

Bu yayın teknik yanlışlar ya da yazım hataları içerebilir. Buradaki bilgiler üzerinde düzenli olarak değişiklik yapılmaktadır; söz konusu değişiklikler sonraki basımlara yansıtılacaktır. IBM, önceden bildirimde bulunmaksızın, bu yayında açıklanan ürünler ve/ya da programlar üzerinde iyileştirmeler ve/ya da değişiklikler yapabilir.

Bu belgede IBM dışı Web sitelerine yapılan göndermeler kullanıcıya kolaylık sağlamak içindir ve bu Web sitelerinin onaylanması anlamına gelmez. Bu Web sitelerinin içerdiği malzeme, bu IBM ürününe ilişkin malzemenin bir parçası değildir ve bu tür Web sitelerinin kullanılmasının sorumluluğu size aittir.

IBM'e bilgi ilettiğinizde, IBM bu bilgileri size karşı hiçbir yükümlülük almaksızın uygun gördüğü yöntemlerle kullanabilir ya da dağıtabilir.

(i) Bağımsız olarak yaratılan programlarla, bu program da içinde olmak üzere diğer programlar arasında bilgi değiş tokuşuna ve (ii) değiş tokuş edilen bilginin karşılıklı kullanımına olanak sağlamak amacıyla bu program hakkında bilgi sahibi olmak isteyen lisans sahipleri şu adrese yazabilirler:

IBM Corporation
Yazılım Birlikte Çalışabilirlik Koordinatörü, Bölüm 49XA
3605 Highway 52 N

Rochester, MN 55901
U.S.A.

Bu tür bilgiler, ilgili kayıt ve koşullar altında ve bazı durumlarda bedelli olarak edinilebilir.

Bu belgede açıklanan lisanslı program ve bu programla birlikte kullanılacak tüm lisanslı malzeme, IBM tarafından, IBM Müşteri Sözleşmesi, IBM Uluslararası Program Lisansı Sözleşmesi ya da eşdeğer herhangi bir sözleşmenin kayıt ve koşulları altında sağlanır.

Burada belirtilen performans verileri denetimli bir ortamda elde edilmiştir. Bu nedenle, başka işletim ortamlarında çok farklı sonuçlar alınabilir. Bazı ölçümler geliştirilme düzeyindeki sistemlerde yapılmıştır ve bu ölçümlerin genel kullanıma sunulan sistemlerde de aynı olacağı garanti edilemez. Ayrıca, bazı sonuçlar öngörü yöntemiyle elde edilmiş olabilir. Dolayısıyla, gerçek sonuçlar farklı olabilir. Bu belgenin kullanıcıları, kendi ortamları için geçerli verileri kendileri doğrulamalıdır.

IBM dışı ürünlerle ilgili bilgiler, bu ürünleri sağlayan firmalardan, bu firmaların yayın ve belgelerinden ve genel kullanıma açık diğer kaynaklardan alınmıştır. IBM bu ürünleri sınınamamıştır ve IBM dışı ürünlerle ilgili performans doğruluğu, uyumluluk gibi iddiaları doğrulayamaz. IBM dışı ürünlerin yeteneklerine ilişkin sorular, bu ürünleri sağlayan firmalara yöneltilmelidir.

IBM'in gelecekteki yönelim ve kararlarına ilişkin tüm bildirimler değişebilir ve herhangi bir duyuruda bulunulmadan bunlardan vazgeçilebilir; bu yönelim ve kararlar yalnızca amaç ve hedefleri gösterir.

Bu belge, günlük iş ortamında kullanılan veri ve raporlara ilişkin örnekler içerir. Örneklerin olabildiğince açıklayıcı olması amacıyla kişi, şirket, marka ve ürün adları belirtilmiş olabilir. Bu adların tümü gerçek dışıdır ve gerçek iş ortamında kullanılan ad ve adreslerle olabilecek herhangi bir benzerlik tümüyle rastlantıdır.

YAYIN HAKKI LİSANSI:

Bu belge, çeşitli işletim platformlarında programlama tekniklerini gösteren, kaynak dilde yazılmış örnek uygulama programları içerir. Bu örnek programları, IBM'e herhangi bir ödemede bulunmadan, örnek programların yazıldığı işletim altyapısına ilişkin uygulama programlama arabirimiyle uyumlu uygulama programlarının geliştirilmesi, kullanılması, pazarlanması ya da dağıtılması amacıyla herhangi bir biçimde kopyalayabilir, değiştirebilir ve dağıtabilirsiniz. Bu örnekler her koşul altında tüm ayrıntılarıyla sınınamamıştır. Dolayısıyla, IBM bu programların güvenilirliği, bakım yapılabilirliği ya da işlevleri konusunda açık ya da örtük güvence veremez.

Bu bilgileri elektronik kopya olarak görüntülediyseniz, fotoğraflar ve renkli resimler görünmeyebilir.

Programlama arabirimi bilgileri

Programlama arabirimi bilgileri (sağlandıysa), bu programla birlikte kullanılmak üzere uygulama yazılımları yaratmanıza yardımcı olmak üzere hazırlanmıştır.

Bu kitap, müşterinin IBM WebSphere MQ hizmetlerini edinmek üzere program yazmasına olanak tanıyan, amaçlanan programlama arabirimlerine ilişkin bilgiler içerir.

Ancak, bu bilgiler tanılama, değiştirme ve ayarlama bilgilerini de içerebilir. Tanılama, değiştirme ve ayarlama bilgileri, uygulama yazılımlarınızda hata ayıklamanıza yardımcı olur.

Önemli: Bu tanılama, değiştirme ve ayarlama bilgilerini bir programlama arabirimi olarak kullanmayın; bu, değişiklik söz konusu olduğunda kullanılır.

Ticari Markalar

IBM, IBM logosu, ibm.com, IBM Corporation 'ın dünya çapında birçok farklı hukuk düzeninde kayıtlı bulunan ticari markalarıdır. IBM ticari markalarının güncel bir listesini Web üzerinde "Telif hakkı ve ticari marka bilgileri" www.ibm.com/legal/copytrade.shtml adresinde bulabilirsiniz. Diğer ürün ve hizmet adları IBM'in veya diğer şirketlerin ticari markaları olabilir.

Microsoft ve Windows, Microsoft Corporation'ın ABD ve/veya diğer ülkelerdeki ticari markalarıdır.

UNIX, The Open Group şirketinin ABD ve diğer ülkelerdeki tescilli ticari markasıdır.

Linux, Linus Torvalds'ın ABD ve/ya da diđer ÷lkelerdeki tescilli ticari markasıdır.

Bu ÷r÷n, Eclipse Project (<http://www.eclipse.org/>) tarafından geliřtirilen yazılımları ierir.

Java ve Java tabanlı t÷m markalar ve logolar, Oracle firmasının ve/ya da iřtiraklerinin markaları ya da tescilli markalarıdır.



Parça numarası:

(1P) P/N: