

9.1

*Kapsayıcılarda IBM MQ*

**IBM**

**Not**

Bu bilgileri ve desteklediđi ürünü kullanmadan önce, "[Özel notlar](#)" sayfa 51 bölümündeki bilgileri okuyun.

Bu basım, yeni basımlarında tersi belirtilmediđi sürece, IBM® MQ sürüm 9 yayın düzeyi 1 'i ve sonraki tüm yayın düzeyleri ve deđişiklikler 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

<b>Taşıyıcılarda IBM MQ</b> .....	<b>5</b>
Taşıyıcılarda IBM MQ için planlama.....	5
Kapsayıcılarda IBM MQ 'ı nasıl kullanmak istediğinizi seçme.....	5
Support for IBM MQ certified containers.....	6
Kendi IBM MQ kapsayıcı resimlerinizi ve grafiklerinizi oluşturmak için destek.....	8
IBM MQ Advanced certified container ile ilgili depolama konuları.....	9
High availability for IBM MQ Advanced certified container.....	10
IBM MQ Advanced certified container için kullanıcı kimlik doğrulaması ve yetkilendirmesi.....	12
Installing and uninstalling the IBM MQ Operator on OpenShift.....	12
IBM MQ işletmeninin OpenShift web konsolunu kullanarak kurulması.....	12
IBM MQ İşletmeni OpenShift CLI kullanılarak kurma.....	14
IBM MQ sertifikalı taşıyıcıların konuşlandırılması.....	16
Preparing your OpenShift project for IBM MQ using the OpenShift CLI.....	16
Deploying a queue manager using the IBM Cloud Pak for Integration Platform Navigator.....	17
Deploying a queue manager using the OpenShift web console.....	18
Deploying a queue manager using the OpenShift CLI.....	18
IBM Cloud Pak for Integration Operations Dashboard ile bütünleştirme.....	20
OpenShift CLI kullanılarak özel MQSC ve INI dosyaları içeren bir resim oluşturma.....	21
IBM MQ sertifikalı taşıyıcıların Helm kullanılarak konuşlandırılması.....	23
Deploying previous CD releases of IBM MQ into an IBM Cloud Private cluster.....	26
IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Private kümesine eklenmesi.....	28
IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Kubernetes Service kümesine eklenmesi.....	29
OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma.....	29
OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması.....	31
Backing up and restoring queue manager configuration using the OpenShift CLI.....	32
Kendi IBM MQ taşıyıcınızı oluşturma.....	33
Bir taşıyıcıyı kullanarak kendi IBM MQ kuyruk yöneticisi görüntüsünün planlanması.....	33
Building a sample IBM MQ queue manager image using Docker.....	34
Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması.....	37
IBM MQ İşletmeni için API başvurusu.....	39
mq.ibm.com/v1beta1 için API başvurusu.....	39
<b>Özel notlar</b> .....	<b>51</b>
Programlama arabirimi bilgileri.....	52
Ticari Markalar.....	52



Kapsayıcılar, bir IBM MQ kuyruk yöneticisi ya da IBM MQ istemci uygulamasını, tüm bağımlılıkları ile yazılım geliştirme için standartlaştırılmış bir birime paketlenizi sağlar.

You can run IBM MQ in the pre-packaged container provided in IBM MQ Advanced and IBM MQ Advanced for Developers. Bu IBM MQ Advanced certified container , desteklenen bir görüntü ve Helm grafiği sunar ve üretimde hazır IBM MQ görüntüsünü Red Hat® OpenShift®, IBM Cloud Private'ye da IBM Cloud Kubernetes Service'ine yerleştirmek için kullanılabilir.

IBM MQ ' i bir IBM Cloud Pak for Integration kapsayıcısında ya da kendi oluşturduğunuz bir taşıyıcıda çalıştırabilirsiniz.

MQ Adv.

CD

IBM MQ Advanced certified container ile ilgili daha fazla bilgi için aşağıdaki bağlantılara bakın.

When planning for IBM MQ in containers, consider the support that IBM MQ provides for various architectural options, such as how high availability is managed, and how to secure your queue managers.

### Bu görev hakkında

Before you plan your IBM MQ in containers architecture, you should familiarize yourself with both the basic IBM MQ concepts (see [IBM MQ Teknik genel bakış](#)) as well as basic Kubernetes/OpenShift concepts (see [OpenShift Container Platform mimarisi](#)).

### Yordam

- “Kapsayıcılarda IBM MQ ' ı nasıl kullanmak istediğinizi seçme” sayfa 5.
- “High availability for IBM MQ Advanced certified container” sayfa 10.
- “IBM MQ Advanced certified container için kullanıcı kimlik doğrulaması ve yetkilendirmesi” sayfa 12.

### istediğinizi seçme

IBM MQ ' un kapsayıcılarda kullanılması için birden çok seçenek vardır: önceden paketlenmiş sertifikalı taşıyıcılar kullanmayı seçebilir ya da kendi resimlerinizi ve konuşlandırma kodunuzu oluşturabilirsiniz.

### IBM MQ Advanced sertifikalı taşıyıcıların kullanılması

Red Hat OpenShift Container Platform' ta konuşlandırmayı planlıyorsanız, onaylanmış kapsayıcıları kullanmak isteyebilirsiniz. Sertifikalı konteyner çeşidi vardır:

- IBM Cloud Pak for Integration için IBM MQ Advanced certified container . Bu, onaylı bir taşıyıcının sürümünü içeren ayrı bir IBM ürünüdür.
- IBM MQ Advanced certified container
- IBM MQ Advanced for Developers sertifikalı taşıyıcı (garanti verilmeyen)

IBM MQ 9.1.4 ve daha önceki CD yayınları da IBM Cloud Private ve IBM Cloud Kubernetes Service üzerinde desteklenmiştir.

Sertifikalı kapların hızla geliştiğini ve bu nedenle yalnızca [Continuous Delivery](#) yayın düzeylerinin altında desteklendiğini unutmayın.

Onaylı taşıyıcılar, önceden oluşturulmuş taşıyıcı görüntülerinin yanı sıra, Red Hat OpenShift Container Platform'ünde çalışmaya ilişkin konuşlandırma kodunu da içerir. From IBM MQ 9.1.5 onwards, queue managers are managed using an IBM MQ Operator. Prior versions of IBM MQ, up to and including version 9.1.5, are managed using Helm charts.

Bazı IBM MQ özellikleri, sertifikalı kapsayıcılar kullanılırken desteklenmez. Aşağıdakilerden herhangi birini yapmak istiyorsanız, kendi resimlerinizi ve grafiklerinizi oluşturmanız gerekir:

- Denetim ya da ileti alışverişi için REST API ' larını kullan
- Aşağıdaki MQ bileşenlerinden herhangi birini kullanın:
  - Managed File Transfer araçları ve kaynakları. Ancak, bir ya da daha fazla Coordination, Command ya da Agent kuyruk yöneticisi sağlamak için sertifikalı taşıyıcıları kullanabilirsiniz.
  - AMQP
  - IBM MQ Bridge to Salesforce
  - IBM MQ Bridge to blockchain (taşıyıcılarda desteklenmez)
- Helm grafiklerini kullanarak konuşlandırma yaparken web sunucusunu kullanın ( IBM Cloud Pak for Integration dışında)
- Kurtarma günlüklerini yapılandırma gibi, **crtmqm**, **strmqm** ve **endmqm** ile kullanılan seçenekleri özelleştirin

## Kendi resimlerinizi ve grafiklerinizi oluşturma

Bu, en esnek taşıyıcı çözümdür, ancak kapsayıcıların yapılandırılmasına ilişkin güçlü becerilere ve sonuçta elde edilen kapsayıcıya "Kendi" ' e gereksinim duymanızı gerektirir. Red Hat OpenShift Container Platform' u kullanmayı planlıyorsanız, kendi görüntülerinizi ve konuşlandırma kodunuzu oluşturmanız gerekir.

Kendi resimlerinizi oluşturmak için örnekler kullanılabilir. Bkz. [“Kendi IBM MQ taşıyıcınızı oluşturma” sayfa 33](#). Sertifikalı kapların bir parçası olarak sağlanan Helm grafikleri GitHub' da yayınlanır ve kendi görüntülerinizi oluştururken örnek olarak kullanılabilir:

- [IBM MQ Advanced certified container için Helm grafiği](#)
- [IBM MQ Advanced for Developers sertifikalı konteyner için Helm grafiği](#)

## İlgili kavramlar

[“Support for IBM MQ certified containers” sayfa 6](#)

IBM MQ onaylı kapsayıcılar yalnızca belirli Kubernetes ortamlarında desteklenir


[“Kendi IBM MQ kapsayıcı resimlerinizi ve grafiklerinizi oluşturmak için destek” sayfa 8](#)

Bir Linux sistemindeki kapsayıcıları kullanıyorsanız dikkate alınacak bilgiler.

Linux

## Support for IBM MQ certified containers

IBM MQ onaylı kapsayıcılar yalnızca belirli Kubernetes ortamlarında desteklenir

 CD yayın düzeyi V9.1.4 ve sonraki yayın düzeylerinde, IBM MQ Advanced certified container , Red Hat OpenShift ile birlikte kullanılmak üzere desteklenir. Bkz. [“Deploying a queue manager using the Helm CLI” sayfa 24](#).

V9.1.4 ' dan önceki CD yayınları, aşağıdaki Kubernetes ortamlarında destekleniyordu:

- IBM Cloud Kubernetes Service
- IBM Cloud Private
- IBM Cloud Private şu ürünü geçir Red Hat OpenShift

Desteklenen Kubernetes sürümleri için, karşıdan yüklenen bir IBM MQ Advanced Helm grafiğindeki `qualification.yaml` ve `Chart.yaml` dosyalarına bakın. Bu sürümler serbest bırakılacak yayın düzeyine göre değişir.

IBM MQ Advanced certified container yalnızca IBM MQ Operator kullanılarak konuşlandırıldığında ya da aşağıdaki Helm grafiklerinden birini kullanırken desteklenir:

- `ibm-mqadvanced-server-prod`
- IBM Cloud Pak for Integration içinde `ibm-mqadvanced-server-integration-prod`

**Not:** IBM MQ İşletmeni 'nin yayınından sonra Helm grafiklerinin kullanılması önerilmemektedir.

Taşıyıcı teknolojisi hızla geliştiği için, IBM MQ Advanced certified container yalnızca bu grafiğin yayın sırasında desteklediği platformların en son sürümünde desteklenir. If you want to use an older platform version, then you might need to use an older version of the IBM MQ Advanced certified container.

IBM MQ Advanced certified container görüntüsü, IBM MQ Continuous Delivery (CD) yayın düzeylerini temel alır. Bunlar, hangisi daha uzun olursa olsun, bir yıla kadar ya da iki CD yayın düzeyi için desteklenir. Long Term Support releases of IBM MQ are not available as a certified container.

From IBM MQ Advanced certified container V4.0 onwards, the image provides an installation of IBM MQ on a Red Hat Universal Base Image (UBI), which includes key Linux libraries and utilities used by IBM MQ. UBI, Red Hat Enterprise Linux anasisteminde çalıştırıldığında Red Hat tarafından desteklenmektedir. IBM MQ Advanced certified container ' in önceki sürümleri desteklenmeyen bir Ubuntu temel görüntüsünü kullandı.

### İlgili kavramlar

“Kendi IBM MQ kapsayıcı resimlerinizi ve grafiklerinizi oluşturmak için destek” sayfa 8  
Bir Linux sistemindeki kapsayıcıları kullanıyorsanız dikkate alınacak bilgiler.

## Linux > MQ Adv. > CD IBM MQ Advanced certified container için sürüm desteği

IBM MQ Advanced certified container, IBM MQ, IBM Cloud Kubernetes Service, IBM Cloud Pak for Integration, ve IBM Cloud Private desteklenen sürümleri arasındaki eşlemeyi gösteren bir çizelge kümesi.

### IBM MQ İşleç

> V 9.1.5

IBM MQ İşletmeni, IBM Cloud Pak for Integration sürüm 2020.2' nin bir parçası olarak ya da bağımsız olarak IBM MQ sürüm 9.1.5 ve üzeri sürümleriyle kullanılmak üzere desteklenir.

IBM MQ Operator, Red Hat OpenShift Container Platform sürüm 4.4 ya da sonraki sürümlerde desteklenir.

### IBM MQ Advanced certified container > V 9.1.5 (Helm grafiği)-kullanımdan kaldırıldı

Helm grafiği `ibm-mqadvanced-server-prod` içerir.

> V 9.1.5 IBM MQ Advanced certified container V5.0.x sürümünden itibaren, Helm grafiği, görüntüsü ve düzeltmeler, IBM Yetkili Kataloğu ve Kayıt Dosyası aracılığıyla verilir. Önceki sürümler Passport Advantage aracılığıyla gönderildi ve düzeltme yayınları IBM Fix Central' den edinilebilir.

Çizelge 1. IBM MQ Advanced certified container desteği

S\u00fcr\u00fcm	IBM MQ sürümü	Destek sonu	Desteklenen platformlar
6.0.x	9.1.5 Sürekli Teslimat Serbest Bırakma	Mart 2021	<a href="#">Ayrıntılı sistem gereksinimleri</a>
5.0.x	9.1.4 Sürekli Teslimat Yayını	Aralık 2020	<a href="#">Ayrıntılı sistem gereksinimleri</a>
4.1.x	9.1.3 Sürekli Teslimat Serbest Bırakma	Temmuz 2020	<a href="#">Ayrıntılı sistem gereksinimleri</a>

## IBM Cloud Pak for Integration için IBM MQ Advanced certified container yazılımı **V9.1.5** (Helm grafiği)-kullanımdan kaldırıldı

Helm grafiği `ibm-mqadvanced-server-integration-prod` çıkar.

Çizelge 2. Version support for the IBM MQ Advanced certified container software for the IBM Cloud Pak for Integration

S\u00fcr\u00fcm	IBM MQ sürümü	IBM Cloud Pak for Integration sürümü
6.0.x	9.1.4 Sürekli Teslimat Yayını	2020.1.1 (Sistem Gereksinimleri)
5.0.x	9.1.3 Sürekli Teslimat Serbest Bırakma	2019.4.1 (Sistem Gereksinimleri)
4.1.x	9.1.3 Sürekli Teslimat Serbest Bırakma	2019.3.2.2 (Sistem Gereksinimleri)
4.0.x	9.1.3 Sürekli Teslimat Serbest Bırakma	2019.3.2 (Sistem Gereksinimleri)
3.0.x	9.1.3 Sürekli Teslimat Serbest Bırakma	2019.3.1 (Sistem Gereksinimleri)

Desteklenen sürüm bilgileri için [IBM Cloud Pak for Integration yayın notları](#) başlıklı konuya bakın.

## Linux Kendi IBM MQ kapsayıcı resimlerinizi ve grafiklerinizi oluşturmak için destek

Bir Linux sistemindeki kapsayıcıları kullanıyorsanız dikkate alınacak bilgiler.

- Kapsayıcı görüntü tarafından kullanılan temel görüntü, desteklenen bir Linux işletim sistemini kullanmalıdır.
- Ürünün taşıyıcı görüntüsünün içine kurmak için IBM MQ kuruluş programlarını kullanmanız gerekir.
- Desteklenen paketlerin bir listesi için bkz. [Linux sistemleri için IBM MQ rpm bileşenleri](#).

- **V9.1.0** Aşağıdaki paketler desteklenmiyor:

- MQSeriesBCBridge
- MQSeriesRDQM



- Kuyruk yöneticisi veri dizini (varsayılan değer olarak /var/mqm), kalıcı durumu alıkoyan bir taşıyıcı birimde saklanmalıdır.

**Önemli:** Birleşim dosya sistemini kullanamazsınız.

Bir anasistem dizinini veri birimi olarak bağlamanız ya da bir veri birimi kapsayıcısı kullanmanız gerekir. Ek bilgi için [Taşıyıcılardaki verileri yönetme](#) başlıklı konuya bakın.

- Kapsayıcı içinde **endmqm** gibi IBM MQ denetim komutlarını çalıştırabilmelisiniz.
- Tanılama amacıyla, taşıyıcının içinden dosya ve izin alabilmeniz gerekir.
- **V 9.1.0** Yerel olarak uygulamaları ayrı kapsayıcılarda çalışan bir kuyruk yöneticisinde yerel olarak bağlamak için, kuyruk yöneticisi için taşıyıcının ad alanlarını diğer taşıyıcılarla paylaşmak için namespace kullanabilirsiniz. Daha fazla bilgi için bkz [“Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması”](#) sayfa 37.

### İlgili kavramlar

“Support for IBM MQ certified containers” sayfa 6

IBM MQ onaylı kapsayıcılar yalnızca belirli Kubernetes ortamlarında desteklenir

Linux

MQ Adv.

V 9.1.5

CD

**IBM MQ Advanced certified**

## container ile ilgili depolama konuları

IBM MQ Advanced certified container iki depolama kipinde çalışır:

- **Ephemeral deposu**, taşıyıcı yeniden başlatıldığında, tüm taşıyıcı durumu atılabildiğinde kullanılır. Bu, gösterim için ortamlar yaratıldığında ya da bağımsız kuyruk yöneticileriyle geliştirildiğinde yaygın olarak kullanılır.
- **Kalıcı depolama**, IBM MQ için ortak yapılandırma ve taşıyıcı yeniden başlatılırsa, yeniden başlatılan kapsayıcıda var olan yapılandırmanın, günlüklerin ve kalıcı iletilerin kullanılabilir olduğunu doğrular.

IBM MQ işleci, ortama bağlı olarak farklılık gösterebilen depolama özelliklerini ve istenen depolama kipini özelleştirme yeteneği sağlar.

### Ephemeral deposu

IBM MQ, durumlu bir uygulamadır ve yeniden başlatma durumunda kurtarma için bu durumu depolamaya devam eder. Ephemeral deposu kullanılıyorsa, yeniden başlatma sırasında tüm kuyruk yöneticisi durumu kaybedilir. Bu bilgiler şunları içerir:

- Tüm İletiler
- Kuyruk yöneticisinin tüm kuyruk yöneticisi iletişim durumu (kanal ileti sıra numaraları)
- Kuyruk yöneticisinin MQ Kümesi kimliği
- Tüm hareket durumu
- Tüm kuyruk yöneticisi yapılandırması
- Tüm yerel tanımlama verileri

Bu nedenle, ephemeral depolamanın bir üretim, test veya geliştirme senaryosu için uygun bir yaklaşım olup olmadığını göz önünde bulundurmanız gerekir. Örneğin, tüm iletilerin kalıcı olmadığı bilindiği ve kuyruk yöneticisi bir MQ Cluster üyesi değil. Tüm ileti sistemi durumunu yeniden başlatmanın yanı sıra, kuyruk yöneticisi yapılandırması da atılır. To enable a completely ephemeral container the IBM MQ configuration must be added to the container image itself (for more information, see [“OpenShift CLI kullanılarak özel MQSC ve INI dosyaları içeren bir resim oluşturma”](#) sayfa 21). Bu işlem tamamlanmazsa, taşıyıcının her yeniden başlatıldığı her defasında IBM MQ 'in yapılandırılması gerekir.

For example, to configure IBM MQ with ephemeral storage the storage type of the QueueManager should include the following:

```
queueManager:  
  storage:
```

```
queueManager:
  type: ephemeral
```

## Kalıcı depolama

IBM MQ olağan durumda, kuyruk yöneticisinin yeniden başlatma işleminden sonra kalıcı iletilerini ve yapılandırmasını korumasını sağlamak için kalıcılık depolamasıyla çalışır. Bu nedenle, varsayılan davranış budur. Çeşitli depolama sağlayıcıları ve her bir destek için farklı yetenekler nedeniyle, bu genellikle yapılandırmanın özelleştirilmesinin gerekli olduğu anlamına gelir. Aşağıda, v1beta1 API 'sında MQ depolama yapılandırmasını özelleştiren ortak alanlar açıklanmaktadır:

- [spec.queueManager.availability](#) , kullanılabilirlik modunu denetler. SingleInstance kullanıyorsanız, yalnızca ReadWriteOnce depolama alanı gereklidir, multiInstance ise doğru dosya kilitleme özelliklerine sahip ReadWriteMany ' i destekleyen bir depolama sınıfı gerektirir. IBM MQ , bir [destek bildirim](#) ve bir [test bildirim](#) sağlar. Kullanılabilirlik kipi, kalıcı birim düzenini de etkiler. Daha fazla bilgi için bkz. “[High availability for IBM MQ Advanced certified container](#)” sayfa 10
- [spec.queueManager.storage](#) , tek tek depolama ayarlarını denetler. Bir kuyruk yöneticisi, bir ile dört kalıcı birim arasında kullanılmak üzere yapılandırılabilir

Aşağıdaki örnekte, tek yönetim ortamı kuyruk yöneticisi kullanan basit bir yapılandırmanın bir parçası gösterilmiştir:

```
spec:
  queueManager:
    storage:
      queueManager:
        enabled: true
```

Aşağıdaki örnek, varsayılan olmayan bir depolama sınıfından ve ek gruplar gerektiren dosya depolamasıyla, çok eşgörunümlü bir kuyruk yöneticisi yapılandırmasının parçacıklarını göstermektedir:

```
spec:
  queueManager:
    availability:
      type: MultiInstance
    storage:
      queueManager:
        enabled: true
        class: ibmc-file-gold-gid
        persistedData:
          enabled: true
          class: ibmc-file-gold-gid
        recoveryLogs:
          enabled: true
          class: ibmc-file-gold-gid
      securityContext:
        supplementalGroups: [99]
```

Linux

MQ Adv.

CD

## High availability for IBM MQ Advanced certified container

IBM MQ Advanced certified container ile yüksek kullanılabilirlik için iki ana seçeneğiniz vardır: **Çok eşgörunümlü kuyruk yöneticisi** (paylaşılan, ağ üzerinden çalışan bir dosya sistemi kullanan etkin bir yedekte bekleme çiftidir) ve **Tek bir esnek kuyruk yöneticisi** (ağ üzerinde çalışan depolamayı kullanan HA için basit bir yaklaşım sunar).

Ayrıca, ayrı olarak **ileti** ve **hizmet** kullanılabilirliğini göz önünde bulundurmanız gerekir. IBM MQ for Multiplatforms ile, bir ileti tam olarak bir kuyruk yöneticisiyle saklanır. Bu nedenle, kuyruk yöneticisi kullanılamaz duruma gelirse, geçici olarak tuttuğu iletilere erişimi geçici olarak kaybedersiniz. Yüksek **ileti** kullanılabilirliğine ulaşmak için, bir kuyruk yöneticisini mümkün olan en kısa sürede geri yükleyebilmeniz gerekir. You can achieve **hizmet** availability by having multiple instances of queues for client applications to use, for example by using an IBM MQ uniform cluster.

Kuyruk yöneticisi iki kısımda düşünülebilmektedir: diskte saklanan veriler ve verilere erişilmesine izin veren çalışan işlemler. Herhangi bir kuyruk yöneticisi farklı bir Kubernetes düğümüne taşınabilir

(Kubernetes Kalıcı Birimler tarafından sağlanan) aynı verileri alıkoyması ve istemci uygulamaları tarafından ağ üzerinde hala adreslenebilir olması koşuluyla. Kubernetes' ta, tutarlı bir ağ kimliği sağlamak için bir Hizmet kullanılır.

IBM MQ , kalıcı birimlerdeki verilerin kullanılabilirliğine dayanır. Bu nedenle, IBM MQ , kullanmakta olduğu depolama alanından daha fazla kullanılabilir olamayacağı için, kalıcı birimleri sağlayan depolamanın kullanılabilirliğini kuyruk yöneticisi kullanılabilirliği açısından kritik öneme sahip olur. Bir kullanılabilirlik bölgesinin kesintisine göz yummak istiyorsanız, disk yazmasını başka bir bölgeye kopyalayan bir birim sağlayıcısı kullanmanız gerekir.

## Çok eşgörünümli kuyruk yöneticisi

Çok eşgörünümli kuyruk yöneticileri, tam olarak iki eşleme ve bir Kubernetes Kalıcı Birim kümesi ile Kubernetes Stateful Set 'in bir parçası olarak çalışan bir **etkin** ve bir **beklemedeki** Kubernetes Pod 'unu içerir. Kuyruk yöneticisi hareket günlükleri ve verileri, paylaşılan bir dosya sistemi kullanılarak iki kalıcı birimde tutulur.

Çok eşgörünümli kuyruk yöneticileri hem **active** , hem de **standby** Pod 'larının kalıcı birime eşzamanlı erişime sahip olmasını gerektirir. To configure this, you use Kubernetes Persistent Volumes with **access mode** set to ReadWritePek çok. IBM MQ , dosya kilitlerinin bir kuyruk yöneticisini hata durumunda yedek sisteme geçişe teşvik etmek için otomatik olarak serbest bırakılmasına dayandığından, birimlerin IBM MQ paylaşılan kütük sistemlerine ilişkin gereksinmeleride karşılaması gerekir. IBM MQ , bir test edilen dosya sistemlerinin listesi oluşturur.

Çok eşgörünümli bir kuyruk yöneticisine ilişkin kurtarma süreleri aşağıdaki etkenler tarafından denetlenir:

1. Paylaşılan kütük sisteminin, etkin yönetim ortamı tarafından ilk olarak alınan kilitleri serbest bırakması için bir hata oluşuktan sonra ne kadar zaman alır.
2. Yedek yönetim ortamının kilitleri elde etmesi ve daha sonra başlatılması ne kadar sürer?
3. How long it takes for the Kubernetes Pod readiness probe to detect that the container is ready. Bu, Helm grafimesinde yapılandırılabilir.
4. IBM MQ istemcilerinin yeniden bağlanması ne kadar sürer?

## Tek bir esnek kuyruk yöneticisi

Tek bir esnek kuyruk yöneticisi, tek bir Kubernetes Pod 'unda çalışan bir kuyruk yöneticisinin tek bir eşgörünümüdür; burada Kubernetes kuyruk yöneticisini izler ve gerektiği şekilde Pod 'un yerini alır.

IBM MQ Paylaşılan dosya sistemlerine ilişkin gereksinimler , tek bir esnek kuyruk yöneticisi (kira temelli kilitleme dışında) kullanırken de uygulanır, ancak paylaşılan bir dosya sistemi kullanmanız gerekmez. Üst üste uygun bir dosya sistemi ile blok depolamayı kullanabilirsiniz. Örneğin, *xf*s ya da *ext4*.

Tek bir esnek kuyruk yöneticisine ilişkin kurtarma süreleri aşağıdaki etkenler tarafından denetlenir:

1. Hızlılık sondasının çalışması ne kadar zaman alır ve kaç tane başarısızlığa tahammül eder. Bu, Helm grafimesinde yapılandırılabilir.
2. Kubernetes Zamanlayıcısı, başarısız olan Pod 'ı yeniden yeni bir düğüme yeniden zamanlamak için ne kadar zaman alır.
3. Taşıyıcı görüntüsünü yeni düğüme karşıdan yüklemek ne kadar sürerse. IfNotPresent **imagePullPolicy** değerini kullanıyorsanız, görüntü o düğümden önceden kullanılabilir olabilir.
4. Yeni kuyruk yöneticisi yönetim ortamının başlatılması ne kadar sürer?
5. How long it takes for the Kubernetes Pod readiness probe to detect that the container is ready. Bu, Helm grafimesinde yapılandırılabilir.
6. IBM MQ istemcilerinin yeniden bağlanması ne kadar sürer?

### Önemli:

Tek bir esnek kuyruk yöneticisi örüntünün bazı yararlar sunmasına rağmen, düğüm hataları etrafındaki sınırlamalarla, uygunluk hedeflerinize erişip erişemediğinizi anlamanız gerekir.

Kubernetes' ta, arızalı bir Pod genellikle hızlı bir şekilde kurtarılır; ancak, tüm düğümün başarısız olması farklı bir şekilde işlenir. Kubernetes Ana Düğümü bir işçi düğümle bağlantı kaybederse, düğümün başarısız olup olmadığını ya da ağ bağlantılılığını kaybetmişse bu düğümün başarısız olup olmadığını saptayamaz. Bu nedenle, Kubernetes , aşağıdaki olaylardan biri gerçekleşinceye kadar bu durumda **işlem yok** seçeneğini alır:

1. Düğüm, Kubernetes Ana Düğümü ile iletişim kurabileceği bir duruma kurtarır.
2. An administrative action is taken to explicitly delete the Pod on the Kubernetes Master Node. Bu, Pod 'un çalışmasını durdurması gerekmez, ancak yalnızca Kubernetes mağazasından silinir. Bu nedenle, bu yönetimle ilgili işlem çok dikkatli bir şekilde alınmalıdır.

### İlgili kavramlar

[Yüksek kullanılabilirlik yapılandırmaları](#)

## Linux > MQ Adv. > CD **IBM MQ Advanced certified container için kullanıcı kimlik doğrulaması ve yetkilendirmesi**

IBM MQ , yetkilendirme için LDAP kullanıcılarını ve gruplarını kullanacak şekilde yapılandırılabilir. Bu, IBM MQ Advanced certified container için önerilen yaklaşıdır.

Red Hat OpenShift Container Platform gibi çok kiracılı bir kapsayıcıda, olası güvenlik sorunlarını önlemek için güvenlik kısıtlamaları yerine yerleştirilir. Örneğin, Red Hat OpenShift Container Platform ' ta varsayılan SecurityContextConstraints ( restricted olarak adlandırılır), rasgele bir kullanıcı kimliği kullanır ve taşıyıcının kendisini yerel olarak yerel bir kullanıcı kimliği olarak önerir. IBM MQ , genellikle kullanıcıların parolalarını denetlemek için ayrıcalık yükseltme özelliğini kullanır. Bu, çok kiracılı kapsayıcı ortamlarda da önerilmez. Bu nedenlerle, çalışan bir taşıyıcının içindeki işletim sistemi kitaplıklarında tanımlı olan kullanıcıların kullanımı IBM MQ sertifikalı taşıyıcılarda desteklenmez.

Kullanıcı kimlik doğrulaması ve yetkilendirmesi için LDAP ' yi kullanmak üzere kuyruk yöneticinizi yapılandırmanız gerekir. Bunu yapmak üzere IBM MQ ' in yapılandırılmasına ilişkin bilgi için bkz. [Bağlantı kimlik doğrulaması: Kullanıcı havuzları ve LDAP yetkilendirmesi](#)

## Linux > MQ Adv. > V 9.1.5 > CD **Installing and uninstalling the IBM MQ Operator on OpenShift**

IBM MQ Operator, Operator Hub ' ı kullanarak OpenShift ' e kurulabilir.

### Başlamadan önce

#### Yordam

- [“IBM MQ İşletmeni OpenShift CLI kullanılarak kurma” sayfa 14.](#)
- [“IBM MQ işletmeninin OpenShift web konsolunu kullanarak kurulması” sayfa 12.](#)

## Linux > MQ Adv. > V 9.1.5 > CD **IBM MQ işletmeninin OpenShift web konsolunu kullanarak kurulması**

IBM MQ Operator, Operator Hub ' ı kullanarak OpenShift ' e kurulabilir.

### Başlamadan önce

OpenShift kümesinin web konsolunuzda oturum açın.

#### Yordam

1. Kurulabilir işlemler listesine IBM Common Services işlemlerini ekleyin.
  - a) Artı simgesini tıklatın. **YAML İçe Aktar** iletişim kutusunu görürsünüz.

b) İletişim kutusuna aşağıdaki kaynak tanımlamasını yapıştırın.

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: opencloud-operators
  namespace: openshift-marketplace
spec:
  displayName: IBMCS Operators
  publisher: IBM
  sourceType: grpc
  image: docker.io/ibmcom/ibm-common-service-catalog:latest
  updateStrategy:
    registryPoll:
      interval: 45m
```

c) **Oluştur**'u tıklatın.

2. Kurulabilir işleçler listesine IBM işleçlerini ekleyin.

a) Artı simgesini tıklatın. **YAML İçer Aktar** iletişim kutusunu görürsünüz.

b) İletişim kutusuna aşağıdaki kaynak tanımlamasını yapıştırın.

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-operator-catalog
  namespace: openshift-marketplace
spec:
  displayName: ibm-operator-catalog
  publisher: IBM Content
  sourceType: grpc
  image: docker.io/ibmcom/ibm-operator-catalog
  updateStrategy:
    registryPoll:
      interval: 45m
```

c) **Oluştur**'u tıklatın.

3. IBM MQ İşletmeni için kullanılacak bir ad alanı oluşturun

IBM MQ işleci, kapsamlı bir tek ad alanına veya tüm ad alanlarına kurulabilir. Bu adım, önceden var olmayan belirli bir ad alanına kuruluş yapmak istiyorsanız gereklidir.

a) Gezinme bölmesinden, **Ana Sayfa > Projeler** 'i tıklatın.

Projects (Projeler) sayfası görüntülenir.

b) **Proje Oluştur** 'u tıklatın. Bir Proje Yarat alanı görüntülenir.

c) Yaratmakta olduğunuz ad alanının ayrıntılarını girin. Örneğin, ad olarak "ibm-mq" belirtebilirsiniz.

d) **Oluştur**'u tıklatın. IBM MQ İşleciniz için ad alanı oluşturulur.

4. IBM MQ işlecini kurun.

a) Gezinme bölmesinden, **İşleçler > OperatorHub**seçeneklerini tıklatın.

OperatorHub sayfası görüntülenir.

b) **Tüm Öğeler** alanında "IBM MQ" değerini girin.

IBM MQ katalog girişi görüntülenir.

c) **IBM MQ**seçeneğini belirleyin.

IBM MQ penceresi görüntülenir.

d) **Kur**'u tıklatın.

İşletmen Aboneliği Oluştur sayfasını görürsünüz.

e) Kuruluş Kipini oluşturduğunuz belirli bir ad alanına ya da küme geniş kapsamına ayarlayın

f) **Abone Oldüğmesini** tıklatın.

Kurulu İşleçler sayfasında IBM MQ 'u göreceksiniz.

g) Kurulu İşleçler sayfasındaki İşletmen 'in durumunu denetleyin, kuruluş tamamlandığında durum, başarılı olacak şekilde değişir.

## Sonraki adım

“IBM MQ sertifikalı taşıyıcıların konuşlandırılması” sayfa 16

Linux

MQ Adv.

V 9.1.5

CD

## IBM MQ İşletmeni OpenShift CLI

### kullanılarak kurma

IBM MQ İşletmen, İşletmen Göbeği kullanılarak OpenShift üzerine takılabilir.

### Başlamadan önce

**oc login** kullanarak OpenShift komut satırı arabiriminde (CLI) oturum açın. Bu adımlar için bir küme yöneticisi olmanız gerekir.

### Yordam

1. IBM Common Services işleçleri için bir OperatorSource oluşturma

a) OperatorSource kaynağını tanımlayan bir YAML dosyası oluşturun

Aşağıdaki içerikle "operator-source-cs.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: opencloud-operators
  namespace: openshift-marketplace
spec:
  displayName: IBMCS Operators
  publisher: IBM
  sourceType: grpc
  image: docker.io/ibmcom/ibm-common-service-catalog:latest
  updateStrategy:
    registryPoll:
      interval: 45m
```

b) OperatorSource uygulamasını sunucuya uygulayın.

```
oc apply -f operator-source-cs.yaml -n openshift-marketplace
```

2. IBM operatörleri için bir OperatorSource oluşturma

a) OperatorSource kaynağını tanımlayan bir YAML dosyası oluşturun

Aşağıdaki içerikle "operator-source-ibm.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1alpha1
kind: CatalogSource
metadata:
  name: ibm-operator-catalog
  namespace: openshift-marketplace
spec:
  displayName: ibm-operator-catalog
  publisher: IBM Content
  sourceType: grpc
  image: docker.io/ibmcom/ibm-operator-catalog
  updateStrategy:
    registryPoll:
      interval: 45m
```

b) OperatorSource uygulamasını sunucuya uygulayın.

```
oc apply -f operator-source-ibm.yaml -n openshift-marketplace
```

3. IBM MQ işleci için kullanılacak bir ad alanı oluşturun

IBM MQ işleci, tek bir ad alanı ya da tüm ad alanları kapsamında kurulabilir. Bu adım yalnızca önceden var olmayan belirli bir ad alanına kurmak istiyorsanız gereklidir.

```
oc new-project ibm-mq
```

#### 4. OperatorHub ' dan kümenin kullanabileceği Operatörlerin listesini görüntüleyin

```
oc get packagemanifests -n openshift-marketplace
```

#### 5. Desteklenen InstallModes ve kullanılabilir Kanalları doğrulamak için IBM MQ İşletmeni 'ni inceleyin.

```
oc describe packagemanifests ibm-mq -n openshift-marketplace
```

#### 6. Bir OperatorGroup nesnesi YAML dosyası oluşturma

OperatorGroup , OperatorGroup ile aynı ad alanındaki tüm Operatörler için gerekli RBAC erişiminin oluşturulacağı hedef ad alanlarını seçen bir OLM kaynağıdır.

İşleç 'e abone olduğunuz ad alanının, AllNamespaces ya da SingleNamespace kipinde İşleç InstallMode ile eşleşen bir OperatorGroup olmalıdır. Kurmak istediğiniz işleç AllNamespaces kullanıyorsa, openshift-operators ad alanında zaten uygun bir OperatorGroup vardır.

Ancak, İşletmen SingleNamespace kipini kullanıyorsa ve yerinde uygun bir OperatorGroup yoksa, bir tane oluşturmanız gerekir.

##### a) Aşağıdaki içerikle "mq-operator-group.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1
kind: OperatorGroup
metadata:
  name: <operatorgroup_name>
  namespace: <namespace>
spec:
  targetNamespaces:
  - <namespace>
```

##### b) OperatorGroup nesnesini oluşturma

```
oc apply -f mq-operator-group.yaml
```

#### 7. Bir ad alanına MQ işlecine abone olmak için bir Abonelik nesnesi YAML dosyası oluşturun

##### a) Aşağıdaki içerikle "mq-sub.yaml" adlı bir dosya oluşturun:

```
apiVersion: operators.coreos.com/v1alpha1
kind: Subscription
metadata:
  name: ibm-mq
  namespace: openshift-operators
spec:
  channel:
    name: ibm-mq
    source: ibm-operator-catalog
    sourceNamespace: openshift-marketplace
```

AllNamespaces **InstallMode** kullanımı için openshift-operators ad alanını belirtin. Aksi takdirde, SingleNamespace **InstallMode** kullanımı için ilgili tek ad alanını belirtin.

##### b) Subscription nesnesini oluşturma

```
oc apply -f mq-sub.yaml
```

#### 8. İşleç durumunu denetle

İşletmen kurulumu başarılı olduktan sonra, bölmenin durumu *Çalışıyor* olarak gösterilir.

AllNamespaces **InstallMode** kullanımı için ad alanı olarak **openshift-operators** belirtin. Aksi takdirde, SingleNamespace **InstallMode** kullanımı için ilgili tek ad alanını belirtin.

## Sonraki adım

["IBM MQ sertifikalı taşıyıcıların konuşlandırılması" sayfa 16](#)

## konuşlandırılması

IBM MQ sürüm 9.1.5 ve üstü, IBM MQ işlecini kullanarak Red Hat OpenShift ' e dağıtılabilir. IBM MQ sürümleri 9.1.5 ve 9.1.4 , Helm kullanılarak Red Hat OpenShift ' a konuşlandırılabilir. Önceki CD sürümleri, Helm kullanılarak bir IBM Cloud Private kümesine ya da bir IBM Cloud Kubernetes Service kümesine konuşlandırılabilir.

### Bu görev hakkında

#### Yordam

- “[Deploying a queue manager using the Helm CLI](#)” sayfa 24.
- “[Deploying previous CD releases of IBM MQ into an IBM Cloud Private cluster](#)” sayfa 26.
- “[IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Private kümesine eklenmesi](#)” sayfa 28.
- “[IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Kubernetes Service kümesine eklenmesi](#)” sayfa 29.

## using the OpenShift CLI

Prepare your Red Hat OpenShift Container Platform cluster, so that it's ready to deploy a queue manager using the IBM MQ Operator. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

### Başlamadan önce

**Not:** IBM MQ ' u başka IBM Cloud Pak for Integration bileşenleriyle birlikte bir projede kullanmayı planlıyorsanız, bu yönergeleri izlemenize gerek yoktur.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

### Bu görev hakkında

IBM MQ Advanced certified container görüntüleri, lisans hakkı denetimi gerçekleştiren bir kapsayıcı kayıt dosyasından çekiliyor. Bu denetim, bir `docker-registry` çekme sırasında saklanan bir yetki anahtarını gerektirir. Henüz bir yetki anahtarınız yoksa, bir yetki anahtarı almak ve bir çekme güvenlik dizgisi yaratmak için bu yönergeleri izleyin.

#### Yordam

1. Kimliğinize atanan yetki anahtarını alın.
  - a) Yetkili yazılımla ilişkili IBM kimliği ve parolasıyla [MyIBM Kapsayıcı Yazılım Kitaplığı](#) olanağında oturum açın.
  - b) In the **Yetki anahtarları** section, select **Kopyalama anahtarı** to copy the entitlement key to the clipboard.
2. Kuyruk yöneticinizi konuşlandırmak istediğiniz projede, yetki anahtarınızı içeren bir güvenlik dizgisi yaratın.

Şu komutu çalıştırın; burada `<entitlement-key>` , 1. adımda alınan anahtardır ve `<user-email>` , yetkili yazılımla ilişkilendirilen IBM kimliğidir.

```
oc create secret docker-registry ibm-entitlement-key \
--docker-server=cp.icr.io \
--docker-username=cp \
--docker-password=<entitlement-key> \
--docker-email=<user-email>
```



## Sonraki adım

[“Deploying a queue manager using the OpenShift CLI” sayfa 18](#)

Linux

MQ Adv.

V 9.1.5

CD

## Deploying a queue manager using the IBM Cloud Pak for Integration Platform Navigator

Bir kuyruk yöneticisini IBM Cloud Pak for Integration Platform Navigator kullanarak bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

### Başlamadan önce

Bir tarayıcıda IBM Cloud Pak for Integration Platform Navigator uygulamasını başlatın.

Bu, bir kuyruk yöneticisini bu Red Hat OpenShift projesinde ilk kez konuşlandırdıysa, [“Preparing your OpenShift project for IBM MQ using the OpenShift CLI” sayfa 16](#) için adımları izleyin.

### Yordam

1. Kuyruk yöneticisini konuşlandırın.

Aşağıdaki örnek, ephemeral (kalıcı olmayan) depolama kullanan ve MQ güvenliğini devre dışı döndüren bir "quick start" kuyruk yöneticisini konuşlandırır. Kuyruk yöneticisinin yeniden başlatılması sırasında iletiler kalıcı olarak saklanmaz. Birçok kuyruk yöneticisi ayarlarını değiştirmek için yapılandırma ayarlayabilirsiniz.

a) IBM Cloud Pak for Integration Platform Navigator'ta **Çalıştırma Zamanı ve Eşgörünümler**'i tıklayın.

b) **Eşgörünüm yarat** ögesini tıklayın.

c) **Kuyruk Yöneticisi** seçeneğini belirleyin ve **İleridüğmesini** tıklayın.

Bir QueueManager yönetim ortamı yaratmak için form görüntülenir.

**Not:** QueueManager yapılandırması YAML 'yi görüntülemek ya da değiştirmek için **Kod** 'u da tıklatabilirsiniz.

d) **Ayrıntılar** bölümünde, **Ad** alanını işaretleyin ya da güncelleştirin ve kuyruk yöneticisi örneğinin yaratılacağı **Ad alanı** 'yı belirtin.

e) IBM Cloud Pak for Integration lisans sözleşmesini kabul ederseniz, **Lisans kabulü** seçeneğini **Açık** olarak değiştirin.

Bir kuyruk yöneticisini konuşlandırmak için lisansı kabul etmeniz gerekir.

f) **Kuyruk Yöneticisi Yapılanışı** kısmında, temeldeki kuyruk yöneticisinin **Name** (Ad) ögesini denetleyin ya da güncelleyin.

By default, the name of the queue manager used by IBM MQ client applications will be the same as the name of the QueueManager, but with any invalid characters (such as hyphens) removed. Belirli bir adı kullanmayı zorlamak istiyorsanız, bunu burada düzenleyebilirsiniz.

g) **Yarat** düğmesini tıklayın.

Yürürlükteki projeden (ad alanı) kuyruk yöneticilerinin listesi görüntülenir. Yeni QueueManager, Pending durumunda olmalıdır.

2. Kuyruk yöneticisinin çalışır durumda olup olmadığını denetleyin

The creation is complete when the QueueManager status is Running.

### İlgili görevler

[“OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma” sayfa 29](#)

Bir Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

[“OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması” sayfa 31](#)

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

## OpenShift web console

Bir kuyruk yöneticisini Red Hat OpenShift Web konsolunu kullanarak bir Red Hat OpenShift Container Platform kümesine konuşturmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

### Başlamadan önce

OpenShift kümesinin web konsolunuzda oturum açın. Kullanmak için var olan bir Proje (ad alanı) seçmeniz ya da yeni bir proje yaratmanız gerekir.

Bu, bir kuyruk yöneticisini bu Red Hat OpenShift projesinde ilk kez konuşturduysa, [“Preparing your OpenShift project for IBM MQ using the OpenShift CLI”](#) sayfa 16 için adımları izleyin.

### Yordam

1. Kuyruk yöneticisini konuşturun.

Aşağıdaki örnek, ephemeral (kalıcı olmayan) depolama kullanan ve MQ güvenliğini devre dışı döndüren bir "quick start" kuyruk yöneticisini konuşturur. Kuyruk yöneticisinin yeniden başlatılması sırasında iletiler kalıcı olarak saklanmaz. Birçok kuyruk yöneticisi ayarlarını değiştirmek için yapılandırmayı ayarlayabilirsiniz.

a) OpenShift web konsolunda, gezinme bölmesinden **Operatörler > Kurulu İşlemler** seçeneğini tıklatın.

b) **IBM MQ** seçeneğini tıklatın.

c) **Kuyruk Yöneticisi** etiketini tıklatın.

d) **Create QueueManager** (QueueManager Yarat) düğmesini tıklatın.

A YAML editor is displayed, containing example YAML for a QueueManager resource.

**Not:** Ayrıca, QueueManager yapılandırmasını görüntülemek ya da değiştirmek için **Formu Düzenle** 'yi de tıklatabilirsiniz.

e) Lisans sözleşmesini kabul ederseniz, **Lisans kabulü** seçeneğini **Açık** olarak değiştirin.

IBM MQ , farklı lisanslar altında kullanılabilir. Geçerli lisanslarla ilgili daha fazla bilgi için bkz. [“mq.ibm.com/v1beta1 için lisans başvurusu”](#) sayfa 39. Bir kuyruk yöneticisini konuşturmak için lisansı kabul etmeniz gerekir.

f) **Yarat** düğmesini tıklatın.

Yürürlükteki projeden (ad alanı) kuyruk yöneticilerinin listesi görüntülenir. Yeni QueueManager , bir Pending durumunda olmalıdır.

2. Kuyruk yöneticisinin çalışır durumda olup olmadığını denetleyin

The creation is complete when the QueueManager status is Running.

### İlgili görevler

[“OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma”](#) sayfa 29

Bir Red Hat OpenShift kümesinde konuşturulmuş bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

[“OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması”](#) sayfa 31

Bir Red Hat OpenShift Container Platform kümesine konuşturulan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

## OpenShift CLI

Bir kuyruk yöneticisini komut satırı arabirimini (CLI) kullanarak bir Red Hat OpenShift Container Platform kümesine konuşturmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

## Başlamadan önce

Red Hat OpenShift Container Platform komut satırı arabirimi' yi kurmanız gerekir.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

Bu, bir kuyruk yöneticisini bu Red Hat OpenShift projesinde ilk kez konuşlandırdıysa, [“Preparing your OpenShift project for IBM MQ using the OpenShift CLI”](#) sayfa 16 için adımları izleyin.

## Yordam

### 1. Kuyruk yöneticisini konuşlandırın.

Aşağıdaki örnek, ephemeral (kalıcı olmayan) depolama kullanan ve MQ güvenliğini devre dışı döndüren bir "quick start" kuyruk yöneticisini konuşlandırır. Kuyruk yöneticisinin yeniden başlatılması sırasında iletiler kalıcı olarak saklanmaz. Bir çok kuyruk yöneticisi ayarlarını değiştirmek için YAML içeriğini ayarlayabilirsiniz.

#### a) QueueManager YAML dosyası yarat

Örneğin, IBM Cloud Pak for Integration içinde temel bir kuyruk yöneticisi kurmak için, aşağıdaki içerikle "mq-quickstart.yaml" dosyasını yaratın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart-cp4i
spec:
  version: 9.1.5.0-r2
  license:
    accept: false
    license: L-RJON-BN7PN3
    use: NonProduction
  web:
    enabled: true
  queueManager:
    name: "QUICKSTART"
  storage:
    queueManager:
      type: ephemeral
  template:
    pod:
      containers:
        - name: qmgr
          env:
            - name: MQSNOAUT
              value: "yes"
```

**Önemli:** IBM Cloud Pak for Integration lisans sözleşmesini kabul ederseniz, `accept: false` öğesini `accept: true` olarak değiştirin. Lisansa ilişkin ayrıntılar için [“mq.ibm.com/v1beta1 için lisans başvurusu”](#) sayfa 39 ' e bakın.

Bu örnek ayrıca, kuyruk yöneticisiyle birlikte konuşlandırılan bir web sunucusunu da içerir; web konsolu, Cloud Pak Identity and Access Manager ile Tekli Oturum Açma özelliği etkinleştirilmiş olarak etkinleştirilmiştir.

Temel kuyruk yöneticisini IBM Cloud Pak for Integration' den bağımsız olarak kurmak için, aşağıdaki içerikle "mq-quickstart.yaml" dosyasını yaratın:

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
  name: quickstart
spec:
  version: 9.1.5.0-r2
  license:
    accept: false
    license: L-APIG-BM7GDH
    use: Development
  web:
    enabled: true
  queueManager:
    name: "QUICKSTART"
  storage:
```

```
queueManager:
  type: ephemeral
template:
  pod:
    containers:
      - name: qmgr
        env:
          - name: MQSNOAUT
            value: "yes"
```

**Önemli:** MQ lisans sözleşmesini kabul ederseniz, `accept: false` ögesini `accept: true` olarak değiştirin. Lisansa ilişkin ayrıntılar için [mq.ibm.com/v1beta1](https://mq.ibm.com/v1beta1) için lisans başvurusu” sayfa 39 ' e bakın.

b) QueueManager nesnesini oluşturma

```
oc apply -f mq-quickstart.yaml
```

2. Kuyruk yöneticisinin çalışır durumda olup olmadığını denetleyin  
Konuşlandırmanın geçerliliğini denetleyerek doğrulayabilirsiniz.

```
oc describe queuemanager <QueueManagerResourceName>
```

Ve sonra durumu kontrol ediyorum.

Örneğin,

```
oc describe queuemanager quickstart
```

, and check that the status.Phase field indicates Running

### İlgili görevler

[“OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma” sayfa 29](#)

Bir Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

[“OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması” sayfa 31](#)

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

Linux

MQ Adv.

V 9.1.4

CD

**IBM Cloud Pak for Integration**

## Operations Dashboard ile bütünleştirme

The ability to trace transactions through IBM Cloud Pak for Integration is provided by the Operations Dashboard.

### Bu görev hakkında

Operations Dashboard ile tümleştirilmenin etkinleştirilmesi, kuyruk yöneticinize bir MQ API çıkışı kurar. API çıkışı, kuyruk yöneticisiyle akan iletiler hakkında izleme verilerini Operations Dashboard veri deposuna gönderecektir.

Yalnızca MQ istemci bağlamaları kullanılarak gönderilen iletilerin izlendiğini unutmayın.

### Yordam

1. İzlemenin etkinleştirildiği bir kuyruk yöneticisini konuşlandır

Varsayılan olarak izleme özelliği devre dışı bırakılır.

IBM Cloud Pak for Integration Platform Navigator kullanarak konuşlandırırıyorsanız, konuşlandırma sırasında izlemeyi etkinleştirebilir; konuşlandırma sırasında **İzlemeyi Etkinleştir** ayarını **Açık** olarak ayarlayarak ve **İzleme Ad Alanı** ayarını, Operations Dashboard olanağının kurulu olduğu ad alanına yerleştirebilirsiniz. Kuyruk yöneticisini konuşlandırma hakkında daha fazla bilgi için bkz. [“Deploying a queue manager using the IBM Cloud Pak for Integration Platform Navigator” sayfa 17](#)

If you are deploying using the [OpenShift CLI](#) or [OpenShift Web konsolu](#), then you can enable tracing with the following YAML snippet:

```
spec:
  tracing:
    enabled: true
    namespace: <Operations_Dashboard_Namespace
```

If you are deploying using Helm, then you can enable tracing by setting `odTracingConfig.enabled=true` and `odTracingConfig.odTracingNamespace=<Operations_Dashboard_Namespace`. Var olan bir kuyruk yöneticisinde Operations Dashboard bütünleştirmesini etkinleştirmek istiyorsanız, Helm yayını yükseltirken bu ayarı uygulayabilirsiniz.

**Önemli:** Kuyruk yöneticisi, MQ Operations Dashboard 'a kaydedilinceye kadar başlatılmaz (sonraki adıma bakın).

Bu özellik etkinleştirildiğinde, kuyruk yöneticisi taşıyıcısına ek olarak iki sidecar taşıyıcısı ("Agent" ve "Collector") çalıştırılır. Bu sidecar taşıyıcılara ilişkin resimler, ana MQ görüntüleriyle aynı kayıta kullanılabilir ve aynı çekme ilkesini kullanacak ve güvenlik dizgisini çekecektir. CPU ve bellek sınırlarını yapılandırmak için kullanılacak ek ayarlar da vardır.

2. Bu, Operations Dashboard bütünleştirmesine sahip bir kuyruk yöneticisinin ilk kez bu ad alanında konuşlandırıldıysa, Operations Dashboard ile [Register \(Kaydet\)](#) seçeneğini de göstermeniz gerekir. Kayıt işlemi, kuyruk yöneticisi Pod 'unun başarıyla başlaması gereken bir Gizli nesne yaratır.

## Linux MQ Adv. V 9.1.5 CD OpenShift CLI kullanılarak özel MQSC ve INI dosyaları içeren bir resim oluşturma

Use an Red Hat OpenShift Container Platform Pipeline to create a new IBM MQ container image, with MQSC and INI files you want to be applied to queue managers using this image. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

### Başlamadan önce

[Red Hat OpenShift Container Platform komut satırı arabirimi](#)' yi kurmanız gerekir.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

Red Hat OpenShift projenizdeki IBM Yetkili Kayıt Defteri için bir OpenShift Sıranız yoksa, [“Preparing your OpenShift project for IBM MQ using the OpenShift CLI” sayfa 16](#) için adımları izleyin.

### Yordam

1. Yarat ImageStream

Bir görüntü akışı ve ilişkili etiketleri, Red Hat OpenShift Container Platform'tan taşıyıcı görüntülerine gönderme yapmak için bir soyutlama sağlar. Görüntü akışı ve etiketleri, hangi resimlerin kullanılabilir olduğunu görmeyi ve havuzdaki resim değişse de gereksinim duyduğunuz belirli görüntüyü kullanmanızı sağlar.

```
oc create imagestream mymq
```

2. Yeni görüntünüz için bir BuildConfig yaratın

`BuildConfig`, yeni görüntünüz için oluşturmaları izin verir; bu, IBM resmi görüntüleri temel alınarak oluşturulur, ancak kapsayıcı başlatma sırasında çalıştırılmasını istediğiniz herhangi bir MQSC ya da INI dosyasını ekler.

- a) `BuildConfig` kaynağını tanımlayan bir YAML dosyası yarat

Örneğin, aşağıdaki içerikle "mq-build-config.yaml" adlı bir dosya yaratın:

```
apiVersion: build.openshift.io/v1
kind: BuildConfig
```

```

metadata:
  name: mymq
  spec:
    source:
      dockerfile: |-
        FROM cp.icr.io/cp/ibm-mqadvanced-server-integration:9.1.5.0-r2-amd64
        RUN printf "DEFINE QLOCAL(foo) REPLACE\n" > /etc/mqm/my.mqsc \
          && printf "Channels:\n\tMQIBindType=FASTPATH\n" > /etc/mqm/my.ini
        LABEL summary "My custom MQ image"
    strategy:
      type: Docker
      dockerStrategy:
        from:
          kind: "DockerImage"
          name: "cp.icr.io/cp/ibm-mqadvanced-server-integration:9.1.5.0-r2-amd64"
        pullSecret:
          name: ibm-entitlement-key
    output:
      to:
        kind: ImageStreamTag
        name: 'mymq:latest-amd64'

```

Kullanmak istediğiniz sürüm ve düzeltmeye ilişkin doğru temel görüntüyü işaret etmek için, IBM MQ tabanının belirtildiği iki yerin yerine geçmeniz gerekir. Düzeltmeler uygulandığında, imajını yeniden oluşturmak için bu adımları yinelemeniz gerekir.

Bu örnek, IBM resmi görüntüsünü temel alan yeni bir görüntü yaratır ve "my.mqsc" ve "my.ini" adlı dosyaları /etc/mqm dizinine ekler. Bu dizinde bulunan herhangi bir MQSC ya da INI dosyası, başlatma sırasında taşıyıcı tarafından uygulanacaktır. INI dosyaları, **crtmqm -ii** seçeneği kullanılarak uygulanır ve var olan INI dosyalarıyla birleştirilir. MQSC dosyaları alfabetik sırayla uygulanır.

Kuyruk yöneticisi başlatılırken *her zaman* çalıştırılacağı için, MQSC komutlarınızın yinelenebilir olması önemlidir. This typically means adding the REPLACE parameter on any DEFINE commands, and adding the IGNSTATE (YES) parameter to any START or STOP commands.

b) Apply the BuildConfig to the server.

```
oc apply -f mq-build-config.yaml
```

3. Görüntünüzü oluşturmak için bir oluşturma çalıştırın

a) Oluşturmayı başlat

```
oc start-build mymq
```

Çıktıyı aşağıdakine benzer bir şekilde görmemiz gerekir:

```
build.build.openshift.io/mymq-1 started
```

b) Oluşturmanın durumunu denetleyin

Örneğin, önceki adımda döndürülen oluşturma tanıtıcısını kullanarak aşağıdaki komutu çalıştırabilirsiniz:

```
oc describe build mymq-1
```

4. Yeni görüntünüzü kullanarak bir kuyruk yöneticisini konuşlandırın

[“Deploying a queue manager using the OpenShift CLI”](#) sayfa 18'ta açıklanan adımları izleyin, yeni özel görüntünüzü YAML'ye ekleyin.

YAML 'nin aşağıdaki parçasını normal QueueManager YAMl' nize ekleyebilirsiniz; burada *ad alanı* , kullandığınız OpenShift projesi/ad alanı ve *resim* daha önce oluşturduğunuz resmin adıdır (örneğin, "mymq:latest-amd64"):

```

spec:
  queueManager:
    image: image-registry.openshift-image-registry.svc:5000/my-namespace/my-image

```

## İlgili görevler

[“Deploying a queue manager using the OpenShift CLI”](#) sayfa 18

Bir kuyruk yöneticisini komut satırı arabirimini (CLI) kullanarak bir Red Hat OpenShift Container Platform kümesine konuşlandırmak için QueueManager özel kaynağını kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

## Linux > MQ Adv. > CD **IBM MQ sertifikalı taşıyıcıların Helm kullanılarak konuşlandırılması**

IBM MQ 9.1.5.0' den, bir kuyruk yöneticisini konuşlandırmak için önerilen yöntem, IBM MQ işlecini kullanmandır. IBM MQ 9.1.5.0 ve önceki CD yayınları, aşağıdaki yönergeler kullanılarak Helm kullanılarak konuşlandırılabilir.

### Bu görev hakkında

#### Yordam

- [“Preparing your OpenShift cluster for IBM MQ on OpenShift using Helm” sayfa 23.](#)
- [“Deploying a queue manager using the Helm CLI” sayfa 24.](#)

## Linux > MQ Adv. > CD **Preparing your OpenShift cluster for IBM MQ on OpenShift using Helm**

Prepare your Red Hat OpenShift Container Platform cluster, so that it's ready to deploy a queue manager using Helm. Bu görev bir küme yöneticisi tarafından tamamlanmalıdır.

### Başlamadan önce

**Not:** IBM Cloud Pak for Integration kullanıyorsanız, kuruluş programı IBM MQ ile birlikte kullanmanız için bir OpenShift projesi (ad alanı) hazırlamış ve bu nedenle bu yönergeleri izlemenize gerek kalmayabilir.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

#### Yordam

1. IBM Helm havuzunu Helm' nin yerel kopyasına eklediğinizden emin olun. Örneğin, aşağıdaki komutu çalıştırabilirsiniz:

```
helm repo add ibm-entitled-charts https://raw.githubusercontent.com/IBM/charts/master/rep/entitled
```

2. Helm sunucusunun ("Tiller" olarak adlandırılır) olduğundan emin olun. Küğinize kurulum. Follow the instructions in [Getting started with Helm on OpenShift to install Helm on your cluster.](#)
3. OpenShift projenizdeki (ad alanı) Hizmet Hesaplarının, doğru Güvenlik Bağlamı Kısıtlamalarını (SCC 'ler) kullanma yetkisine sahip olduğundan emin olun.

**V 9.1.5** IBM MQ , "restricted" varsayılan SCC altında çalışır, bu nedenle bu adım olağan şekilde atlanabilir.

Değişiklikleri SCC ' lere uygulamak için bir OpenShift kümesi yöneticisi tarafından yapılması gerekir. Her bir Helm grafik sürümü, bu Helm grafiği için tek tek BENİOKU dosyasında belgelenmiş olan SCC ' ler için farklı gereksinimlere sahiptir:

```
helm inspect readme ibm-entitled-charts/ibm-mqadvanced-server-prod
```

Her bir README 'de SCC' lerin yetkilendirilmesinde kullanılan yönergeler vardır. IBM MQ Helm grafiklerinin kendi kullanımı için bir Hizmet Hesabı oluşturmaya dikkat edin. Bu, "group" (grup) düzeyinde (ad alanındaki tüm Hizmet Hesapları için) SCC izinlerinin uygulanması gerektiği anlamına gelir.

4. Seçilen taşıyıcı kaydınızdan görüntü çekmek için geçerli bir "resim çekme sıranız" sağladığınızdan emin olun

IBM MQ Advanced certified container görüntüleri, lisans hakkı denetimi gerçekleştiren bir kapsayıcı kayıt dosyasından çekiliyor. Bu denetim, bir `docker-registry` çekme sırasında saklanan bir yetki anahtarını gerektirir. Henüz bir yetki anahtarınız yoksa, bir yetki anahtarı almak ve bir çekme güvenlik dizgisi yaratmak için bu yönergeleri izleyin.

a) Kimliğinize atanan yetki anahtarını alın.

- i) Yetkili yazılımla ilişkili IBM kimliği ve parolasıyla [MyIBM Kapsayıcı Yazılım Kitaplığı](#) olanağında oturum açın.
- ii) In the *Yetki anahtarları* section, select **Kopyalama anahtarı** to copy the entitlement key to the clipboard.

b) Kuyruk yöneticinizi konuşlandırmak istediğiniz ad alanında gizli bilgi yaratın.

- Şu komutu çalıştırın; burada `<entitlement-key>` , 1. adımda alınan anahtardır ve `<user-email>` , yetkili yazılımla ilişkilendirilen IBM kimliğidir.

```
oc create secret docker-registry ibm-entitlement-key \
--docker-server=cp.icr.io \
--docker-username=cp \
--docker-password=<entitlement-key> \
--docker-email=<user-email>
```

## Sonraki adım

[“Deploying a queue manager using the Helm CLI” sayfa 24](#)

## Linux MQ Adv. V 9.1.4 CD Deploying a queue manager using the Helm CLI

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesinde konuşlandırmak için Helm ögesini kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

## Başlamadan önce

Helm V2 ve Red Hat OpenShift Container Platform komut satırı arabirimini kurmanız gerekir. IBM Cloud Pak for Integration kullanmıyorsa, [“Preparing your OpenShift cluster for IBM MQ on OpenShift using Helm” sayfa 23](#) adımlarını izleyin.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

## Yordam

1. IBM Helm havuzunu Helm' nin yerel kopyasına eklediğinizden emin olun. Örneğin, aşağıdaki komutu çalıştırabilirsiniz:

```
helm repo add ibm-entitled-charts https://raw.githubusercontent.com/IBM/charts/master/repo/entitled
```

2. Kuyruk yöneticinize ilişkin yapılandırma seçeneklerini gözden geçirin

Konuşlandırma adımında hem kuruluş, hem de yapılandırma adımları bulunur. Kuyruk yöneticinizin bazı ayarları, konuşlandırma sırasında ayarlanmalıdır ve bunların değiştirilmesi yeniden devreye alma gerektirir.

Aşağıdaki komutlardan birini çalıştırarak, kullanılabilir tüm konuşlandırma seçeneklerine ilişkin ayrıntılar için Helm grafit README (BENİOKU) çizelgesini görüntüleyebilirsiniz:

- IBM Cloud Pak for Integration içinde IBM MQ Advanced certified container için:

```
helm inspect readme ibm-entitled-charts/ibm-mqadvanced-server-integration-prod
```



- IBM MQ Advanced certified container için:

```
helm inspect readme ibm-entitled-charts/ibm-mqadvanced-server-prod
```

Tipik olarak, en az aşağıdaki parametrelere gereksinim duyarsınız:

- Yayın adı. Örneğin: `my-release`
- Uzak Helm havuzu. Örneğin: `ibm-entitled-charts`
- Helm grafiği: örneğin, `ibm-mqadvanced-server-prod` ya da `ibm-mqadvanced-server-integration-prod`
- Resim çekme gizli adı. Örneğin: `entitled-registry`. IBM Cloud Pak for Integration içinde MQ için önceden tanımlanmış bir projeye konuşlandırıyorsanız, buna gerek yoktur.

### 3. Kuyruk yöneticisini konuşlandırın.

Varsayılan olarak, Helm grafiğinizde, Red Hat OpenShift Container Platform kümesinde varsayılan [Storage Class](#) (Depolama Sınıfı) ayarınız olduğunu varsayar.

Örneğin, IBM Cloud Pak for Integration' ta temel bir kuyruk yöneticisi kurmak için aşağıdaki komutu çalıştırın:

```
helm install \
--tls \
--name my-release \
ibm-entitled-charts/ibm-mqadvanced-server-integration-prod \
--set license=accept \
--set tls.hostname=my.cluster \
--set tls.generate=true
```

`tls.hostname` alanına herhangi bir ana makine adı girebilirsiniz (bu, veri girilmesi zorunlu bir alandır, ancak bu örnekte olduğu gibi, yeni bir kendinden onaylı sertifika oluşturuyoruz)

Temel kuyruk yöneticisini IBM Cloud Pak for Integration' den bağımsız olarak kurmak için aşağıdaki komutu çalıştırabilirsiniz:

```
helm install \
--name my-release \
ibm-entitled-charts/ibm-mqadvanced-server-prod \
--set license=accept \
--set image.pullSecret=ibm-entitlement-key
```

## İlgili görevler

[“OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma” sayfa 29](#)

Bir Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

[“OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması” sayfa 31](#)

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

## Linux > MQ Adv. > V 9.1.5 > CD **Deploying a queue manager with IBM**

### **Cloud File Storage, using the Helm CLI**

Example scenario to use Helm to deploy a queue manager onto a Red Hat OpenShift on IBM Cloud cluster, using IBM Cloud File Storage. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır

## **Başlamadan önce**

Helm V2 ve Red Hat OpenShift Container Platform komut satırı arabiriminin kurmanız gerekir. IBM Cloud Pak for Integration kullanmıyorsa, [“Preparing your OpenShift cluster for IBM MQ on OpenShift using Helm” sayfa 23](#) adımlarını izleyin.

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

## Yordam

1. IBM Helm havuzunu Helm' nin yerel kopyasına eklediğinizden emin olun.

Örneğin, aşağıdaki komutu çalıştırabilirsiniz:

```
helm repo add ibm-entitled-charts https://raw.githubusercontent.com/IBM/charts/master/repo/entitled
```

2. Kuyruk yöneticisini konuşlandırın.

IBM Cloud File Storage kullanırken, genellikle `ibmc-file-gold-gid` depolama sınıfını kullanarak en iyi sonuçları görürsünüz. Bu depolama sınıfı, doğru dosya sistemi grubundaki kullanıcılar tarafından yazılabilen depolamayı etkinleştirir.

Örneğin, IBM Cloud Pak for Integration' ta temel bir kuyruk yöneticisi kurmak için aşağıdaki komutu çalıştırın:

```
helm install \
--tls \
--name my-release \
ibm-entitled-charts/ibm-mqadvanced-server-integration-prod \
--set license=accept \
--set tls.hostname=my.cluster \
--set tls.generate=true \
--set dataPVC.storageClassName=ibmc-file-gold-gid \
--set security.context.supplementalGroups={99}
```

`tls.hostname` alanına herhangi bir ana makine adı girebilirsiniz (bu, zorunlu bir alandır, ancak burada kullanılmaz; bu örnekte, yeni bir kendinden onaylı sertifika oluşturuyoruz).

Temel kuyruk yöneticisini IBM Cloud Pak for Integration' den bağımsız olarak kurmak için aşağıdaki komutu çalıştırabilirsiniz:

```
helm install \
--name my-release \
ibm-entitled-charts/ibm-mqadvanced-server-prod \
--set license=accept \
--set image.pullSecret=ibm-entitlement-key \
--set dataPVC.storageClassName=ibmc-file-gold-gid \
--set security.context.supplementalGroups={99}
```

## İlgili görevler

[“OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma” sayfa 29](#)

Bir Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

[“OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması” sayfa 31](#)

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

Linux

MQ Adv.

CD

## Deploying previous CD releases of IBM MQ into an IBM Cloud Private cluster

For CD versions of IBM MQ earlier than 9.1.4, use the IBM Cloud Private management console to deploy a queue manager into IBM Cloud Private.

## Başlamadan önce



**Uyarı:**

**V 9.1.4**

Bu konuşlandırma IBM MQ 9.1.4 ya da sonraki sürümlerde desteklenmez.

Bu görev, önceden [IBM Cloud Private kümesine bir IBM MQ görüntüsü ekledi](#) olduğuna varsayar.

The Helm chart README .md file is available from the IBM Cloud Private catalog entry, which is displayed after you complete Alt Adım, or from the command line by adding your IBM Cloud Private's **local-charts** repository as a remote Helm repository and running the following command:

```
helm inspect readme remote_repo_name/ibm-mqadvanced-server-prod
```

Gerekli güvenlik bağlamını destekleyen bir PodSecurityPolicy'ya da bir SecurityContextConstraint ( Red Hat OpenShift üzerinde IBM Cloud Private için) olmalıdır. Örnekler de dahil olmak üzere, Helm grafik README .md dosyasından ayrıntılar bulunabilir.

Helm yayınınızı nasıl yapılandırabileceğinize ilişkin ayrıntılar, Helm grafik README .md dosyasında da bulunabilir.

#### Not:

- Varsayılan olarak gerekli güvenlik ayarlarını desteklemeyen bir IBM Cloud Private ortamında devreye alıyorsanız, IBM Cloud Private ürün belgelerindeki Varsayılan olmayan bir ad alanında yükseltilmiş ayrıcalıklar gerektiren Helm grafiklerinin konuşlandırılması içindeki yönergeleri izleyerek devreye alımınızı etkinleştirin.
- If you are using SELinux, you must meet the IBM MQ requirements described in IBM MQ support for SELinux on Red Hat Enterprise Linux.

## Bu görev hakkında

IBM Cloud Private , şirket içi, containerize uygulamaları yönetmek için bir platform sunar. After you have added an IBM MQ image into an IBM Cloud Private cluster, you can use either the IBM Cloud Private management console or the command line to deploy a queue manager.

## Yordam

- IBM Cloud Private Management Console olanağının kullanılması
  - IBM Cloud Private yönetim konsolunu bir web tarayıcısında açın ve **Catalog**(Katalog) seçeneğini tıklatın.

IBM Cloud Private ürün belgelerindeki Yönetim konsolunu kullanarak IBM Cloud Private kümesine erişilmesi başlıklı konuya bakın.
  - Listeden `ibm-mqadvanced-server-prod` grafimesini seçin.
  - Configure**(Yapılandır) seçeneğini belirleyin ve aşağıdaki yapılandırma adımlarını tamamlayın:
    - Bir yayın adı girin.
    - Lisans sözleşmelerini okuyun ve kabul edin.
    - dataPVC** bölümünün altında, **storageclass** ögesini istediğiniz depolama sınıfınıza ayarlayın. Varsayılan depolama sınıfını seçmek için boş bırakın.
    - resim** bölümü altında, havuzu tam görüntü yoluna ayarlayın. Örneğin:

```
mycluster.icp:8500/namespace_name/ibm-mqadvanced-server-prod
```
    - Resim** bölümünün altında, etiketi görüntü etiketine ayarlayın. Örneğin:

```
9.1.3.0-r1
```
    - Görüntü kaydına erişmek için Kubernetes ' in gizli bir güvenlik dizgisi gerekiyorsa, bunu **pullSecret**olarak ekleyin.
    - queueManager** bölümünün altında, kuyruk yöneticisinin adını belirleyin.
  - Kuyruk yöneticinizi *Helm yayın düzeyi*olarak konuşlandırmak için **Kur** düğmesini tıklatın.
- Komut satırını kullanma
  - cloudctl** ' u IBM Cloud Private kümeniz için erişecek şekilde yapılandırın.

See [IBM Cloud Private CLI 'nın kurulması](#) in the IBM Cloud Private product documentation.

- b) Ensure that you have added your IBM Cloud Private's **local-charts** repository as a remote Helm repository.
- c) Grafiği kurun.

Bu deęiřtirgeleri belirterek ařaęıdaki komutu alıřtırın:

- a. Yayın adı (örneğin, my-release)
- b. ibm-mqadvanced-server-prod grafimesini ieren uzak helm havuzunun adı (örneğin, my-repo)
- c. Görüntü havuzu (örneğin, mycluster.icp:8500/namespace\_name/ibm-mqadvanced-server-prod)
- d. Resim etiketi (örneğin, 9.1.3.0-r1)

```
helm install --name my-release --repo my-repo ibm-mqadvanced-server-prod --set license=accept --set image.repository=mycluster.icp:8500/namespace_name/ibm-mqadvanced-server-prod --set image.tag=9.1.3.0-r1 --tls
```

### İlgili görevler

[“Deploying a queue manager using the Helm CLI” sayfa 24](#)

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesinde konuşlandırmak için Helm ögesini kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

[“IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Private kümesine eklenmesi” sayfa 28](#)

For CD versions of IBM MQ earlier than 9.1.4, prepare your IBM Cloud Private cluster to deploy a production-ready image for IBM MQ.

[“IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Kubernetes Service kümesine eklenmesi” sayfa 29](#)

For CD versions of IBM MQ earlier than 9.1.4, import a production-ready image for IBM MQ into IBM Cloud Kubernetes Service.

Linux

MQ Adv.

CD

## IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Private kümesine eklenmesi

For CD versions of IBM MQ earlier than 9.1.4, prepare your IBM Cloud Private cluster to deploy a production-ready image for IBM MQ.

### Bu görev hakkında



**Uyarı:** **V 9.1.4** Bu ie aktarma, IBM MQ 9.1.4 ya da sonraki sürümlerde desteklenmez.

Bir IBM MQ görüntüsünü Passport Advantage ' den yükleyebilir ve bir IBM Cloud Private taşıyıcısına aktarabilirsiniz.

### Yordam

1. En son IBM MQ görüntüsünü [Passport Advantage ve Passport Advantage Express web sitesi](#)' den yükleyin.

Kullanılabilir karřıdan yüklemelere iliřkin ayrıntılar için [IBM MQ 9.1](#) ' e gidin ve karřıdan yüklemek istedięiniz yayının sekmesini tıklatın. Ařaęı yüklenecek paranın adı ve numarası bir izelgede listelenir.

2. Karřıdan yüklenen arřiv dosyasını IBM Cloud Privateiine alın.

See [Adding IBM software to the IBM Cloud Private Catalog](#) in the IBM Cloud Private product documentation.

## Sonraki adım

Artık Kuyruk yöneticisini IBM Cloud Private' e konuşlandır' a hazırsınız.

### İlgili görevler

[“Deploying a queue manager using the Helm CLI” sayfa 24](#)

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesinde konuşlandırmak için Helm ögesini kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

[“Deploying previous CD releases of IBM MQ into an IBM Cloud Private cluster” sayfa 26](#)

For CD versions of IBM MQ earlier than 9.1.4, use the IBM Cloud Private management console to deploy a queue manager into IBM Cloud Private.

[“IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Kubernetes Service kümesine eklenmesi” sayfa 29](#)

For CD versions of IBM MQ earlier than 9.1.4, import a production-ready image for IBM MQ into IBM Cloud Kubernetes Service.

Linux

MQ Adv.

CD

## IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Kubernetes Service kümesine eklenmesi

For CD versions of IBM MQ earlier than 9.1.4, import a production-ready image for IBM MQ into IBM Cloud Kubernetes Service.

### Bu görev hakkında



Uyarı:

V 9.1.4

Bu içe aktarma, IBM MQ 9.1.4 ya da sonraki sürümlerde desteklenmez.

Bir IBM MQ görüntüsünü Passport Advantage ' den yükleyebilir ve bir IBM Cloud Kubernetes Service kümesine aktarabilirsiniz.

### Yordam

1. En son IBM MQ görüntüsünü [Passport Advantage ve Passport Advantage Express web sitesi](#)' den yükleyin.

Kullanılabilir karşidan yüklemelere ilişkin ayrıntılar için [IBM MQ 9.1](#) ' e gidin ve karşidan yüklemek istediğiniz yayının sekmesini tıklayın. Aşağı yüklenecek parçanın adı ve numarası bir çizelgede listelenir.

2. Karşidan yüklenen arşiv dosyasını IBM Cloud Kubernetes Service'ine alın.

Bkz. [Genel Kubernetes kapsayıcılarında IBM Cloud Private görüntülerinin çalıştırılması](#).

### İlgili görevler

[“Deploying a queue manager using the Helm CLI” sayfa 24](#)

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesinde konuşlandırmak için Helm ögesini kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

[“Deploying previous CD releases of IBM MQ into an IBM Cloud Private cluster” sayfa 26](#)

For CD versions of IBM MQ earlier than 9.1.4, use the IBM Cloud Private management console to deploy a queue manager into IBM Cloud Private.

[“IBM MQ görüntüsünün önceki CD yayınlarının bir IBM Cloud Private kümesine eklenmesi” sayfa 28](#)

For CD versions of IBM MQ earlier than 9.1.4, prepare your IBM Cloud Private cluster to deploy a production-ready image for IBM MQ.

Linux

MQ Adv.

V 9.1.4

CD

## OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma

Bir Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

## Bu görev hakkında

Bir uygulamayı Red Hat OpenShift kümesi dışındaki bir IBM MQ kuyruk yöneticisine bağlamak için bir [OpenShift Rotası](#) gerekir.

You must enable TLS on your IBM MQ queue manager and client application, because [Sunucu Adı Göstergesi \(SNI\)](#) is only available in the TLS protocol. Red Hat OpenShift Container Platform Yöneticisi, SNI ' yi IBM MQ kuyruk yöneticisine yöneltmek için SNI kullanır.

OpenShift Rotası için gerekli yapılandırma, istemci uygulamanızın SNI davranışına bağlıdır.

SNI üstbilgisini TLS 1.2 ya da üstü olarak ayarlamak için, TLS iletişiminiz için bir CipherSpec ya da CipherSuite kullanılmalıdır.

SNI, aşağıdaki koşullar karşılanırsa MQ kanalına ayarlanır:

- IBM MQ C İstemcisi V8 ya da sonraki bir yayın düzeyidir.
- Java/JMS İstemcisi V9.1.1 ya da sonraki bir yayın düzeyidir ve Java kuruluşu `javax.net.ssl.SNIHostName` sınıfını destekler.
- .NET istemcisi yönetilmeyen kiptir.

Bağlantı adı olarak bir anasistem adı belirtilirse ve aşağıdaki koşullar karşılanırsa, SNI Ana Makine Adı (Anasistem Adı) olarak ayarlanır:

- .NET İstemcisi yönetilen kiptir.
- AMQP ya da XR istemcisi kullanılır.
- The Java/JMS Clients are used with **AllowOutboundSNI** set to `Hayır`.

SNI ayarlanmadı ve aşağıdaki koşullar altında boştur:

- IBM MQ C Client, V7.5 ya da önceki bir sürümdür.
- IBM MQ C Client is used with **AllowOutboundSNI** set to `Hayır`.
- Java/JMS İstemcileri, `javax.net.ssl.SNIHostName` sınıfını desteklemeyen bir Java kuruluşuyla birlikte kullanılır.

## Örnek

### **Anasistem adı tabanlı OpenShift Rotaları: SNI ' yi ana makine adına ayarlayan istemci uygulamaları için**

Aşağıdaki Helm grafikleri, bir uygulamayı IBM MQ kuyruk yöneticisine bağlamak için otomatik olarak bir anasistem adı ( OpenShift Rotası) yaratır. SNI ' yi ana makine adına ayarlayan istemci uygulamaları bu OpenShift Rotasını kullanabilir.

- `ibm-mqadvanced-server-dev`
- `ibm-mqadvanced-server-prod`
- `ibm-mqadvanced-server-integration-prod` ( IBM Cloud Pak for Integration).

Bu grafikleri kullanmıyorsa ve kendi ana makine adınızı temel alan OpenShift Rotasını oluşturmanız gerekirse, kümenize aşağıdaki `yaml` ' yi uygulayabilirsiniz:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: <provide a unique name for the Route>
  namespace: <namespace of your MQ deployment>
spec:
  to:
    kind: Service
    name: <name of the Kubernetes Service for your MQ deployment (for example "<Helm Release>-ibm-
mq")>
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

## **MQ kanalı tabanlı OpenShift Routes: SNI ' yi MQ kanalına ayarlayan istemci uygulamaları için**

SNI ' yi MQ kanalına ayarlayan istemci uygulamaları, bağlanmak istediğiniz her kanal için yeni bir OpenShift Rotası yaratılmasını gerektirir. Doğru kuyruk yöneticisine yönlendirmeye izin vermek için Red Hat OpenShift kümenizde benzersiz kanal adları da kullanmanız gerekir.

Yeni OpenShift Routes 'in her biri için gerekli olan ana makine adını belirlemek üzere, her kanal adını burada belgelendiği gibi bir SNI adresiyle eşlemeniz gerekir: <https://www.ibm.com/support/pages/ibm-websphere-mq-how-does-mq-provide-multiple-certificates-certlabl-capability>

Daha sonra, kümeniz için aşağıdaki yam1 ' i uygulayarak yeni bir OpenShift Rotası (her kanal için) oluşturmanız gerekir:

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  name: <provide a unique name for the Route>
  namespace: <the namespace of your MQ deployment>
spec:
  host: <SNI address mapping for the channel>
  to:
    kind: Service
    name: <the name of the Kubernetes Service for your MQ deployment (for example "<Helm Release>-ibm-mq")>
  port:
    targetPort: 1414
  tls:
    termination: passthrough
```

### **İstemci uygulaması bağlantı ayrıntılarının yapılandırılması**

İstemci bağlanmanız için kullanılacak anasistem adını aşağıdaki komutu çalıştırarak belirleyebilirsiniz:

```
oc get route <Name of hostname based Route (for example "<Helm Release>-ibm-mq-qm")>
-n <namespace of your MQ deployment> -o jsonpath="{.spec.host}"
```

İstemci bağlantınızın kapısı, OpenShift Container Platform (OCP) Yöneticisi tarafından kullanılan kapı olarak ayarlanmalıdır-normalde 443.

### **İlgili görevler**

[“Deploying a queue manager using the Helm CLI” sayfa 24](#)

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesinde konuşlandırmak için Helm ögesini kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

[“OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması” sayfa 31](#)

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

## **Linux** **MQ Adv.** **V 9.1.4** **CD** **OpenShift kümesinde devreye alınan IBM MQ Console ile bağlantı kurulması**

Bir Red Hat OpenShift Container Platform kümesine konuşlandırılan bir kuyruk yöneticisinin IBM MQ Console ile bağlantı kurma yöntemi.

### **Bu görev hakkında**

IBM MQ işlecini kullanıyorsanız, IBM MQ Console URL 'si QueueManager ayrıntı sayfasında OpenShift web konsolunun ya da IBM Cloud Pak for Integration Platform Navigator' in sayfasında bulunabilir. Diğer bir seçenek olarak, aşağıdaki komutu çalıştırarak OpenShift CLI 'sından da bulunabilir:

```
oc get queuemanager <QueueManager Name> -n <namespace of your MQ deployment> --output
jsonpath='{.status.adminUiUrl}'
```

## Örnek

Aşağıdaki Helm grafikleri, IBM MQ Console' e erişmek için otomatik olarak bir OpenShift Rotası oluşturur.

- ibm-mqadvanced-server-dev
- ibm-mqadvanced-server-integration-prod ( IBM Cloud Pak for Integration).

Aşağıdaki komutu çalıştırarak OpenShift Route 'un anasistem adını alabilirsiniz:

```
oc get route <Route Name (for example "<Helm Release>-ibm-mq-web")>
-n <namespace of your MQ deployment> --output jsonpath='{.spec.host}'
```

IBM MQ Console 'a aşağıdaki URL' yi kullanarak erişebilirsiniz:

```
https://<Route Hostname>/ibmmq/console
```

## İlgili görevler

[“Deploying a queue manager using the Helm CLI” sayfa 24](#)

Bir kuyruk yöneticisini bir Red Hat OpenShift Container Platform kümesinde konuşlandırmak için Helm ögesini kullanın. Bu görev bir proje yöneticisi tarafından tamamlanmalıdır.

[“OpenShift kümesinde devreye alınan bir kuyruk yöneticisine bağlanma” sayfa 29](#)

Bir Red Hat OpenShift kümesinde konuşlandırılmış bir kuyruk yöneticisine bağlanmak için kullanılan yapılandırma örnekleri kümesi.

Linux

MQ Adv.

CD

## Backing up and restoring queue manager configuration using the OpenShift CLI

Kuyruk yöneticisi yapılandırmalarının yedeklenmesi, kuyruk yöneticisi yapılandırmalarının kaybolması durumunda, kuyruk yöneticisini tanımlarından yeniden oluşturmanıza yardımcı olabilir. Bu yordam kuyruk yöneticisi günlük verilerini yedeklemez. İletilerin geçici niteliği nedeniyle, geçmiş günlük verilerinin geri yükleme sırasında ilgisiz olması olasılığı vardır.

## Başlamadan önce

Log into your cluster using **cloudctl login** (for IBM Cloud Pak for Integration), or **oc login**.

## Yordam

- Kuyruk yöneticisi yapılandırması yedekle.

Bir IBM MQ kuyruk yöneticisinin yapılandırmasını dökümünü almak için **dmpmqcfig** komutunu kullanabilirsiniz.

- a) Kuyruk yöneticinizin kapsülünün adını alın.

Örneğin, İşleci kullanıyorsanız, aşağıdaki komutu çalıştırabilirsiniz; burada *queue\_manager\_name* , QueueManager kaynağınız adıdır:

```
oc get pods --selector app.kubernetes.io/name=ibm-mq,app.kubernetes.io/instance=queue_manager_name
```

Örneğin, Helm kullanıyorsanız, aşağıdaki komutu çalıştırabilirsiniz; burada *release\_name* , Helm yayın düzeyinizin adıdır.

```
oc get pods --selector release=release_name
```

- b) Kapsülde **dmpmqcfig** komutunu çalıştırın, çıkışı yerel makinenizdeki bir dosyaya yönlendirin.

**dmpmqcfig** , kuyruk yöneticisinin MQSC yapılandırmasını çıkarır.

```
oc exec -it pod_name -- dmpmqcfig > backup.mqsc
```



- Kuyruk yöneticisi yapılanışını geri yükleyin.

Önceki adımda belirtilen yedekleme yordamını izlemiş olması, kuyruk yöneticisi yapılanışını içeren bir backup.mqsc kütüğüne sahip olmalıdır. Bu dosyayı yeni bir kuyruk yöneticisine uygulayarak yapılandırmayı geri yükleyebilirsiniz.

- a) Kuyruk yöneticinizin kapsülünün adını alın.

Örneğin, İşleci kullanıyorsanız, aşağıdaki komutu çalıştırabilirsiniz; burada *queue\_manager\_name* , QueueManager kaynağınız adıdır:

```
oc get pods --selector app.kubernetes.io/name=ibm-mq,app.kubernetes.io/instance=queue_manager_name
```

Örneğin, Helm kullanıyorsanız, aşağıdaki komutu çalıştırabilirsiniz; burada *release\_name* , Helm yayın düzeyinizin adıdır.

```
oc get pods --selector release=release_name
```

- b) Modülde **runmqsc** komutunu çalıştırın, backup.mqsc dosyasının içeriğini yönlendirin.

```
oc exec -i pod_name -- runmqsc < backup.mqsc
```

## Kendi IBM MQ taşıyıcınız oluşturma

Önceden "Docker taşıyıcı görüntüsü" olarak adlandırılan, kendi kendine yapılı bir taşıyıcı geliştirin. Bu, en esnek taşıyıcı çözümdür, ancak kapsayıcıların yapılandırılmasına ilişkin güçlü becerilere ve sonuçta elde edilen kapsayıcıya "Kendi" ' e gereksinim duymanızı gerektirir.

### Başlamadan önce

Kendi kapsayıcınız geliştirmeden önce, bunun yerine IBM tarafından sağlanan önceden paketlenmiş kaplardan birini kullanıp kullanamayacağınızı göz önünde bulundurun. Bkz. [IBM MQ kapsayıcılarında](#)

### Bu görev hakkında

When you package IBM MQ as a container image, changes to your application can be deployed to test and staging systems quickly and easily. Bu, işletmeniz içinde sürekli teslimat için önemli bir avantaj olabilir.

### Yordam

- Docker komutunu kullanarak IBM MQ taşıyıcı görüntüsünün nasıl oluşturulabilmesiyle ilgili bilgi edinmek için aşağıdaki alt konulara bakın:
  - [Linux](#) [“Kendi IBM MQ kapsayıcı resimlerinizi ve grafiklerinizi oluşturmak için destek” sayfa 8](#)
  - [“Bir taşıyıcıyı kullanarak kendi IBM MQ kuyruk yöneticisi görüntüsünün planlanması” sayfa 33](#)
  - [“Building a sample IBM MQ queue manager image using Docker” sayfa 34](#)
  - [“Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması” sayfa 37](#)

### İlgili kavramlar

[Taşıyıcılarda IBM MQ](#)

## Bir taşıyıcıyı kullanarak kendi IBM MQ kuyruk yöneticisi görüntüsünün planlanması

Bir taşıyıcıda bir IBM MQ kuyruk yöneticisi çalıştırırken dikkate alınması gereken bazı gereksinimler vardır. Örnek taşıyıcı görüntüsü, bu gereksinimleri ele almak için bir yol sağlar; ancak, kendi görüntünüzü kullanmak istiyorsanız, bu gereksinimlerin nasıl işlendiğini göz önünde bulundurmanız gerekir.

## Süreç denetimi

Bir taşıyıcıyı çalıştırdığınızda, temel olarak, daha sonra alt süreçleri yumurtlayabilen tek bir süreç (taşıyıcının içinde 1 PID) çalıştırıyorsunuz demektir.

Ana işlem sona ererse, taşıyıcı yürütme ortamı taşıyıcıyı durdurur. Bir IBM MQ kuyruk yöneticisi, arka planda birden çok işlem çalıştırılmasını gerektirir.

Bu nedenle, kuyruk yöneticisi çalışır durumda olduğundan ana işleminizin etkin kalmasını sağlamak gerekir. Örneğin, kuyruk yöneticisinin bu süreçten (örneğin, denetim sorguları gerçekleştirerek) etkin olup olmadığını kontrol etmek iyi bir uygulamadır.

## /var/mqm veri yerleştiriliyor

Kapsayıcılar, bir birim olarak /var/mqm ile yapılandırılmalıdır.

Bunu yaparken, taşıyıcı ilk başlatıldığında birimin dizini boş olur. Bu dizin genellikle kuruluş sırasında doldurulur, ancak kuruluş ve yürütme ortamı bir taşıyıcı kullanılırken ayrı ortamlardır.

**V9.1.0** To solve this, when your container starts, you can use the `crtmqdir` command to populate /var/mqm when it runs for the first time.

## Building a sample IBM MQ queue manager image using Docker

Bir taşıyıcıda IBM MQ kuyruk yöneticisini çalıştırmak için örnek bir taşıyıcı görüntüsü oluşturmak için bu bilgileri kullanın.

### Bu görev hakkında

Firstly, you build a base image containing an Red Hat Universal Base Image file system and a clean installation of IBM MQ.

İkinci olarak, temel kullanıcı kimliği ve parola güvenliği sağlamak için bazı IBM MQ yapılandırması ekleyen temel tabanın üzerine başka bir taşıyıcı görüntü katmanı oluşturursunuz.

Son olarak, bu görüntüyü dosya sistemi olarak kullanarak, anasistem dosya sistemindeki bir taşıyıcıya özgü bir birim tarafından sağlanan /var/mqm içeriğiyle birlikte bir kapsayıcı çalıştırıyorsunuz.

### Yordam

- Bir taşıyıcıda IBM MQ kuyruk yöneticisini çalıştırmak için örnek bir taşıyıcı görüntüsünün oluşturulmasına ilişkin bilgi edinmek için aşağıdaki alt konulara bakın:
  - [“Örnek temel IBM MQ kuyruk yöneticisi görüntüsü oluşturulması” sayfa 34](#)
  - [“Örnek yapılandırılmış bir IBM MQ kuyruk yöneticisi görüntüsü oluşturulması” sayfa 35](#)

## Örnek temel IBM MQ kuyruk yöneticisi görüntüsü oluşturulması

Kendi kap resminizde IBM MQ 'yi kullanmak için, öncelikle temiz bir IBM MQ kuruluşuna sahip bir temel görüntü oluşturmada gerekir. Aşağıdaki adımlar, GitHub' da barındırılan örnek kodu kullanarak örnek temel bir resmin nasıl oluşturulacağı gösterilmektedir.

### Yordam

- Üretim kabınız görüntüsünü oluşturmak için [mq-container GitHub havuzu](#) ' ta sağlanan make dosyalarını kullanın.  
GitHub' da [Taşıyıcı görüntüsü oluşturma](#) konusundaki yönergeleri izleyin.

### Sonuçlar

Şimdi IBM MQ kurulu bir temel taşıyıcı görüntüsünüz var.

## Örnek yapılandırılmış bir IBM MQ kuyruk yöneticisi görüntüsü oluşturulması

Soysal temel IBM MQ taşıyıcı görüntüsünü oluşturduktan sonra, güvenli erişime izin vermek için kendi yapılandırmanızı uygulamanız gerekir. Bunu yapmak için, genel görüntüyü bir üst öge olarak kullanarak kendi taşıyıcı görüntü katmanınızı yaratın.

### Başlamadan önce

Bir IBM MQ 9.1 görüntüsü için, Red Hat OpenShift Container Platform "restricted" Security Constraint Constraint (SCC) olanağını kullanarak güvenli erişimi yapılandırabilirsiniz. "Restricted" SCC rasgele kullanıcı kimliklerini kullanır ve farklı bir kullanıcıya çevrilerek ayrıcalık yükseltmeyi önler. IBM MQ 9.1 RPM tabanlı kuruluş programı bir mqm kullanıcı ve grubuna dayanır ve yürütülebilir programlar üzerinde setuid bitlerini kullanır.

Bu kısıtlama IBM MQ 9.2' ta kaldırılır.

### Yordam

1. Yeni bir dizin yaratın ve aşağıdaki içerikle birlikte `config.mqsc` adlı bir dosya ekleyin:

```
DEFINE CHANNEL(PASSWORD.SVRCONN) CHLTYPE(SVRCONN)
SET CHLAUTH(PASSWORD.SVRCONN) TYPE(BLOCKUSER) USERLIST('nobody') +
DESCR('Allow privileged users on this channel')
SET CHLAUTH('*') TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(NOACCESS) DESCR('BackStop rule')
SET CHLAUTH(PASSWORD.SVRCONN) TYPE(ADDRESSMAP) ADDRESS('*') USERSRC(CHANNEL) CHCKCLNT(REQUIRED)
ALTER AUTHINFO(SYSTEM.DEFAULT.AUTHINFO.IDPWOS) AUTHTYPE(IDPWOS) ADOPTCTX(YES)
REFRESH SECURITY TYPE(CONNAUTH)
```

Önceki örneğe basit kullanıcı kimliği ve parola doğrulaması kullandığına dikkat edin. Ancak, teşebbüsünüzün gerektirdiği herhangi bir güvenlik yapılandırmasını uygulayabilirsiniz.

2. Aşağıdaki içerikle birlikte, `Dockerfile` adlı bir dosya oluşturun:

```
FROM mq
RUN useradd johndoe -G mqm && \
    echo johndoe:passwd | chpasswd
COPY config.mqsc /etc/mqm/
```

Burada:

- johndoe , eklemek istediğiniz kullanıcı kimliğidir.
- passwd , özgün paroladır

3. Aşağıdaki komutu kullanarak özel taşıyıcı görüntünüzü oluşturun:

```
sudo docker build -t mymq .
```

Burada " . ", az önce yarattığınız iki dosyayı içeren dizindir.

Docker daha sonra bu görüntüyü kullanarak geçici bir taşıyıcı yaratır ve geri kalan komutları çalıştırır.

The **RUN** command adds a user named johndoe with password passwd and the **COPY** command adds the `config.mqsc` file into a specific location known by the parent image.

**Not:** Red Hat Enterprise Linux (RHEL) üzerinde, **docker** (RHEL V7) ya da **podman** (RHEL V7 ya da RHEL V8) komutunu kullanıyorsunuz. **podman** durumunda, komutun başında **sudo** ' a gereksinim duyarsınız.

4. Yeni oluşturduğunuz disk görüntüyle yeni bir kapsayıcı oluşturmak için yeni özelleştirilmiş görüntüyü çalıştırın.

Yeni görüntü katmanınız, çalıştırılacak belirli bir komutu belirtmemiş, bu nedenle üst görüntüden devralınmıştır. Üst ögenin giriş noktası (kod GitHub' ta kullanılabilir):

- Kuyruk yöneticisi yaratır
- Kuyruk yöneticisini başlatır
- Varsayılan bir dinleyici oluşturur

- Bundan sonra, /etc/mqm/config.mqsc.' den herhangi bir MQSC komutu çalıştırır

Yeni uyarlanmış görüntüyü çalıştırmak için aşağıdaki komutları verin:

```
sudo docker run \
  --env LICENSE=accept \
  --env MQ_QMGR_NAME=QM1 \
  --volume /var/example:/var/mqm \
  --publish 1414:1414 \
  --detach \
  mymq
```

Burada:

### İlk env parametresi

Bir ortam değişkenini kapsayıcıya geçirir. Bu, IBM IBM WebSphere MQ için lisansı kabul ettiğinizi kabul eder. Lisans değişkenini, lisansı görüntülemek için görüntülemek üzere de ayarlayabilirsiniz.

IBM MQ lisanslarına ilişkin ek ayrıntılar için [IBM MQ lisans bilgileri](#) başlıklı konuya bakın.

### İkinci env parametresi

Kullanmakta olduğunuz kuyruk yöneticisi adını belirler.

### Birim parametresi

Tells the container that whatever MQ writes to /var/mqm should actually be written to /var/example on the host.

Bu seçenek, taşıyıcıyı daha sonra kolayca silebileceğiniz ve kalıcı verileri sakladığınız anlamına gelir. Bu seçenek, günlük dosyalarının görüntülenmesini kolaylaştırır.

### Değiştirge yayınla

Anasistemdeki kapıları, taşıyıcıdaki kapılarla eşler. Taşıyıcı, varsayılan değer olarak kendi iç IP adresiyle çalıştırılır; bu da, göstermek istediğiniz herhangi bir kapıyı özellikle eşlememeniz gerektiğini gösterir.

Bu örnekte, bu, anasistemdeki 1414 numaralı kapının taşıyıcıdaki 1414 numaralı bağlantı noktasına eşlenmesini anlamına gelir.

### Parametreyi ayır

Taşıyıcıyı arka plandaki çalıştırır.

## Sonuçlar

Yapılandırılmış bir taşıyıcı görüntüsü oluşturdu ve docker **ps** komutunu kullanarak çalışan taşıyıcıları görüntüleyebiliyorsunuz. Docker **top** komutunu kullanarak taşıyıcılarınızda çalışan IBM MQ işlemlerini görüntüleyebilirsiniz.



### Uyarı:

Docker **logs** **\$\$CONTAINER\_ID** komutunu kullanarak bir taşıyıcının günlüklerini görüntüleyebilirsiniz.

## Sonraki adım

- Taşıyıcınız **ps** docker.komutunu kullandığınızda gösterilmiyorsa, taşıyıcı başarısız olabilir. Docker **ps -a** komutunu kullanarak başarısız olan taşıyıcıları görebilirsiniz.
- Docker **ps -a** komutunu kullandığınızda, taşıyıcı tanıtıcısı görüntülenir. Docker **run** komutunu verdiğinizde bu kimlik de yazdırılır.
- Docker **logs** **\$\$CONTAINER\_ID** komutunu kullanarak bir taşıyıcının günlüklerini görüntüleyebilirsiniz.
- You can set the maximum number of open files by using the command **sysctl fs.file-max=524288**.

## V 9.1.0 Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması

With process namespace sharing between containers in Docker, you can run applications that require a local binding connection to IBM MQ in separate containers from the IBM MQ queue manager.

### Bu görev hakkında

Bu işlevsellik IBM MQ 9.0.3 ve sonraki kuyruk yöneticilerinde desteklenir.

Aşağıdaki kısıtlamalara uymanız gerekir:

- --pid bağımsız değişkenini kullanarak, taşıyıcıların PID ad alanını paylaşmalısınız.
- You must share the containers IPC namespace using the --ipc argument.
- Aşağıdakilerden birini yapmak gerekir:
  1. Taşıyıcıların UTS ad alanını --uts bağımsız değişkenini kullanarak anasistemle paylaşır ya da
  2. Taşıyıcıların, -h ya da --hostname bağımsız değişkenini kullanarak aynı anasistem adına sahip olduğundan emin olun.
- IBM MQ veri dizinini, /var/mqm dizini altındaki tüm kapsayıcılar için kullanılabilir bir birime bağlamanız gerekir.

You can try this functionality out, by completing the following steps on a Linux system that already has Docker installed.

Aşağıdaki örnek, örnek IBM MQ kapsayıcı görüntüsünü kullanır. Bu resmin ayrıntılarını [Girhub](#)' te bulabilirsiniz.

### Yordam

1. Aşağıdaki komutu girerek, biriminiz olarak işlev görmeniz için geçici bir dizin oluşturun:

```
mkdir /tmp/dockerVolume
```

2. Create a queue manager (QM1) in a container, with the name sharedNamespace, by issuing the following command:

```
docker run -d -e LICENSE=accept -e MQ_QMGR_NAME=QM1 --volume /tmp/dockerVol:/mnt/mqm --uts host --name sharedNamespace ibmcom/mq
```

3. Start a second container called secondaryContainer, based off ibmcom/mq, but do not create a queue manager, by issuing the following command:

```
docker run --entrypoint /bin/bash --volumes-from sharedNamespace --pid container:sharedNamespace --ipc container:sharedNamespace --uts host --name secondaryContainer -it --detach ibmcom/mq
```

4. Her iki kuyruk yöneticisinin durumunu görmek için, ikinci taşıyıcıda **dspm** komutunu çalıştırın ve aşağıdaki komutu verin:

```
docker exec secondaryContainer dspmq
```

5. Diğer taşıyıcıda çalışan kuyruk yöneticisine yönelik MQSC komutlarını işlemek için aşağıdaki komutu çalıştırın:

```
docker exec -it secondaryContainer runmqsc QM1
```

## Sonuçlar

You now have local applications running in separate containers, and you can now successfully run commands like **dspmqr**, **amqspqr**, **amqsgqr**, and **runmqsc** as local bindings to the QM1 queue manager from the secondary container.

Beklediğiniz sonucu görmezseniz, daha fazla bilgi için bkz. [“Ad alanı uygulamalarınızda sorun giderme” sayfa 38](#).

### V9.1.0 Ad alanı uygulamalarınızda sorun giderme

Paylaşılan ad alanlarını kullanırken, tüm ad alanlarını (IPC, PID ve UTS/hostname) ve bağlı birimleri paylaştığınızdan emin olmanız gerekir, aksi takdirde uygulamalarınız çalışmaz.

İzlemeniz gereken kısıtlamaların listesi için bkz. [“Yerel bağ tanımlama uygulamalarının ayrı taşıyıcılarda çalıştırılması” sayfa 37](#).

Uygulamanız listelenen tüm kısıtlamaları karşılamıyorsa, taşıyıcının başladığı yerde sorunlarla karşılaşabilirsiniz, ancak beklediğiniz işlev çalışmaz.

Aşağıdaki listede bazı sık rastlanan nedenler ve kısıtlamalardan birini karşılamayı unutup unutmadığınızı göremediğiniz davranışlar yer alır.

- Ad alanını (UTS/PID/IPC) ya da taşıyıcıların anasistem adını paylaşmayı unutursanız ve birimi sisteme bağlarsanız, taşıyıcınız kuyruk yöneticisini görebilir, ancak kuyruk yöneticisiyle etkileşimde bulunamaz.
  - **dspmqr** komutları için aşağıdakileri görürsünüz:

```
docker exec container dspmqr
QMNAME(QM1)                STATUS(Status not available)
```

- **runmqsc** komutları ya da kuyruk yöneticisine bağlanmayı deneyen diğer komutlar için bir AMQ8146 hata iletisi alabilirsiniz:

```
docker exec -it container runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024.
Starting MQSC for queue manager QM1.
AMQ8146: IBM MQ queue manager not available
```

- Gerekli tüm ad alanlarını paylaşır, ancak `/var/mqm` dizinine bir paylaşılan birim bağlamazsanız ve geçerli bir IBM MQ veri yolunuz varsa, komutlarınız AMQ8146 hata iletilerini de alır.

Ancak **dspmqr**, kuyruk yöneticinizi hiç göremez ve bunun yerine boş bir yanıt döndürür:

```
docker exec container dspmqr
```

- Gerekli tüm ad alanlarını paylaşır, ancak `/var/mqm` dizinine paylaşılan bir birim bağlamazsanız ve geçerli bir IBM MQ veri yolunuz yoksa (ya da IBM MQ veri yolu yoksa), veri yolu bir IBM MQ kuruluşunun temel bileşeni olduğundan çeşitli hatalar görürsünüz. Veri yolu olmadan IBM MQ çalışmaz.

Aşağıdaki komutlardan herhangi birini çalıştırırsanız ve bu örneklerde gösterilenlere benzer yanıtlar görürseniz, dizini sisteme yerleştirdiğinizi ya da bir IBM MQ veri dizini oluşturduğunu doğrulamanız gerekir:

```
docker exec container dspmqr
'No such file or directory' from /var/mqm/mqs.ini
AMQ6090: IBM MQ was unable to display an error message FFFFFFFF.
AMQffff
```

```
docker exec container dspmqrver
AMQ7047: An unexpected error was encountered by a command. Reason code is 0.
```

```
docker exec container mqrc
<file path>/mqrc.c[1152]
lpiObtainQMDetails --> 545261715
```

```
docker exec container crtmqm QM1
AMQ8101: IBM MQ error (893) has occurred.

docker exec container strmqm QM1
AMQ6239: Permission denied attempting to access filesystem location '/var/mqm'.
AMQ7002: An error occurred manipulating a file.

docker exec container endmqm QM1
AMQ8101: IBM MQ error (893) has occurred.

docker exec container dlrmqm QM1
AMQ7002: An error occurred manipulating a file.

docker exec container strmqweb
<file path>/mqrc.c[1152]
lpiObtainQMDetails --> 545261715
```

## Linux > MQ Adv. > V 9.1.5 > CD **IBM MQ İşletmeni için API başvurusu**

IBM MQ , OpenShift Container Platform ile yerel bütünleştirme sağlayan bir Kubernetes Operator sağlar.

## Linux > MQ Adv. > V 9.1.5 > CD **mq.ibm.com/v1beta1 için API başvurusu**

v1beta1 API 'si, QueueManager kaynaklarını yaratmak ve yönetmek için kullanılabilir.

## Linux > MQ Adv. > V 9.1.5 > CD **mq.ibm.com/v1beta1 için lisans başvurusu**

spec.license.license alanı, kabul etmekte olduğunuz lisansa ilişkin lisans tanıtıcısını içermelidir. Geçerli değerler şunlardır:

Değer: spec.license.license	Değer: spec.license.use	Lisans bilgisi
L-RJON-BN7PN3	Production ya da NonProduction	<a href="#">IBM Cloud Pak for Integration 2020.2</a>
L-RJON-BPHL2Y		<a href="#">IBM Cloud Pak for Integration Sınırlı Sürüm 2020.2</a>
L-APIG-BJAKBF	Production ya da Development	<a href="#">IBM MQ Advanced 9.1.5</a>
L-APIG-BM7GDH	Development	<a href="#">IBM MQ Advanced for Developers 9.1.5</a>

Note that the license *sürüm* is specified, which is not always the same as the version of IBM MQ.

## Linux > MQ Adv. > V 9.1.5 > CD **QueueManager için API başvurusu (mq.ibm.com/v1beta1)**

### QueueManager

QueueManager , uygulamalara kuyruğa alma ve yayınlama/abone olma hizmetleri sağlayan bir IBM MQ sunucudur.

Alan	Tanım
apiVersion dizgi	APIVersion, bir nesnenin bu gösterimine ilişkin sürümü oluşturulan şemayı tanımlar. Sunucular, tanınan şemaları en son iç değere dönüştürmeli ve tanınmayan değerleri reddedebilir. Daha fazla bilgi: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#resources</a> .
kind dizgi	Tür, bu nesnenin temsil ettiği REST kaynağını temsil eden bir dizgi değeridir. Sunucular, bu durumu istemci göndereceği uç noktadan çıkartabilir. Güncellenemez. CamelCase içinde. Daha fazla bilgi: <a href="https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-çesit">https://git.k8s.io/community/contributors/devel/sig-architecture/api-conventions.md#types-çesit</a> .
metadata	
spec "QueueManagerBelirtimi" sayfa 44	QueueManager' un istenen durumu.
status "QueueManagerDurumu" sayfa 45	The observed state of the QueueManager.

## Kullanılabilirlik

Kuyruk yöneticisine ilişkin kullanılabilirlik ayarları; örneğin, etkin-bekleme çiftinin kullanılmasının da kullanılmayacağı gibi.

Burada görünür:

- "QueueManagerYapılanışı" sayfa 42

Alan	Tanım
type dizgi	Kullanılacak kullanılabilirliğin tipi. Otomatik olarak (bazı durumlarda) Kubernetes tarafından yeniden başlatılacak tek bir Pod için "SingleInstance" seçeneğini kullanın. Biri "etkin" Kuyruk Yöneticisi olan bir çift Pod çifti için "MultiInstance" ögesini kullanın; diğeri de yedekte bekleme durumuna sahip olur. See <a href="#">Kapsayıcılarda IBM MQ için yüksek kullanılabilirlik</a> in the latest version of IBM MQ.

## Lisans

Lisansı kabul ettiğinizi ve hangi lisans ölçümlerinin kullanılacağını denetleyen ayarlar.

Burada görünür:

- "QueueManagerBelirtimi" sayfa 44

Alan	Tanım
use dizgi	Lisansın birden çok kullanımla desteklendiği durumlarda yazılımın nasıl kullanılacağına ilişkin denetimler ayarlanıyor. Geçerli değerler için <a href="https://ibm.biz/BdqvCF">https://ibm.biz/BdqvCF</a> adresine bakın.
accept boole	Bu yazılımla ilişkili lisansı kabul edip etmediğiniz (gerekli).
license dizgi	Kabul ettiğiniz lisansın tanıtıcısı. Bu, kullanmakta olduğunuz MQ sürümüne ilişkin doğru lisans tanıtıcısı olmalıdır. Geçerli değerler için <a href="https://ibm.biz/BdqvCF">https://ibm.biz/BdqvCF</a> adresine bakın.



Alan	Tanım
metric dizgi	Hangi lisans metriğinin kullanılacağını belirten ayar. Örneğin, "ProcessorValueUnit", "VirtualProcessorCore" ya da "ManagedVirtualServer".

### Sınırlar

QueueManagerResourceList , CPU & bellek ayarlarını tanımlar.

Burada görünür:

- “Kaynaklar” sayfa 47

Alan	Tanım
cpu	
memory	

### LocalObjectBaşvurusu

LocalObjectReferansı, aynı ad alanı içinde başvuru nesneyi bulmanıza olanak verecek yeterli bilgiyi içerir.

Burada görünür:

- “QueueManagerBelirtimi” sayfa 44

Alan	Tanım
name dizgi	Başvurunun adı. Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names">https://kubernetes.io/docs/concepts/overview/working-with-objects/names/#names</a> TODO: Diğer yararlı alanları ekleyin. apiVersion, tür, uid?.

### PKI

Anahtarları ve sertifikaları İletim Arabirim Katmanı Güvenliği (TLS) ya da MQ Advanced Message Security (AMS) ile kullanmak üzere tanımlamak için genel anahtar altyapısı ayarları.

Burada görünür:

- “QueueManagerBelirtimi” sayfa 44

Alan	Tanım
keys “PKISource” sayfa 41 dizisi	Kuyruk Yöneticisi 'nin anahtar havuzuna eklemek için özel anahtarlar.
trust “PKISource” sayfa 41 dizisi	Kuyruk Yöneticisi 'nin anahtar havuzuna eklenecek sertifikalar.

### PKISource

PKISource, anahtarlar ya da sertifikalar gibi bir Genel Anahtar Altyapısı bilgileri kaynağı tanımlar.

Burada görünür:

- “PKI” sayfa 41

Alan	Tanım
name dizgi	Ad, anahtar ya da sertifika için etiket olarak kullanılır. Küçük harfli alfasayısal bir dizgi olmalıdır.

Alan	Tanım
secret <a href="#">“Güvenlik dizgisi”</a> sayfa 48	Kubernetes Güvenlik Dizgisini kullanarak bir anahtar sağlayın.

## QueueManagerYapılanışı

QueueManagerConfig, kuyruk yöneticisi taşıyıcısı ve temeldeki kuyruk yöneticisine ilişkin ayarları tanımlar.

Burada görünür:

- [“QueueManagerBelirtimi”](#) sayfa 44

Alan	Tanım
logFormat dizgi	Bu taşıyıcı için kullanılacak günlük biçimi. Taşıyıcıdaki JSON biçimli günlükler için "JSON" olanağını kullanın. Metin biçimli iletiler için "Temel" seçeneğini kullanın.
metrics <a href="#">“QueueManagerMetrikleri”</a> sayfa 43	Prometheusstili metriklerle ilişkin ayarlar.
readinessProbe <a href="#">“QueueManagerReadinessProbe”</a> sayfa 43	Hazır olma durumunu denetleyen ayarlar.
resources <a href="#">“Kaynaklar”</a> sayfa 47	Kaynak gereksinimlerini denetleyen ayarlar.
storage <a href="#">“QueueManagerSaklama Alanı”</a> sayfa 46	Kuyruk Yöneticisi 'nin Kalıcı Birimler ve Depolama Sınıfları kullanımını denetlemek için kullanılan depolama ayarları.
availability <a href="#">“Kullanılabilirlik”</a> sayfa 40	Kuyruk yöneticisine ilişkin kullanılabilirlik ayarları; örneğin, etkin-bekleme çiftinin kullanılmasının da kullanılmayacağı gibi.
imagePullPolicy dizgi	kubelet belirtilen görüntüyü çekmeye çalıştığında bu kontrollerin ayarlanması.
livenessProbe <a href="#">“QueueManagerLivenessProbe”</a> sayfa 42	Hızlılık sondasını denetleyen ayarlar.
debug boole	Hata ayıklama iletilerinin taşıyıcıya özel koddan, taşıyıcı günlüğüne kaydedilip kaydedilmeyeceğini belirleyin.
image dizgi	Kullanılacak taşıyıcı görüntüsü.
name dizgi	Temeldeki MQ Kuyruk Yöneticisi 'nin adı ( metadata.nameöğesinden farklıysa). Ad için Kubernetes kurallarına uymayan bir Kuyruk Yöneticisi adı istiyorsanız bu alanı kullanın (örneğin, temel harfler içeren bir ad).

## QueueManagerLivenessProbe

Hızlılık sondasını denetleyen ayarlar.

Burada görünür:

- [“QueueManagerYapılanışı”](#) sayfa 42

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı.

Alan	Tanım
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar">https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar</a> .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için?
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir.
timeoutSeconds tamsayı	Araştırıcı zamanaşımına geçtikten sonra geçecek süre (saniye). Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar">https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar</a> .

## QueueManagerMetrikleri

Prometheusstili metriklerle ilişkin ayarlar.

Burada görünür:

- [“QueueManagerYapılanışı” sayfa 42](#)

Alan	Tanım
enabled boole	Prometheusuyumlu bir metrik uç noktasının etkinleştirilip etkinleştirilmeyeceğini belirleyin.

## QueueManagerOptionalVolume

MQ kurtarma günlükleri içinPersistentVolume ayrıntıları. Çok eşgörünümli kuyruk yöneticisi kullanılırken gereklidir.

Burada görünür:

- [“QueueManagerSaklama Alanı” sayfa 46](#)

Alan	Tanım
class dizgi	Bu birim için kullanılacak depolama sınıfı. Yalnızca "type", "persistent-claim" ise geçerlidir.
enabled boole	Bu birimin ayrı bir birim olarak etkinleştirilip etkinleştirilmeyeceğini ya da varsayılan "queueManager" birimine yerleştirilip yerleştirilmeyeceğini belirleyin.
size dizgi	PersistentVolume (PersistentVolume) değeri Kubernetes' e geçer. Yalnızca "type", "persistent-claim" ise geçerlidir.
sizeLimit dizgi	"ephemeral" birimi kullanırken büyüklük sınırı. Dosyalar hala geçici bir dizine yazılıyor, bu nedenle boyutu sınırlandırmak için bu seçeneği kullanabilirsiniz. Yalnızca type "ephemeral" ise geçerlidir.
type dizgi	Kullanılacak birim tipi. Kalıcı olmayan bir "emptyDir" birimi yaratmak için ephemeral öğesini ya da kalıcı bir birimi kullanmak için persistent-claim öğesini seçin.

## QueueManagerReadinessProbe

Hazır olma durumunu denetleyen ayarlar.

Burada görünür:

- [“QueueManagerYapılanışı” sayfa 42](#)

Alan	Tanım
failureThreshold tamsayı	Araştırıcı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı.
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar">https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar</a> .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için?
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir.
timeoutSeconds tamsayı	Araştırıcı zamaşımına geçtikten sonra geçecek süre (saniye). Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar">https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar</a> .

## QueueManagerBelirtimi

QueueManager' un istenen durumu.

Burada görünür:

- [“QueueManager” sayfa 39](#)

Alan	Tanım
license <a href="#">“Lisans” sayfa 40</a>	Lisansı kabul ettiğinizi ve hangi lisans ölçümlerinin kullanılacağını denetleyen ayarlar.
pki <a href="#">“PKI” sayfa 41</a>	Anahtarları ve sertifikaları İletim Arabirim Katmanı Güvenliği (TLS) ya da MQ Advanced Message Security (AMS) ile kullanmak üzere tanımlamak için genel anahtar altyapısı ayarları.
queueManager <a href="#">“QueueManagerYapılanışı” sayfa 42</a>	QueueManagerConfig, kuyruk yöneticisi taşıyıcısı ve temeldeki kuyruk yöneticisine ilişkin ayarları tanımlar.
securityContext <a href="#">“SecurityContext” sayfa 48</a>	Kuyruk Yöneticisi Pod 'sına eklenecek güvenlik ayarları securityContext.
tracing <a href="#">“TracingConfig” sayfa 49</a>	Cloud Pak for Integration Operations Dashboard ile bütünleştirme bütünleştirmesine ilişkin ayarlar.
version dizgi	Kullanılacak MQ sürümünü denetleyen ayar (gerekli). Örneğin: "9.1.5.0-r2", taşıyıcı görüntüsünün ikinci düzeltmesini kullanarak MQ sürüm 9.1.5.0değerini belirtecektir. Taşıyıcıya özgü düzeltmeler, temel görüntüye ilişkin düzeltmeler gibi, sık sık düzeltmelere uygulanır.
affinity	Standart Kubernetes benzeşim kuralları. Daha fazla bilgi için bkz. <a href="https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#affinity-v1-core">https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#affinity-v1-core</a> .
imagePullSecrets <a href="#">“LocalObjectBaşvurusu” sayfa 41 dizisi</a>	Bu QueueManagertarafından kullanılan görüntülerin herhangi birini çekmek için kullanılacak aynı ad alanındaki sirlara yapılan başvuruların isteğe bağlı bir listesi. Belirtilirse, bu sirların kullanması için tek tek puller uygulamalarına geçilecek. Örneğin, docker durumunda, yalnızca DockerConfig tipi sirlar onurlandırılır. Daha fazla bilgi için bkz. <a href="https://kubernetes.io/docs/concepts/containers/images#specifying-imagepullsirlar-on-a-pod">https://kubernetes.io/docs/concepts/containers/images#specifying-imagepullsirlar-on-a-pod</a> .

Alan	Tanım
template <a href="#">“Şablon” sayfa 48</a>	Kubernetes kaynakları için gelişmiş şablon oluşturma. The template allows users to override how IBM MQ generates the underlying Kubernetes resources, such as StatefulSet, Pods and Services. Bu, yalnızca ileri düzey kullanıcılar içindir; yanlış kullanıldığında MQ ' nun olağan çalışmasını kesintiye uğratabilecek bir potansiyeldir. QueueManager kaynağındaki başka bir yerde belirlenen değerler, şablondaki ayarlarla geçersiz kılınır.
terminationGracePeriod Seconds tamsayı	İsteğe bağlı olarak, Pod 'un zarafetle sonlandırması gereken saniye cinsinden süre. Değer eksi olmayan bir tamsayı olmalıdır. Sıfır değeri, silmeyi hemen gösterir. Kuyruk yöneticisinin sona erdirilmesinin denendiği hedef zaman, uygulama bağlantısını kesme aşamaları üst seviyeye iletilmeye çalışılır. Gerekliyse, önemli kuyruk yöneticisi bakım görevleri kesintiye uğratılır.
web <a href="#">“WebServerYapılandırması” sayfa 50</a>	MQ web sunucusuna ilişkin ayarlar.

### QueueManagerDurumu

The observed state of the QueueManager.

Burada görünür:

- [“QueueManager” sayfa 39](#)

Alan	Tanım
endpoints <a href="#">“QueueManagerStatusEndpoint” sayfa 46</a> dizisi	Bu Kuyruk Yöneticinin gösterileceği uç noktalarla ilgili bilgiler (API ya da UI uç noktaları gibi).
name dizgi	Kuyruk Yöneticisi 'nin adı.
versions <a href="#">“QueueManagerStatusVersion” sayfa 46</a>	Kullanılmakta olan MQ sürümü ve IBM Yetkili Kayıt Defterinden sağlanan diğer sürümler.
adminUiUrl dizgi	Yönetici UI 'si için URL.
conditions <a href="#">“QueueManagerStatusCondition” sayfa 45</a> dizisi	Koşullar, Kuyruk Yöneticisi 'nin durumuna ilişkin en son kullanılabilir gözlemlenmeleri gösterir.

### QueueManagerStatusCondition

QueueManagerStatusCondition , Kuyruk Yöneticisinin koşullarını tanımlar.

Burada görünür:

- [“QueueManagerDurumu” sayfa 45](#)

Alan	Tanım
message dizgi	Son geçişle ilgili ayrıntıları belirten insan tarafından okunabilir ileti.
type dizgi	Koşul tipi.
lastTransitionTime dizgi	Koşul bir durumdan diğerine geçişte son kez.

## QueueManagerStatusEndpoint

QueueManagerStatusEndpoint , QueueManager' a ilişkin uç noktaları tanımlar.

Burada görünür:

- [“QueueManagerDurumu” sayfa 45](#)

Alan	Tanım
name dizgi	Uç noktenin adı.
type dizgi	Bir UI uç noktası için 'UI' gibi, bir API uç noktası için 'API', API belgeleri için 'OpenAPI' gibi bir uç noktana ilişkin tip.
uri dizgi	Uç noktana ilişkin URI.

## QueueManagerStatusVersion

Kullanılmakta olan MQ sürümü ve IBM Yetkili Kayıt Defterinden sağlanan diğer sürümler.

Burada görünür:

- [“QueueManagerDurumu” sayfa 45](#)

Alan	Tanım
available <a href="#">“QueueManagerStatusVersion Kullanılabilir” sayfa 46</a>	Diğer MQ sürümleri IBM Yetkili Kayıt Defterinden kullanılabilir.
reconciled dizgi	Kullanılmakta olan IBM MQ sürümü. Özel bir resim belirtilirse, bu, gerçekte kullanılmakta olan MQ sürümüyle eşleşmeyebilir.

## QueueManagerStatusVersionKullanılabilir

Diğer MQ sürümleri IBM Yetkili Kayıt Defterinden kullanılabilir.

Burada görünür:

- [“QueueManagerStatusVersion” sayfa 46](#)

Alan	Tanım
channels dizi	MQ sürümünü otomatik olarak güncelleştirmek için kullanılacak kanallar.
versions <a href="#">“Sürümler” sayfa 50</a> dizisi	MQ ' ın belirli sürümleri kullanılabilir.

## QueueManagerSaklama Alanı

Kuyruk Yöneticisi 'nin Kalıcı Birimler ve Depolama Sınıfları kullanımını denetlemek için kullanılan depolama ayarları.

Burada görünür:

- [“QueueManagerYapılanışı” sayfa 42](#)

Alan	Tanım
persistedData <a href="#">“QueueManagerOptionalVolume” sayfa 43</a>	Yapılandırma, kuyruklar ve iletiler de içinde olmak üzere MQ kalıcı verileri için Persistent Volume ayrıntıları. Çok eşgörünümlü kuyruk yöneticisi kullanılırken gereklidir.

Alan	Tanım
queueManager <a href="#">“QueueManagerBirimi” sayfa 47</a>	/var/mqmaltındaki normalde herhangi bir veri için varsayılan PersistentVolume . Başka bir birim belirtilmediyse, tüm kalıcı verileri ve kurtarma günlüklerini içerir.
recoveryLogs <a href="#">“QueueManagerOptionalVolume” sayfa 43</a>	MQ kurtarma günlükleri için PersistentVolume ayrıntıları. Çok eşgözümlü kuyruk yöneticisi kullanılırken gereklidir.

## QueueManagerBirimi

/var/mqmaltındaki normalde herhangi bir veri için varsayılan PersistentVolume . Başka bir birim belirtilmediyse, tüm kalıcı verileri ve kurtarma günlüklerini içerir.

Burada görünür:

- [“QueueManagerSaklama Alanı” sayfa 46](#)

Alan	Tanım
class dizgi	Bu birim için kullanılacak depolama sınıfı. Yalnızca "type", "persistent-claim" ise geçerlidir.
size dizgi	PersistentVolume (PersistentVolume) değeri Kubernetes' e geçer. Yalnızca "type", "persistent-claim" ise geçerlidir.
sizeLimit dizgi	"ephemeral" birimi kullanırken büyüklük sınırı. Dosyalar hala geçici bir dizine yazılıyor, bu nedenle boyutu sınırlandırmak için bu seçeneği kullanabilirsiniz. Yalnızca type "ephemeral" ise geçerlidir.
type dizgi	Kullanılacak birim tipi. Kalıcı olmayan bir "emptyDir" birimi yaratmak için ephemeral ögesini ya da kalıcı bir birimi kullanmak için persistent-claim ögesini seçin.

## İstekler

QueueManagerResourceList , CPU & bellek ayarlarını tanımlar.

Burada görünür:

- [“Kaynaklar” sayfa 47](#)

Alan	Tanım
memory	
cpu	

## Kaynaklar

Kaynak gereksinimlerini denetleyen ayarlar.

Burada görünür:

- [“QueueManagerYapılanışı” sayfa 42](#)

Alan	Tanım
limits <a href="#">“Sınırlar” sayfa 41</a>	QueueManagerResourceList , CPU & bellek ayarlarını tanımlar.
requests <a href="#">“İstekler” sayfa 47</a>	QueueManagerResourceList , CPU & bellek ayarlarını tanımlar.

## Güvenlik dizgisi

Kubernetes Güvenlik Dizgisini kullanarak bir anahtar sağlayın.

Burada görünür:

- “PKISource” sayfa 41

Alan	Tanım
items dizi	Keys inside the Kubernetes secret which should be added to the Queue Manager container.
secretName dizgi	Kubernetes gizli adının adı.

## SecurityContext

Kuyruk Yöneticisi Pod 'sına eklenecek güvenlik ayarları securityContext.

Burada görünür:

- “QueueManagerBelirtimi” sayfa 44

Alan	Tanım
supplementalGroups dizi	Taşıyıcının birincil GID ' ye ek olarak, her taşıyıcıda çalıştırılan ilk işlem için uygulanan grupların listesi. Belirtilmemişse, herhangi bir taşıyıcıya grup eklenmez.
fsGroup tamsayı	Bir kaptaki tüm kaplar için geçerli olan özel bir tamamlayıcı grup. Bazı birim tipleri, Kubelet 'in bu birimin sahipliğini kapsül sahibi olacak şekilde değiştirmesine olanak tanır: 1. İye GID, FSGroup 2 olarak kabul edilebilir. setgid biti ayarlanır (birimde oluşturulan yeni dosyalar FSGroup 'a aittir) 3. İzin bitleri ya da rw-rw ile 'd' dir ---- Eğer ayarlanmazsa, Kubelet herhangi bir birimin sahipliğini ve izinlerini değiştirmeyecektir.
initVolumeAsRoot boole	Bu, PersistentVolume'ı kullanıma hazırlayan taşıyıcı tarafından kullanılan securityContext ' ı etkiler. Yeni sağlanan birimlere erişmek için kök kullanıcı olmanız gereken bir depolama sağlayıcısı kullanıyorsanız, bu ayarı "true" olarak ayarlayın. Bunu "true" olarak ayarlamak, hangi Güvenlik Bağlamı Kısıtlamaları (SCC) nesnesini kullanabildiğinizi etkiler ve kök kullanıcıya izin veren SCC ' yi kullanma yetkiniz yoksa, Kuyruk Yöneticisi başlatılamayabilir. Daha fazla bilgi için bkz. <a href="https://docs.openshift.com/container-platform/latest/authentication/managing-security-context-constraints.html">https://docs.openshift.com/container-platform/latest/authentication/managing-security-context-constraints.html</a> .

## Şablon

Kubernetes kaynakları için gelişmiş şablon oluşturma. The template allows users to override how IBM MQ generates the underlying Kubernetes resources, such as StatefulSet, Pods and Services. Bu, yalnızca ileri düzey kullanıcılar içindir; yanlış kullanıldığında MQ ' nun olağan çalışmasını kesintiye uğratabilecek bir potansiyeldir. QueueManager kaynağındaki başka bir yerde belirlenen değerler, şablondaki ayarlarla geçersiz kılınır.

Burada görünür:

- “QueueManagerBelirtimi” sayfa 44

Alan	Tanım
pod	Pod için kullanılan şablona ilişkin geçersiz kılmalar. Bkz. <a href="https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#podspec-v1-core">https://kubernetes.io/docs/reference/generated/kubernetes-api/v1.17/#podspec-v1-core</a> .



## TracingAgent

Yalnızca Cloud Pak for Integration ' da isteğe bağlı İzleme Aracısı ayarlarını yapılandırabilirsiniz.

Burada görünür:

- [“TracingConfig” sayfa 49](#)

Alan	Tanım
image dizgi	Kullanılacak taşıyıcı görüntüsü.
imagePullPolicy dizgi	kubelet belirtilen görüntüyü çekmeye çalışıldığında bu kontrollerin ayarlanması.
livenessProbe <a href="#">“TracingProbe” sayfa 49</a>	Hızlılık sondasını denetleyen ayarlar.
readinessProbe <a href="#">“TracingProbe” sayfa 49</a>	Hazır olma durumunu denetleyen ayarlar.

## TracingCollector

Yalnızca Cloud Pak for Integration içinde isteğe bağlı Tracing Collector ile ilgili ayarları yapılandırabilirsiniz.

Burada görünür:

- [“TracingConfig” sayfa 49](#)

Alan	Tanım
image dizgi	Kullanılacak taşıyıcı görüntüsü.
imagePullPolicy dizgi	kubelet belirtilen görüntüyü çekmeye çalışıldığında bu kontrollerin ayarlanması.
livenessProbe <a href="#">“TracingProbe” sayfa 49</a>	Hızlılık sondasını denetleyen ayarlar.
readinessProbe <a href="#">“TracingProbe” sayfa 49</a>	Hazır olma durumunu denetleyen ayarlar.

## TracingConfig

Cloud Pak for Integration Operations Dashboard ile bütünleştirme bütünleştirmesine ilişkin ayarlar.

Burada görünür:

- [“QueueManagerBelirtimi” sayfa 44](#)

Alan	Tanım
agent <a href="#">“TracingAgent” sayfa 49</a>	Yalnızca Cloud Pak for Integration ' da isteğe bağlı İzleme Aracısı ayarlarını yapılandırabilirsiniz.
collector <a href="#">“TracingCollector” sayfa 49</a>	Yalnızca Cloud Pak for Integration içinde isteğe bağlı Tracing Collector ile ilgili ayarları yapılandırabilirsiniz.
enabled boole	İzleme aracılığıyla Cloud Pak for Integration Operations Dashboard ile bütünleştirmenin etkinleştirilip etkinleştirilmeyeceğini belirleyin.
namespace dizgi	Cloud Pak for Integration Operations Dashboard olanağının kurulu olduğu ad alanı.

## TracingProbe

Hazır olma durumunu denetleyen ayarlar.

Burada görünür:

- [“TracingCollector” sayfa 49](#)

Alan	Tanım
failureThreshold tamsayı	Araştırmacı başarılı olduktan sonra başarısız olarak değerlendirilecek art arda başarısızlıklar alt sınırı.
initialDelaySeconds tamsayı	Hızlılık sondaları başlatılmadan önce başlatıldıktan sonra geçecek süre (saniye). Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar">https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar</a> .
periodSeconds tamsayı	Sondayı (saniye cinsinden) ne sıklıkta (saniye cinsinden) gerçekleştirmek için?
successThreshold tamsayı	Sondanın başarısız olduktan sonra başarılı sayılması için arka arkaya en az başarı elde edilmesi gerekir.
timeoutSeconds tamsayı	Araştırmacı zamanaşımına geçtikten sonra geçecek süre (saniye). Daha fazla bilgi: <a href="https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar">https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle#container-problar</a> .

## Sürümler

QueueManagerStatusVersion , MQ' un bir sürümünü tanımlar.

Burada görünür:

- [“QueueManagerStatusVersionKullanılabilir” sayfa 46](#)

Alan	Tanım
name dizgi	QueueManagerprogramının bu sürümüne ilişkin "name" sürümü. Bunlar, spec . version alanı için geçerli değerlerdir.

## WebServerYapılandırması

MQ web sunucusuna ilişkin ayarlar.

Burada görünür:

- [“QueueManagerBelirtimi” sayfa 44](#)

Alan	Tanım
enabled boole	Web sunucusunun etkinleştirilip etkinleştirilmeyeceği.

## 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 uyuşmadığı 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 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](http://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 ülkelerdeki tescilli ticari markasıdır.

Bu ürün, Eclipse Project (<http://www.eclipse.org/>) tarafından geliştirilen yazılımları içerir.

Java ve Java tabanlı tüm markalar ve logolar, Oracle firmasının ve/ya da iştiraklerinin markaları ya da tescilli markalarıdır.







Parça numarası:

(1P) P/N: