9.0

IBM MQ 구성

참고

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, 721 페이지의 『주의사항』에 있는 정보를 확인하십시오.

이 개정판은 새 개정판에 별도로 명시하지 않는 한, IBM® MQ 의 버전 9릴리스 0및 모든 후속 릴리스와 수정에 적용됩니다.

목차

IBM MQ 구성

하나 이상의 컴퓨터에서 하나 이상의 큐 관리자를 작성하고 개발, 테스트, 프로덕션 시스템에서 해당 큐 관리자가 비즈니스 데이터를 포함하는 메시지를 처리하도록 구성합니다.

IBM MQ을(를) 구성하기 전에 IBM MQ 기술 개요의 IBM MQ 개념에 대해 읽으십시오. 계획에서 IBM MQ 환경을 계획하는 방법을 읽어보십시오.

IBM MQ에서 큐 관리자와 관련 자원을 작성, 구성, 관리하기 위해 다양한 방법을 사용할 수 있습니다. 이러한 방법에는 명령행 인터페이스, 그래픽 사용자 인터페이스, 관리 API가 포함됩니다. 이 인터페이스에 대한 자세한 정보는 IBM MQ 관리를 참조하십시오.

큐 관리자 작성, 시작, 중지, 삭제 방법에 대한 지시사항은 5 페이지의 『Multiplatforms에서 큐 관리자 작성 및 관리』의 내용을 참조하십시오.

IBM MQ 설치 및 애플리케이션을 함께 연결하는 데 필요한 컴포넌트를 작성하는 방법에 대한 정보는 128 페이지의 『분산 큐잉 구성』의 내용을 참조하십시오.

다른 방법을 사용하여 IBM MQ 서버에 클라이언트를 연결하는 방법에 대한 지시사항은 14 페이지의 『서버와 클라이언트 간의 연결 구성』의 내용을 참조하십시오.

큐 관리자 클러스터를 구성하는 방법에 대한 지시사항은 216 페이지의 『큐 관리자 클러스터 구성』의 내용을 참조하십시오.

구성 정보를 변경하여 IBM MQ 또는 큐 관리자의 동작을 변경할 수 있습니다. 자세한 정보는 78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』의 내용을 참조하십시오. 이 제품 문서에 명시된 경우를 제외하고는 일반적으로 구성 변경사항을 적용하기 위해 큐 관리자를 재시작하지 않아도 됩니다.

z/OS IBM MQ for z/OS® 구성 방법에 대한 지시사항은 592 페이지의 『z/OS에서 큐 관리자 구성』의 내용을 참조하십시오.

관련 태스크

592 페이지의 『z/OS에서 큐 관리자 구성』
IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

관련 정보

IBM MQ 기술 개요
로컬 IBM MQ 오브젝트 관리
원격 IBM MQ 오브젝트 관리
IBM i IBMi 관리
z/OS IBM MQ for z/OS 관리
계획 중
z/OS z/OS에서 IBM MQ 환경 계획

Multi Multiplatforms에서 큐 관리자 작성 및 관리

메시지 및 큐를 사용하려면 먼저 최소한 하나의 큐 관리자 및 연관된 오브젝트를 작성하고 시작해야 합니다. 큐 관리자는 연관된 자원, 특히 큐에서 소유하는 자원을 관리합니다. 큐 관리자는 IBM MQ 오브젝트 작성, 수정, 표시, 삭제를 위한 메시지 큐잉 인터페이스(MQI) 호출 및 명령에 대한 큐잉 서비스를 애플리케이션에 제공합니다.

시작하기 전에

중요사항: IBM MQ는 공백을 포함하는 시스템 이름을 지원하지 않습니다. 공백을 포함하는 시스템 이름으로 컴퓨터에 IBM MQ를 설치하면 큐 관리자를 작성할 수 없습니다.

큐 관리자를 작성하기 전에 특히 프로덕션 환경에서 몇 가지 사항을 고려해야 합니다. 다음 체크리스트를 검토하십시오.

큐 관리자와 연관된 설치

큐 관리자를 작성하려면 IBM MQ 제어 명령 **crtmqm**을 사용하십시오. **crtmqm** 명령은 큐 관리자를 **crtmqm** 명령이 실행된 설치와 자동으로 연관시킵니다. 큐 관리자에 작용하는 명령의 경우, 큐 관리자와 연관된 설치에서 명령을 실행해야 합니다. setmqm 명령을 사용하여 큐 관리자의 연관된 설치를 변경할 수 있습니다. Windows 설치 프로그램은 mqm 그룹에 대한 설치를 수행하는 사용자를 추가하지 않습니다. 자세한 내용은 UNIX, Linux®, and Windows 에서 IBM MQ 를 관리하는 권한의 내용을 참조하십시오.

이름 지정 규칙

모든 플랫폼에서 큐 관리자와 통신할 수 있도록 대문자 이름을 사용하십시오. 이름은 입력한 그대로 지정됩니다. 많은 입력에 따른 불편을 방지하려면 불필요하게 긴 이름을 사용하지 마십시오.

고유 큐 관리자 이름 지정

큐 관리자를 작성할 경우 다른 큐 관리자가 네트워크의 임의 위치에서 동일한 이름을 사용하지 않아야 합니다. 큐 관리자를 작성할 때 큐 관리자 이름을 검사하지 않으므로, 고유하지 않은 이름을 사용하면 분산 큐잉 목적으로 채널을 작성할 수 없습니다. 또한 발행/구독 메시징을 위해 네트워크를 사용하는 경우, 구독은 구독을 작성한 큐 관리자 이름과 연관됩니다. 따라서 클러스터 또는 계층의 큐 관리자 이름이 같으면 발행이 큐 관리자에 도달할 수 없습니다.

고유하게 하는 한 가지 방법은 각 큐 관리자 이름에 자체의 고유한 노드 이름으로 접두부를 붙이는 것입니다. 예를 들어 노드가 `ACCOUNTS`, 큐 관리자 이름을 `ACCOUNTS.SATURN.QUEUE.MANAGER`로 지정할 수 있습니다. 여기서 `SATURN`은 특정 큐 관리자를 식별하고 `QUEUE.MANAGER`는 모든 큐 관리자에 지정할 수 있는 확장자입니다. 이를 생략할 수 있기는 하지만 `ACCOUNTS.SATURN` 및 `ACCOUNTS.SATURN.QUEUE.MANAGER`는 다른 큐 관리자 이름인 점을 참고하십시오.

다른 엔터프라이즈와의 통신을 위해 IBM MQ를 사용하는 경우에는 사용자 고유의 엔터프라이즈 이름을 접두부로 포함할 수도 있습니다. 이 작업은 수행 방법이 더 어려워 예에서는 다루지 않습니다.

참고: 제어 명령의 큐 관리자 이름은 대소문자를 구분합니다. 이는 이름 `jupiter.queue.manager` 및 `JUPITER.queue.manager`로 두 개의 큐 관리자를 작성할 수 있음을 의미합니다. 그러나 이러한 조합은 피하는 것이 좋습니다.

큐 관리자 수 제한

자원이 허용하는 한 많은 큐 관리자를 작성할 수 있습니다. 그러나 각 큐 관리자에서는 고유한 자원이 필요하므로 일반적으로 10개 큐 관리자에서 각각 10개 큐를 갖는 것보다 노드에서 하나의 큐 관리자가 100개 큐를 갖는 것이 좋습니다.

프로덕션 시스템에서는 단일 큐 관리자로 많은 프로세서를 이용할 수 있지만 큐 관리자가 많아 서버 시스템이 클수록 보다 효과적으로 실행될 수 있습니다.

기본 큐 관리자 지정

각 노드마다 기본 큐 관리자가 있어야 하지만 기본 큐 관리자 없이 노드에서 IBM MQ를 구성할 수도 있습니다. 기본 큐 관리자는 애플리케이션이 `MQCONN` 호출에서 큐 관리자 이름을 지정하지 않는 경우 연결하는 큐 관리자입니다. 또한 큐 관리자 이름을 지정하지 않고 `runmqsc` 명령을 호출할 때 MQSC 명령을 처리하는 큐 관리자이기도 합니다.

큐 관리자를 기본값으로 지정하면 노드에 대한 기존의 기본 큐 관리자 스펙이 바뀝니다.

기본 큐 관리자를 변경하면 다른 사용자 또는 애플리케이션에 영향을 줄 수 있습니다. 현재 연결된 애플리케이션은 추가 MQI 호출에서 원래 연결 호출의 핸들을 사용하므로 변경에 따른 영향을 받지 않습니다. 이 핸들을 사용하면 동일한 큐 관리자에 호출이 전달됩니다. 기본 큐 관리자를 변경한 후 연결하는 모든 애플리케이션은 새 기본 큐 관리자에 연결됩니다. 원하는 결과일지라도 기본값을 변경하기 전에 이를 고려해야 합니다.

기본 큐 관리자 작성은 8 페이지의 『기본 큐 관리자 작성』에서 설명됩니다.

데드-레터 큐 지정

데드-레터 큐는 메시지가 지정된 목적지로 라우트될 수 없을 때 메시지를 넣는 로컬 큐입니다.

네트워크의 각 큐 관리자에 데드-레터 큐를 두어야 합니다. 이 큐를 정의하지 않으면 애플리케이션 프로그램의 오류로 인해 채널이 닫히고 관리 명령에 대한 응답이 수신되지 않을 수 있습니다.

예를 들어 애플리케이션이 다른 큐 관리자의 큐에 메시지를 넣으려고 하지만 큐 이름을 잘못 입력하면 채널은 중지되고 메시지는 전송 큐에 그대로 남습니다. 그러면 다른 애플리케이션이 해당 메시지에 이 채널을 사용할 수 없게 됩니다.

큐 관리자에 데드-레터 큐가 있으면 채널이 영향을 받지 않습니다. 미전달 메시지는 수신측 데드-레터 큐에 놓이고 채널과 해당 전송 큐를 계속 사용할 수 있습니다.

큐 관리자를 작성하는 경우 **-u** 플래그를 사용하여 데드-레터 큐의 이름을 지정하십시오. MQSC 명령을 사용하여 사용될 데드-레터 큐를 지정하기 위해 이미 정의한 큐 관리자의 속성을 대체할 수도 있습니다. MQSC 명령 ALTER의 예는 큐 관리자에 대한 작업을 참조하십시오.

기본 전송 큐 지정

전송 큐는 리모트 큐 관리자로 전송 중인 메시지가 전송 이전에 큐잉되는 로컬 큐입니다. 기본 전송 큐는 명시적으로 정의된 전송 큐가 없을 때 사용되는 큐입니다. 각 큐 관리자에 기본 전송 큐가 지정될 수 있습니다.

큐 관리자를 작성하는 경우 **-d** 플래그를 사용하여 기본 전송 큐의 이름을 지정하십시오. 실제로 큐를 작성하는 것은 아니며 나중에 명시적으로 수행해야 합니다. 자세한 정보는 로컬 큐에 대한 작업을 참조하십시오.

필요한 로깅 매개변수 지정

로깅 유형과 로그 파일의 경로 및 크기를 포함하는 로깅 매개변수를 `crtmqm` 명령에 지정할 수 있습니다.

개발 환경에서는 기본 로깅 매개변수가 적절해야 합니다. 그러나 예를 들어 다음과 같은 경우 기본값을 변경할 수 있습니다.

- 대용량 로그를 지원할 수 없는 로우엔드 시스템 구성입니다.
- 큐에 동시에 많은 긴 메시지가 있는 것으로 예상합니다.
- 큐 관리자를 통해 많은 지속 메시지가 전달되는 것으로 예상합니다.

로깅 매개변수를 설정한 경우, 일부 매개변수는 큐 관리자를 삭제하고 이름은 같지만 다른 로깅 매개변수로 재작성하는 방법으로만 변경할 수 있습니다.

매개변수 로깅에 대한 자세한 정보는 361 페이지의 『고가용성, 복구 및 재시작 구성』의 내용을 참조하십시오.

UNIX **IBM MQ for UNIX 시스템만 해당**

crtmqm 명령을 사용하기 전에 별도 로컬 파일 시스템에서도 큐 관리자 디렉토리 /var/mqm/qmgrs/*qmgr*을 작성할 수 있습니다. **crtmqm**을 사용할 때, mqm에 소유된 빈 /var/mqm/qmgrs/*qmgr* 디렉토리가 존재할 경우 이 디렉토리를 큐 관리자 데이터에 사용합니다. 디렉토리를 mqm이 소유하지 않는 경우에는 First Failure Support Technology (FFST) 메시지와 함께 작성이 실패합니다. 디렉토리가 비어 있지 않으면 새 디렉토리가 작성됩니다.

이 태스크 정보

큐 관리자를 작성하려면 IBM MQ 제어 명령 **crtmqm**을 사용하십시오. 자세한 정보는 **crtmqm**의 내용을 참조하십시오. **crtmqm** 명령은 필요한 기본 오브젝트와 시스템 오브젝트를 자동으로 작성합니다(시스템 기본 오브젝트 참조). 기본 오브젝트는 사용자가 작성하는 오브젝트 정의의 기반을 형성합니다. 큐 관리자 조작에는 시스템 오브젝트가 필요합니다.

Windows Windows 시스템에서는 **crtmqm** 명령의 *sax* 옵션을 사용하여 큐 관리자의 여러 인스턴스를 시작할 수 있습니다.

큐 관리자 및 해당 오브젝트를 작성한 경우 **strmqm** 명령을 사용하여 큐 관리자를 시작할 수 있습니다.

프로시저

- 큐 관리자 작성 및 관리에 도움이 되는 정보는 다음 하위 주제를 참조하십시오.

  - 8 페이지의 『기본 큐 관리자 작성』
  - 9 페이지의 『기존 큐 관리자를 기본값으로 만들기』
  - 10 페이지의 『큐 관리자 작성 후 구성 파일 백업』
  - 10 페이지의 『큐 관리자 시작』
  - 11 페이지의 『큐 관리자 중지』
  - 13 페이지의 『큐 관리자 재시작』

– 13 페이지의 『큐 관리자 삭제』

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

592 페이지의 『z/OS에서 큐 관리자 구성』
IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

관련 정보

QM1 큐 관리자 작성
시스템 및 기본 오브젝트
crtmqm

Multi 기본 큐 관리자 작성

기본 큐 관리자는 애플리케이션이 MQCONN 호출에 큐 관리자 이름을 지정하지 않은 경우에 연결하는 큐 관리자입니다. 또한 큐 관리자 이름을 지정하지 않고 **runmqsc** 명령을 호출할 때 MQSC 명령을 처리하는 큐 관리자이기도 합니다. 큐 관리자를 작성하려면 IBM MQ 제어 명령 **crtmqm**을 사용하십시오.

시작하기 전에

기본 큐 관리자를 작성하기 전에 5 페이지의 『Multiplatforms에서 큐 관리자 작성 및 관리』에 설명된 고려사항을 읽으십시오.

UNIX **crtmqm**을 사용하여 UNIX에 큐 관리자를 작성할 때, mqm이 소유한 빈 /var/mqm/qmgrs/*qmgr* 디렉토리가 있을 경우 이 디렉토리가 큐 관리자 데이터에 사용됩니다. 이 디렉토리를 mqm이 소유하고 있지 않은 경우에는 First Failure Support Technology(FFST) 메시지가 표시되면서 큐 관리자 작성이 실패합니다. 디렉토리가 비어 있지 않은 경우 큐 관리자 데이터에 사용할 새 디렉토리가 작성됩니다.

이 고려사항은 /var/mqm/qmgrs/*qmgr* 디렉토리가 별도의 로컬 파일 시스템에 있는 경우에도 적용됩니다.

이 태스크 정보

crtmqm 명령을 사용하여 큐 관리자를 작성하면 명령이 자동으로 필수 기본 오브젝트 및 시스템 오브젝트를 작성합니다. 기본 오브젝트는 사용자가 작성하는 오브젝트 정의의 기반이 되고, 시스템 오브젝트는 큐 관리자 조작에 필요합니다.

관련 매개변수를 명령에 포함시켜 큐 관리자가 사용할 기본 전송 큐의 이름 및 데드-레터 큐의 이름 등을 정의할 수도 있습니다.

Windows Windows에서 **crtmqm** 명령의 **sax** 옵션을 사용하여 큐 관리자의 다중 인스턴스를 시작할 수 있습니다.

crtmqm 명령 및 해당 구문에 대한 자세한 정보는 **crtmqm**의 내용을 참조하십시오.

프로시저

- 기본 큐 관리자를 작성하려면 **crtmqm** 명령을 **-q** 플래그와 함께 사용하십시오.

  crtmqm 명령의 다음 예에서는 SATURN.QUEUE.MANAGER라는 기본 큐 관리자를 작성합니다.

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

  설명:

  -q
  이 큐 관리자가 기본 큐 관리자임을 나타냅니다.

  -d MY.DEFAULT.XMIT.QUEUE
  이 큐 관리자가 사용할 기본 전송 큐의 이름입니다.

참고: IBM MQ는 기본 전송 큐를 자동으로 작성하지 않습니다. 사용자가 직접 정의해야 합니다.

-u SYSTEM.DEAD.LETTER.QUEUE
IBM MQ가 설치에 작성하는 기본 데드-레터 큐의 이름입니다.

SATURN.QUEUE.MANAGER
큐 관리자의 이름입니다. 이 이름은 `crtmqm` 명령에서 지정되는 마지막 매개변수여야 합니다.

다음에 수행할 작업

큐 관리자 및 해당 오브젝트를 작성한 경우 **strmqm** 명령을 사용하여 큐 관리자를 시작하십시오.

관련 태스크

10 페이지의 『큐 관리자 작성 후 구성 파일 백업』
IBM MQ 구성 정보는 UNIX, Linux, and Windows의 구성 파일에 저장됩니다. 큐 관리자를 작성한 후 구성 파일을 백업하십시오. 그런 다음 문제점을 초래하는 다른 큐 관리자를 작성하는 경우 문제점의 원인을 제거할 때 백업을 복원할 수 있습니다.

관련 정보

큐 관리자에 대한 작업
로컬 큐에 대한 작업
시스템 및 기본 오브젝트

Multi 기존 큐 관리자를 기본값으로 만들기

Windows 및 Linux에서 텍스트 편집기를 사용하거나 IBM MQ Explorer를 사용하여 수동으로 기존 큐 관리자를 기본 큐 관리자로 만들 수 있습니다.

이 태스크 정보

텍스트 편집기를 사용하여 기존 큐 관리자를 기본 큐 관리자로 만들려면 다음 단계를 수행하십시오.

Windows Linux Windows 및 Linux(x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 이렇게 변경할 경우 9 페이지의 『IBM MQ Explorer를 사용하여 큐 관리자를 기본으로 설정』의 내용을 참조하십시오.

기본 큐 관리자를 작성할 때 해당 이름이 IBM MQ 구성 파일 (`mqs.ini`) 에 있는 `DefaultQueueManager` 스탠자의 `Name` 속성에 삽입됩니다. 스탠자 및 해당 컨텐츠가 없으면 자동으로 작성됩니다.

프로시저

- 기존 큐 관리자를 기본 큐 관리자로 만들려면 `Name` 속성의 큐 관리자 이름을 새 기본 큐 관리자의 이름으로 변경하십시오. 이 작업은 텍스트 편집기를 사용하여 수동으로 수행할 수 있습니다.
- 노드에 기본 큐 관리자가 없고 기존 큐 관리자를 기본 큐 관리자로 만들려면 필요한 이름으로 *`DefaultQueueManager`* 스탠자를 작성하십시오.
- 잘못해서 다른 큐 관리자를 기본 큐 관리자로 만들었거나 원래 기본 큐 관리자로 되돌리려면 `mqs.ini` 파일의 `DefaultQueueManager` 스탠자를 편집하여 원하지 않는 기본 큐 관리자를 원하는 기본 큐 관리자로 바꾸십시오.

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

Windows Linux IBM MQ Explorer를 사용하여 큐 관리자를 기본으로 설정

Windows 및 Linux(x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 기존 큐 관리자를 기본 큐 관리자로 만들 수 있습니다.

이 태스크 정보

Windows 및 Linux(x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 기존 큐 관리자를 기본 큐 관리자로 만들려면 다음 단계를 수행하십시오.

텍스트 편집기를 사용하여 수동으로 변경하려면 9 페이지의 『기존 큐 관리자를 기본값으로 만들기』의 내용을 참조하십시오.

프로시저

1. IBM MQ Explorer를 여십시오.
2. **IBM WebSphere MQ**를 마우스의 오른쪽 단추로 클릭한 후 **특성...**을 선택하십시오. **IBM MQ의 특성** 패널이 표시됩니다.
3. **기본 큐 관리자 이름** 필드에 기본 큐 관리자의 이름을 입력하십시오.
4. **확인**을 클릭하십시오.

ULW 큐 관리자 작성 후 구성 파일 백업

IBM MQ 구성 정보는 UNIX, Linux, and Windows의 구성 파일에 저장됩니다. 큐 관리자를 작성한 후 구성 파일을 백업하십시오. 그런 다음 문제점을 초래하는 다른 큐 관리자를 작성하는 경우 문제점의 원인을 제거할 때 백업을 복원할 수 있습니다.

이 태스크 정보

일반적으로 새 큐 관리자를 작성할 때마다 구성 파일을 백업합니다.

구성 파일은 다음과 같은 두 가지 유형이 있습니다.

- 제품을 설치하면 IBM MQ 구성 파일(`mqs.ini`)이 작성됩니다. 이 파일에는 큐 관리자를 작성 또는 삭제할 때마다 업데이트되는 큐 관리자 목록이 포함됩니다. 노드당 하나의 `mqs.ini` 파일이 있습니다.
- 새 큐 관리자를 작성하면 새 큐 관리자 구성 파일(`qm.ini`)이 자동으로 작성됩니다. 이 파일에는 큐 관리자에 대한 구성 매개변수가 포함됩니다.

V 9.0.0 AMQP 서비스를 설치한 경우 다음 추가 구성 파일을 백업해야 합니다.

- Windows Windows 시스템 `amqp_win.properties`
- Linux UNIX UNIX 및 Linux 시스템: `amqp_unix.properties`

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

501 페이지의 『IBM MQ 큐 관리자 데이터 백업 및 복원』
큐 관리자 및 큐 관리자 데이터를 백업하고 큐 관리자 구성만 백업하며 백업 큐 관리자를 사용하는 방식으로 하드웨어 고장으로 인해 발생 가능한 손상으로부터 큐 관리자를 보고할 수 있습니다.

Multi 큐 관리자 시작

큐 관리자를 작성할 때 명령 또는 MQI 호출을 처리하려면 큐 관리자를 시작해야 합니다.

이 태스크 정보

`strmqm` 명령을 사용하여 큐 관리자를 시작할 수 있습니다. **`strmqm`** 명령 및 옵션에 대한 설명은 strmqm을 참조하십시오.

Windows Linux 또는 Windows 및 Linux(x86 및 x86-64 플랫폼) 시스템의 경우 IBM MQ Explorer를 사용하여 큐 관리자를 시작할 수 있습니다.

Windows Windows의 경우 IBM MQ Explorer를 사용하여 시스템이 시작될 때 자동으로 큐 관리자를 시작할 수 있습니다. 자세한 정보는 IBM MQ Explorer를 사용한 관리를 참조하십시오.

프로시저

- **strmqm** 명령을 사용하여 큐 관리자를 시작하려면 시작하려는 큐 관리자의 이름을 이 명령 뒤에 함께 입력하십시오.
  예를 들어, QMB라는 큐 관리자를 시작하려면 다음 명령을 입력하십시오.

  ```
  strmqm QMB
  ```

  참고: strmqm 명령은 작업을 수행하는 큐 관리자와 연관된 설치에서 사용해야 합니다. `dspmq -o installation` 명령을 사용하여 큐 관리자가 연관된 설치를 찾을 수 있습니다.

  `strmqm` 명령은 큐 관리자가 시작되어 연결 요청을 승인할 수 있을 때까지 제어를 리턴하지 않습니다.
- Windows Linux
  IBM MQ Explorer를 사용하여 큐 관리자를 시작하려면 다음의 단계를 완료하십시오.
  a) IBM MQ Explorer를 여십시오.
  b) 네비게이터 보기에서 큐 관리자를 선택하십시오.
  c) **시작**을 누르십시오.

결과

큐 관리자가 시작됩니다.

큐 관리자 시작을 위해 몇 초 이상이 소요되는 경우 IBM MQ는 시작 진행 상황을 나타내는 정보 메시지를 간헐적으로 실행합니다.

Multi 큐 관리자 중지

endmqm 명령을 사용하여 큐 관리자를 중지할 수 있습니다. 이 명령은 세 가지 방법 즉, 제어된 종료 또는 정상 종료(Quiesced shutdown), 즉시 종료, 강제 종료(Preemptive shutdown)를 사용하여 큐 관리자를 중지합니다. 또는 Windows 및 Linux에서 IBM MQ Explorer를 사용하여 큐 관리자를 중지할 수 있습니다.

이 태스크 정보

endmqm 명령으로 단일 인스턴스 큐 관리자를 중지하는 방법에는 다음 세 가지가 있습니다.

제어된(또는 정상) 종료
: 기본적으로 **endmqm** 명령은 지정된 큐 관리자의 정상 종료(Quiesced shutdown)를 수행합니다. 정상 종료는 연결된 모든 애플리케이션이 끊어질 때까지 대기합니다. 따라서 완료하는 데 다소 시간이 걸릴 수 있습니다.

즉시 종료
: 즉시 종료의 경우, 현재 MQI 호출을 모두 완료할 수 있지만 새 호출은 실패합니다. 이 종료 유형은 애플리케이션이 큐 관리자로부터 연결이 끊어질 때까지 대기하지 않습니다.

강제 종료(preemptive shutdown)
: 큐 관리자가 즉시 중지됩니다. 큐 관리자가 정상적인 **endmqm** 명령으로 중지되지 않는 경우와 같이 예외적인 경우에만 이 유형의 종료를 사용하십시오.

endmqm 명령을 실행하면 다중 인스턴스 큐 관리자의 모든 인스턴스가 단일 인스턴스 큐 관리자가 중지될 때와 동일한 방식으로 중지됩니다. **endmqm** 명령은 다중 인스턴스 큐 관리자의 대기 인스턴스 중 하나 또는 활성 인스턴스에서 실행할 수 있습니다. 그러나 큐 관리자를 종료하려면 활성 인스턴스에서 **endmqm**을 실행해야 합니다.

endmqm 명령 및 해당 옵션에 대한 자세한 설명은 endmqm을 참조하십시오.

팁: 큐 관리자를 종료하는 중에 발생하는 문제점은 대개 애플리케이션 때문입니다. 예를 들어, 애플리케이션이 다음과 같을 경우,

- MQI 리턴 코드를 올바르게 검사하지 않는 경우
- 일시정지 알림을 요청하지 않는 경우
- 큐 관리자 연결을 끊지 않고 종료하는 경우(MQDISC 호출 발행)

큐 관리자를 중지하려고 할 때 문제점이 발생하면 Ctrl-C를 사용하여 **endmqm** 명령을 중단할 수 있습니다. 그런 다음 다른 **endmqm** 명령을 실행할 수 있지만, 이번에는 필요한 종료의 유형을 지정하는 매개변수를 사용하여 실행할 수 있습니다.

Windows Linux **endmqm** 명령을 사용하는 대신 Windows 및 Linux에서 IBM MQ Explorer 를 사용하여 제어 또는 즉시 종료를 수행하여 큐 관리자를 중지할 수 있습니다.

프로시저

- **endmqm** 명령을 사용하여 큐 관리자를 중지하려면 매개변수(필요한 경우)와 시작하려는 큐 관리자의 이름을 이 명령 뒤에 함께 입력하십시오.

  참고: 작업 중인 큐 관리자와 연관된 설치에서 **endmqm** 명령을 사용해야 합니다. 큐 관리자와 연관된 설치를 찾으려면 `dspmq -o installation` 명령을 사용하십시오.

  - 제어된(정상) 종료를 수행하려면 다음 예에 표시된 것과 같이, **endmqm** 명령을 입력하십시오. 그러면 QMB 라는 큐 관리자가 중지됩니다.

    ```
    endmqm QMB
    ```

    또는 다음 예에 표시된 것과 같이, **endmqm** 명령을 **-c** 매개변수와 함께 입력하는 것은 `endmqm QMB` 명령과 같습니다.

    ```
    endmqm -c QMB
    ```

    두 경우 모두, 제어 권한이 즉시 사용자에게 리턴되므로, 큐 관리자가 중지될 때 이에 대한 알림이 사용자에게 통보되지 않습니다. 제어를 리턴하기 전에 모든 애플리케이션이 중지되고 큐 관리자가 종료될 때까지 명령이 대기하도록 하려면 다음 예제에 표시된 대로 **-w** 매개변수를 사용하십시오.

    ```
    endmqm -w QMB
    ```

  - 즉시 종료를 수행하려면 다음 예에 표시된 것과 같이 **endmqm** 명령을 **-i** 매개변수와 함께 입력하십시오.

    ```
    endmqm -i QMB
    ```

  - 강제 종료(preemptive shutdown)를 수행하려면 다음 예에 표시된 것과 같이 **endmqm** 명령을 **-p** 매개변수와 함께 입력하십시오.

    ```
    endmqm -p QMB
    ```

    주의: 강제 종료를 수행할 경우 연결된 애플리케이션에서 예측할 수 없는 결과가 초래될 수 있습니다. 정상적인 **endmqm** 명령을 사용하여 큐 관리자를 중지하려는 다른 모든 시도가 실패할 때까지는 이 옵션을 사용하지 마십시오. ULW 강제 종료가 작동하지 않을 경우 수동으로 큐 관리자 중지를 대신 시도하십시오.

  - 자동 클라이언트 다시 연결을 요청하려면 **endmqm** 명령을 `-r` 매개변수와 함께 입력하십시오. 이 매개변수는 클라이언트를 큐 관리자 그룹의 다른 큐 관리자에 다시 연결하는 데 영향을 미칩니다.

    참고: 기본 **endmqm** 명령을 사용하여 큐 관리자를 종료하면 자동 클라이언트 다시 연결을 트리거하지 않습니다.

  - 활성 인스턴스를 종료한 후 다중 인스턴스 큐 관리자의 대기 인스턴스로 전송하려면 다중 인스턴스 큐 관리자의 활성 인스턴스에서 **endmqm** 명령을 `-s` 매개변수와 함께 입력하십시오.

- 다중 인스턴스 큐 관리자의 대기 인스턴스는 종료하고 활성 인스턴스는 실행 중인 상태로 유지하려면 다중 인스턴스 큐 관리자의 대기 인스턴스에서 **endmqm** 명령을 `-x` 매개변수와 함께 입력하십시오.

- Windows Linux
  Windows 및 Linux에서 IBM MQ Explorer를 사용하여 큐 관리자를 중지하려면 다음 단계를 완료하십시오.
  a) IBM MQ Explorer를 여십시오.
  b) 네비게이터 보기에서 큐 관리자를 선택하십시오.
  c) **중지**를 누르십시오.
  큐 관리자 종료 패널이 표시됩니다.
  d) **제어됨** 또는 **즉시**를 선택하십시오.
  e) **확인**을 클릭하십시오.
  큐 관리자가 중지됩니다.

관련 정보
유지보수 레벨 업데이트를 Windows의 다중 인스턴스 큐 관리자에 적용
유지보수 레벨 업데이트를 UNIX 및 Linux의 다중 인스턴스 큐 관리자에 적용

Multi 큐 관리자 재시작

strmqm 명령을 사용하여 큐 관리자를 재시작할 수 있으며 Windows 및 Linux x86-64 시스템의 경우 IBM MQ Explorer에서 큐 관리자를 재시작할 수 있습니다.

이 태스크 정보

strmqm 명령을 사용하여 큐 관리자를 재시작할 수 있습니다. **strmqm** 명령 및 해당 옵션에 대한 설명은 strmqm을 참조하십시오.

Windows Linux Windows 및 Linux x86-64 시스템의 경우 큐 관리자를 시작할 때와 동일한 방식으로 IBM MQ Explorer를 사용하여 큐 관리자를 재시작할 수 있습니다.

프로시저

- **strmqm** 명령을 사용하여 큐 관리자를 재시작하려면 재시작하려는 큐 관리자의 이름을 이 명령 뒤에 함께 입력하십시오.
  예를 들어, `strmqm saturn.queue.manager`라는 큐 관리자를 시작하려면 다음 명령을 입력하십시오.

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

- Windows Linux
  IBM MQ Explorer를 사용하여 큐 관리자를 시작하려면 다음의 단계를 완료하십시오.
  a) IBM MQ Explorer를 여십시오.
  b) 네비게이터 보기에서 큐 관리자를 선택하십시오.
  c) **시작**을 클릭하십시오.

결과

큐 관리자가 재시작됩니다.

큐 관리자 재시작을 위해 몇 초 이상이 소요되는 경우 IBM MQ는 시작 진행 상황을 나타내는 정보 메시지를 간헐적으로 실행합니다.

Multi 큐 관리자 삭제

dltmqm 명령을 사용하여 큐 관리자를 삭제할 수 있습니다. 또는 Windows 및 Linux 시스템의 경우 IBM MQ Explorer를 사용하여 큐 관리자를 삭제할 수 있습니다.

시작하기 전에

주의:

- 큐 관리자 삭제는 극단적인 조치입니다. 모든 큐와 해당 메시지 및 모든 오브젝트 정의를 포함하여 큐 관리자와 연관된 모든 자원이 삭제되기 때문입니다. **dltmqm** 명령을 사용하는 경우, 작업을 취소할 수 있는 프롬프트가 표시되지 않습니다. Enter 키를 누르면 연관된 자원을 모두 잃게 됩니다.
- Windows Windows의 경우 큐 관리자를 삭제하면 큐 관리자가 자동 시작 목록에서도 제거됩니다(10 페이지의 『큐 관리자 시작』에 설명되어 있음). 명령이 완료되면 `IBM MQ queue manager ending` 메시지가 표시됩니다. 큐 관리자가 삭제되었음을 알리지 않습니다.
- 클러스터 큐 관리자를 삭제해도 클러스터에서는 제거되지 않습니다. 자세한 정보는 dltmqm에 있는 사용 참고사항을 참조하십시오.

이 태스크 정보

dltmqm 명령을 사용하여 큐 관리자를 삭제할 수 있습니다. **dltmqm** 명령 및 해당 옵션에 대한 설명은 dltmqm을 참조하십시오. 이 명령은 신뢰할 수 있는 관리자만 사용할 수 있습니다. (보안에 대한 정보는 UNIX, Linux, and Windows에서 보안 설정을 참조하십시오.)

Windows Linux 또는 Windows 및 Linux(x86 및 x86-64 플랫폼) 시스템의 경우 IBM MQ Explorer를 사용하여 큐 관리자를 삭제할 수 있습니다.

프로시저

- **dltmqm** 명령을 사용하여 큐 관리자를 삭제하려면 다음의 단계를 완료하십시오.
  a) 큐 관리자를 중지합니다.
  b) 다음 명령을 실행하십시오.

  ```
  dltmqm QMB
  ```

  참고: 작업 중인 큐 관리자와 연관된 설치에서 **dltmqm** 명령을 사용해야 합니다. `dspmq -o installation` 명령을 사용하여 큐 관리자가 연관된 설치를 찾을 수 있습니다.
- Windows Linux
  IBM MQ Explorer를 사용하여 큐 관리자를 삭제하려면 다음의 단계를 완료하십시오.
  a) IBM MQ Explorer를 여십시오.
  b) 네비게이터 보기에서 큐 관리자를 선택하십시오.
  c) 큐 관리자가 중지되지 않은 경우 중지하십시오.
  큐 관리자를 중지하려면 큐 관리자를 마우스 오른쪽 단추로 클릭한 후 **중지**를 클릭하십시오.
  d) 큐 관리자를 삭제합니다.
  큐 관리자를 삭제하려면 큐 관리자를 마우스 오른쪽 단추로 클릭한 후 **삭제**를 클릭하십시오.

결과

큐 관리자가 삭제됩니다.

서버와 클라이언트 간의 연결 구성

IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

이 태스크 정보

IBM MQ에서 오브젝트 간 논리 통신 링크를 채널이라고 합니다. IBM MQ MQI clients를 서버에 연결하는 데 사용되는 채널을 MQI 채널이라고 합니다. IBM MQ MQI client의 IBM MQ 애플리케이션이 서버의 큐 관리자와 통신할 수 있도록 링크의 각 끝에서 채널 정의를 설정합니다.

MQI 채널을 정의하기 전에 사용할 통신 양식을 결정하고 채널의 각 측에 연결을 정의해야 합니다.

프로시저

1. 사용할 통신 양식을 결정하십시오.
   15 페이지의 『사용할 통신 유형』 를 참조하십시오.
2. 채널의 각 측에 연결을 정의하십시오.
   연결을 정의하려면 다음을 수행해야 합니다.
   a) 연결을 구성합니다.
   b) 채널 정의에 필요한 매개변수의 값을 기록합니다.
   c) 리스너를 시작하여 서버가 IBM MQ MQI client에서 수신되는 네트워크 요청을 감지할 수 있게 합니다.

관련 개념

17 페이지의 『확장된 트랜잭션 클라이언트 구성』
이 주제 콜렉션에서는 각 트랜잭션 관리자 범주에 대해 확장된 트랜잭션 기능을 구성하는 방법을 설명합니다.

27 페이지의 『MQI 채널 정의』
새 채널 정의를 작성하려면 같은 채널 이름과 호환 가능한 채널 유형을 사용하여 연결의 양 끝에 하나씩 **두 개** 채널 정의를 작성해야 합니다. 이 경우 채널 유형은 *server-connection* 및 *client-connection*입니다.

43 페이지의 『MQI 채널에 대한 채널 엑시트 프로그램』
UNIX, Linux, and Windows의 IBM MQ MQI client 환경에서는 세 가지 유형의 채널 엑시트를 사용할 수 있습니다.

관련 태스크

33 페이지의 『다른 플랫폼에서 서버 연결 및 클라이언트 연결 정의 작성』
적용할 컴퓨터에서 각 채널 정의를 작성할 수 있습니다. 그러나 클라이언트 컴퓨터에서 채널 정의를 작성하는 방법에는 제한이 있습니다.

36 페이지의 『서버에서 서버 연결 및 클라이언트 연결 정의 작성』
서버에서 두 정의 모두 작성한 다음 클라이언트가 클라이언트 연결 정의를 사용할 수 있도록 설정할 수 있습니다.

47 페이지의 『클라이언트를 큐 공유 그룹에 연결』
큐 공유 그룹의 멤버인 서버에서 클라이언트와 큐 관리자 간 MQI 채널을 작성하여 클라이언트를 큐 공유 그룹에 연결할 수 있습니다.

48 페이지의 『구성 파일을 사용하여 클라이언트 구성』
텍스트 파일의 속성을 사용하여 클라이언트를 구성할 수 있습니다. 이러한 속성은 환경 변수 또는 다른 플랫폼별 방식으로 대체될 수 있습니다.

관련 정보

큐 관리자에 IBM MQ MQI 클라이언트 애플리케이션 연결
DISPLAY CHLAUTH
SET CHLAUTH

사용할 통신 유형

플랫폼에 따라 다른 통신 프로토콜을 지원합니다. 전송 프로토콜 선택은 IBM MQ MQI client 및 서버 플랫폼의 조합에 따라 다릅니다.

MQI 채널의 전송 프로토콜 유형

클라이언트 및 서버 플랫폼에 따라 MQI 채널의 전송 프로토콜 유형은 최대 4개입니다.

- TCP/IP
- LU 6.2
- NetBIOS
- SPX

MQI 채널을 정의할 때 각 채널 정의는 전송 프로토콜(전송 유형) 속성을 지정해야 합니다. 서버는 한 개의 프로토콜로 제한되지 않으므로 채널 정의에 따라 다른 프로토콜을 지정할 수 있습니다. IBM MQ MQI clients의 경우 여러 전송 프로토콜을 사용하는 대체 MQI 채널이 있는 것이 유용할 수 있습니다.

또한 전송 프로토콜 선택은 IBM MQ 클라이언트와 서버 플랫폼의 특정 조합에 따라서 달라집니다. 가능한 조합은 다음 테이블에 표시되어 있습니다.

표 *1. 전송 프로토콜 - IBM MQ MQI client*와 서버 플랫폼의 조합

전송 프로토콜	**IBM MQ MQI client**	**IBM MQ 서버**
TCP/IP	IBM i UNIX Windows	IBM i UNIX Windows z/OS
LU 6.2	UNIX [1] Windows	IBM i UNIX [1] Windows z/OS
NetBIOS	Windows	Windows
SPX	Windows	Windows

참고:

1. Linux(POWER platform) 제외

관련 개념

185 페이지의 『Windows에서 TCP 연결 정의』
대상의 주소를 지정하기 위해 송신측에서 채널을 구성하고 수신측에서 리스너 프로그램을 실행하여 TCP 연결을 정의합니다.

192 페이지의 『UNIX and Linux에서 TCP 연결 정의』
송신측의 채널 정의는 대상의 주소를 지정합니다. 리스너 또는 inet 디먼은 수신측에서의 연결용으로 구성됩니다.

209 페이지의 『IBM i에서 TCP 연결 정의』
연결 이름 필드를 사용하여 채널 정의에서 TCP 연결을 정의할 수 있습니다.

676 페이지의 『z/OS에서 TCP 연결 정의』
TCP 연결을 정의하려면 여러 설정을 구성해야 합니다.

187 페이지의 『Windows에서 LU 6.2 연결 정의』
SNA는 두 시스템 간에 LU 6.2 대화가 설정될 수 있도록 구성되어야 합니다.

195 페이지의 『UNIX and Linux에서 LU 6.2 연결 정의』
SNA는 두 시스템 간에 LU 6.2 대화가 설정될 수 있도록 구성되어야 합니다.

211 페이지의 『IBM i에서 LU 6.2 연결 정의』
완전한 LU 6.2 연결의 모드 이름, TP 이름 및 연결 이름을 사용하여 LU 6.2 통신 세부사항을 정의합니다.

188 페이지의 『Windows에서 NetBIOS 연결 정의』

NetBIOS 연결은 Windows를 실행하는 클라이언트 및 서버에만 적용됩니다. IBM MQ는 다른 IBM MQ 제품에 대한 NetBIOS 연결을 설정할 때 세 가지 유형의 NetBIOS 자원(세션, 명령 이름)을 사용합니다. 이러한 자원은 각각 NetBIOS 설치 중 선택적으로 또는 기본적으로 설정되는 한계가 있습니다.

관련 참조

17 페이지의 『TCP/IP 연결 한계』
단일 TCP/IP 포트에 큐 대기할 수 있는 미해결 연결 요청의 수는 플랫폼에 따라 다릅니다. 한계에 도달하면 오류가 발생합니다.

678 페이지의 『APPC/MVS를 사용하여 z/OS에 대한 LU 6.2 연결 정의』
LU6.2 연결을 정의하려면 여러 설정을 구성해야 합니다.

TCP/IP 연결 한계

단일 TCP/IP 포트에 큐 대기할 수 있는 미해결 연결 요청의 수는 플랫폼에 따라 다릅니다. 한계에 도달하면 오류가 발생합니다.

이 연결 한계는 IBM MQ 서버에 접속할 수 있는 최대 클라이언트 수와 같지 않습니다. 서버 시스템 자원에 의해 판별되는 레벨까지 서버에 추가 클라이언트를 연결할 수 있습니다. 연결 요청에 대한 백로그 값이 다음 테이블에 표시되어 있습니다.

표 2. *TCP/IP* 포트에서 큐잉되는 최대 미해결 연결 요청 수

서버 플랫폼	최대 연결 요청 수
AIX AIX®	100
HP-UX HP-UX	20
Linux Linux	100
IBM i IBM i	255임)
Solaris Solaris	100
Windows Windows Server	100
Windows Windows Workstation	100
z/OS z/OS	255임)

연결 한계에 도달하면 클라이언트는 `MQCONN` 호출에서 MQRC_HOST_NOT_AVAILABLE 리턴 코드를 수신하고 클라이언트 오류 로그의 AMQ9202 오류 (UNIX and Linux 시스템의 경우 `/var/mqm/errors/AMQERROn.LOG` 또는 Windows 의 IBM MQ 클라이언트 설치의 오류 서브디렉토리에 있는 `amqerr0n.log`) 를 수신합니다. 클라이언트가 `MQCONN` 요청을 재시도하면 성공할 수도 있습니다.

수행할 수 있는 연결 요청의 수를 늘리고 이 제한으로 인해 오류가 생성되는 것을 피하기 위해 각각 다른 포트를 청취하는 다중 리스너를 보유하거나 둘 이상의 큐 관리자를 둘 수 있습니다.

확장된 트랜잭션 클라이언트 구성

이 주제 콜렉션에서는 각 트랜잭션 관리자 범주에 대해 확장된 트랜잭션 기능을 구성하는 방법을 설명합니다.

각 플랫폼에서 확장된 트랜잭션 클라이언트는 다음 외부 트랜잭션 관리자에 대한 지원을 제공합니다.

XA 준수 트랜잭션 관리자
: 확장된 트랜잭션 클라이언트는 CICS® 및 Tuxedo와 같은 XA 준수 트랜잭션 관리자를 지원하기 위한 XA 자원 관리자 인터페이스를 제공합니다.

Microsoft 트랜잭션 서버(Windows 시스템 전용)
: Windows 시스템에서만 XA 자원 관리자 인터페이스가 MTS(Microsoft Transaction Server)도 지원합니다. 확장된 트랜잭션 클라이언트에서 제공되는 IBM MQ MTS 지원은 MTS와 XA 자원 관리자 인터페이스 간 브릿지를 제공합니다.

WebSphere® Application Server
: 이전 버전의 IBM WebSphere MQ에서는 WebSphere Application Server 4 또는 5를 지원했고 확장 트랜잭션 클라이언트를 사용하기 위해 특정 구성 태스크를 수행해야 했습니다. WebSphere Application Server 6 이상에는 IBM WebSphere MQ 또는 IBM MQ 메시징 제공자가 포함되므로 확장 트랜잭션 클라이언트를 사용할 필요가 없습니다.

관련 개념

18 페이지의 『XA 준수 트랜잭션 관리자 구성』
먼저 IBM MQ 기본 클라이언트를 구성한 후 다음 주제의 정보를 사용하여 확장된 트랜잭션 기능을 구성합니다.

26 페이지의 『MTS(Microsoft Transaction Server)』
MTS를 트랜잭션 관리자로 사용하기 전에 필요한 추가 구성은 없습니다. 그러나 몇 가지 참고해야 할 점이 있습니다.

XA 준수 트랜잭션 관리자 구성

먼저 IBM MQ 기본 클라이언트를 구성한 후 다음 주제의 정보를 사용하여 확장된 트랜잭션 기능을 구성합니다.

참고: 이 절에서는 The Open Group에 의해 발행된 분산 트랜잭션 처리*: XA* 스펙에 나와 있는 XA 인터페이스에 대한 기본적 이해가 있는 것으로 가정합니다.

확장 트랜잭션 클라이언트를 구성하려면 다음에 설명된 대로 먼저 IBM MQ 기본 클라이언트를 구성해야 합니다.

- AIX AIX에서 IBM MQ 클라이언트 설치
- HP-UX HP-UX에서 IBM MQ 클라이언트 설치
- Linux Linux에서 IBM MQ 클라이언트 설치
- Solaris Solaris에서 IBM MQ 클라이언트 설치
- Windows Windows에서 IBM MQ 클라이언트 설치
- IBM i IBM i에서 IBM MQ 클라이언트 설치

플랫폼에 해당하는 정보를 사용하여 CICS 및 Tuxedo와 같은 XA 준수 트랜잭션 관리자에 대해 확장된 트랜잭션 기능을 구성할 수 있습니다.

트랜잭션 관리자는 큐 관리자에 연결되는 클라이언트 애플리케이션이 사용하는 것과 동일한 MQI 채널을 사용하는 자원 관리자로 큐 관리자와 통신합니다. 트랜잭션 관리자가 자원 관리자(xa_) 함수 호출을 발행하면 큐 관리자에 대한 호출을 전달하고 큐 관리자로부터 다시 출력을 수신하기 위해 MQI 채널이 사용됩니다.

트랜잭션 관리자가 큐 관리자를 자원 관리자로 열기 위해 xa_open 호출을 발행하여 MQI 채널을 시작하거나 클라이언트 애플리케이션이 MQCONN 또는 MQCONNX 호출을 발행하여 MQI 채널을 시작할 수 있습니다.

- 트랜잭션 관리자가 MQI 채널을 시작하고 같은 스레드에서 클라이언트 애플리케이션이 나중에 MQCONN 또는 MQCONNX를 호출하는 경우 MQCONN 또는 MQCONNX 호출이 성공적으로 완료되고 연결 핸들이 애플리케이션에 리턴됩니다. 애플리케이션은 MQRC_ALREADY_CONNECTED 이유 코드와 함께 MQCC_WARNING 완료 코드를 수신하지 않습니다.
- 클라이언트 애플리케이션이 MQI 채널을 시작하고 같은 스레드에서 트랜잭션 관리자가 나중에 xa_open을 호출하는 경우 xa_open 호출이 이 MQI 채널을 사용하는 큐 관리자에게 전달됩니다.

실패 후 복구 상황에서 실행 중인 클라이언트 애플리케이션이 없는 경우 트랜잭션 관리자는 전용 MQI 채널을 사용하여 실패 시 큐 관리자가 참여했던 불완전한 작업 단위를 복구할 수 있습니다.

확장된 트랜잭션 클라이언트를 XA 준수 트랜잭션 관리자와 함께 사용하는 경우 다음 조건을 참고하십시오.

- 단일 스레드 내에서 클라이언트 애플리케이션은 한 번에 한 큐 관리자에만 연결할 수 있습니다. 이 제한은 확장된 트랜잭션 클라이언트를 사용하는 경우에만 적용되며 IBM MQ 기본 클라이언트를 사용하는 클라이언트 애플리케이션은 단일 스레드 내에서 둘 이상의 큐 관리자에 동시에 연결할 수 있습니다.

- 클라이언트 애플리케이션의 각 스레드는 서로 다른 큐 관리자에 연결할 수 있습니다.
- 클라이언트 애플리케이션은 공유 연결 핸들을 사용할 수 없습니다.

확장된 트랜잭션 기능을 구성하려면 자원 관리자 역할을 하는 각 큐 관리자에 대한 다음 정보를 트랜잭션 관리자에게 제공해야 합니다.

- xa_open 문자열
- XA 스위치 구조에 대한 포인터

큐 관리자를 자원 관리자로 열기 위해 트랜잭션 관리자가 xa_open을 호출하면 호출 시 인수 *xa_info*로 xa_open 문자열이 확장된 트랜잭션 클라이언트에 전달됩니다. 확장된 트랜잭션 클라이언트는 다음과 같은 방식으로 xa_open 문자열의 정보를 사용합니다.

- 클라이언트 애플리케이션이 아직 MQI 채널을 시작하지 않은 경우 서버 큐 관리자에 대한 MQI 채널을 시작함
- 트랜잭션 관리자가 자원 관리자로 여는 큐 관리자와 클라이언트 애플리케이션이 연결하는 큐 관리자와 같은지 확인함
- 큐 관리자가 동적 등록을 사용하는 경우 트랜잭션 관리자의 ax_reg 및 ax_unreg 함수를 찾음

xa_open 문자열의 형식 및 확장된 트랜잭션 클라이언트가 xa_open 문자열의 정보를 사용하는 방법에 대한 자세한 정보는 20 페이지의 『xa_open 문자열의 형식』의 내용을 참조하십시오.

XA 스위치 구조는 트랜잭션 관리자가 확장된 트랜잭션 클라이언트가 제공한 xa_ 함수를 찾을 수 있게 하고 큐 관리자가 동적 등록을 사용하는지 여부를 지정합니다. 확장된 트랜잭션 클라이언트에 제공되는 XA 스위치 구조에 대한 정보는 23 페이지의 『XA 스위치 구조』의 내용을 참조하십시오.

특정 트랜잭션 관리자에 대해 확장된 트랜잭션 기능을 구성하는 방법에 대한 정보 및 확장된 트랜잭션 클라이언트에서 트랜잭션 관리자를 사용하는 것에 대한 그 외 정보는 다음 절을 참조하십시오.

- 25 페이지의 『CICS를 위한 확장된 트랜잭션 클라이언트 구성』
- 26 페이지의 『Tuxedo를 위한 확장된 트랜잭션 클라이언트 구성』

관련 개념

22 페이지의 『xa_open 문자열의 CHANNEL, TRPTYPE, CONNAME 및 QMNAME 매개변수』
이 정보를 사용하여 확장된 트랜잭션 클라이언트가 연결할 큐 관리자를 판별하는 데 이러한 매개변수를 사용하는 방법에 대해 이해할 수 있습니다.

23 페이지의 『xa_open에 대한 추가 오류 처리』
xa_open 호출은 특정 환경에서 실패합니다.

관련 태스크

24 페이지의 『TLS 채널에서 확장된 트랜잭션 클라이언트 사용』
xa_open 문자열을 사용하여 TLS 채널을 설정할 수 없습니다. 클라이언트 채널 정의 테이블(ccdt)을 사용하려면 다음 지시사항을 따르십시오.

관련 참조

23 페이지의 『TPM 및 AXLIB 매개변수』
확장된 트랜잭션 클라이언트는 TPM 및 AXLIB 매개변수를 사용하여 트랜잭션 관리자의 ax_reg 및 ax_unreg 함수를 찾습니다. 이러한 함수는 큐 관리자가 동적 등록을 사용하는 경우에만 사용됩니다.

23 페이지의 『확장된 트랜잭션 처리에서 실패 후 복구』
실패 후 트랜잭션 관리자는 불완전한 작업 단위를 복구할 수 있어야 합니다. 이를 수행하려면 트랜잭션 관리자가 실패 시 불완전한 작업 단위에 참여했던 큐 관리자를 자원 관리자로 열 수 있어야 합니다.

z/OS 확장 트랜잭션 클라이언트 연결을 위한 *IBM MQ for z/OS* 고려사항

일부 XA 트랜잭션 관리자는 일반적으로 IBM MQ for z/OS에 연결하는 클라이언트에 사용 가능한 기능과 호환되지 않는 일련의 트랜잭션 조정 호출을 사용합니다.

호환되지 않는 시퀀스가 감지되면 IBM MQ for z/OS는 연결에 대해 이상종료를 발행하고 클라이언트에 오류 응답을 리턴합니다.

예를 들어, `xa_prepare`는 이상종료 5C6-00D4007D를 수신하고, 리턴 코드 -3(XAER_RMERR)이 클라이언트에 리턴됩니다.

또 다른 예는 xa_end 가 이상 종료 5C6-00D40079를 수신하는 것입니다.

이 상황에 처하는 트랜잭션 관리자를 위해, 트랜잭션 관리자가 IBM MQ for z/OS와 상호작용할 수 있도록 다음과 같이 조치하십시오.

- APAR PI73140에 대한 수정사항을 적용합니다.
- 트랜잭션 관리자가 사용하는 서버 연결 채널에 대해 PI73140에서 제공하는 변경을 사용 가능하게 설정하십시오.

  SVRCONN 채널의 설명 필드에 CSQSERVICE1 키워드(대문자)를 지정하여 변경을 사용 가능하게 설정하십시오.

CSQSERVICE1 키워드를 갖는 채널에는 다음과 같은 제한사항이 따릅니다.

- GROUP 복구 단위 속성 지정이 허용되지 않습니다. QMGR 복구 단위 속성 지정만 허용됩니다. 처리는 xa_open 호출에 지정된 이름으로 판별됩니다. 큐 공유 그룹 이름이 사용되는 경우, XA 연결은 그룹 복구 단위를 요청합니다.

  xa_info 매개변수에서 큐 공유 그룹 이름을 지정하는 xa_open 호출은 *xaer_inval*이 발생하여 실패합니다.
- *MQGMO_LOCK* 및 *MQGMO_UNLOCK* 옵션은 허용되지 않습니다. *MQGMO_LOCK* 또는 *MQGMO_UNLOCK*이 있는 MQGET 호출은 MQRC_ENVIRONMENT_ERROR가 발생하여 실패합니다.

관련 개념

18 페이지의 『XA 준수 트랜잭션 관리자 구성』
먼저 IBM MQ 기본 클라이언트를 구성한 후 다음 주제의 정보를 사용하여 확장된 트랜잭션 기능을 구성합니다.

xa_open 문자열의 형식

xa_open 문자열에는 정의된 매개변수 이름 및 값 쌍이 들어 있습니다.

xa_open 문자열은 다음과 같은 형식을 가집니다.

```
parm_name1 = parm_value1, parm_name2 = parm_value2, ...
```

여기서, `parm_name` 은 매개변수의 이름이고 `parm_value` 는 매개변수의 값입니다. 매개변수의 이름은 대소문자를 구분하지 않지만 별도로 지정하지 않는 한 매개변수의 값은 대소문자를 구분합니다. 매개변수는 순서와 관계없이 지정할 수 있습니다.

매개변수의 이름, 의미 및 올바른 값은 다음과 같습니다.

이름
: **의미 및 올바른 값**

CHANNEL
: MQI 채널의 이름입니다.

  이 매개변수는 선택적 매개변수입니다. 이 매개변수가 제공되면 CONNAME 매개변수도 제공되어야 합니다.

TRPTYPE
: MQI 채널의 통신 프로토콜입니다. 다음 프로토콜이 올바른 값입니다.

  LU62
  : SNA LU 6.2

  NETBIOS
  : NetBIOS

  SPX
  : IPX/SPX

  TCP
  : TCP/IP

  이 매개변수는 선택적 매개변수입니다. 이 매개변수가 생략되면 기본값 TCP로 가정합니다. 이 매개변수의 값은 대소문자를 구분하지 않습니다.

CONNAME

MQI 채널의 서버 측 큐 관리자의 네트워크 주소입니다. 이 매개변수의 올바른 값은 TRPTYPE 매개변수의 값에 따라 다릅니다.

LU62

CPI-C 부가 정보 입력 항목을 식별하는 기호 대상 이름입니다.

파트너 LU의 네트워크 규정 이름은 올바른 값이 아니며 파트너 LU 알리어스도 아닙니다. 이는 트랜잭션 프로그램(TP) 이름 및 모드 이름을 지정하기 위한 추가 매개변수가 없기 때문입니다.

NETBIOS

NetBIOS 이름입니다.

SPX

4바이트 네트워크 주소, 6바이트 노트 주소 및 선택적 2바이트 소켓 번호입니다. 이러한 값은 16진 표기법으로 지정되어야 합니다. 마침표로 네트워크 및 노드 주소를 구분해야 하고 소켓 번호가 제공되는 경우 괄호로 묶어야 합니다. 예를 들면, 다음과 같습니다.

```
0a0b0c0d.804abcde23a1(5e86)
```

소켓 번호가 생략되는 경우 기본값 5e86으로 가정됩니다.

TCP

호스트 이름 또는 IP 주소이며 선택적으로 괄호로 묶은 포트 번호가 뒤에 올 수 있습니다. 포트 번호가 생략되는 경우 기본값 1414로 가정됩니다. 큐 관리자에 대한 여러 호스트 및 포트는 세미콜론 구분 기호를 사용하여 지정할 수 있습니다. 예를 들어 다음과 같습니다.

```
host1(1415);host2(1416);host3(1417)
```

이 매개변수는 선택적 매개변수입니다. 이 매개변수가 제공되면 CHANNEL 매개변수도 제공되어야 합니다.

QMNAME

MQI 채널의 서버 측 큐 관리자의 이름입니다. 이 이름은 공백이거나 단일 별표(*)일 수 없으며 별표로 시작되는 이름일 수 없습니다. 이는 이 매개변수가 이름으로 특정 큐 관리자를 식별함을 의미합니다.

이 매개변수는 필수 매개변수입니다.

클라이언트 애플리케이션이 특정 큐 관리자에 연결되면 같은 큐 관리자에 의해 트랜잭션 복구가 처리되어야 합니다.

애플리케이션이 z/OS 큐 관리자에 연결하는 경우 이 애플리케이션 특정 큐 관리자의 이름 또는 큐 공유 그룹(QSG)의 이름을 지정할 수 있습니다. 큐 관리자 이름 또는 큐 공유 그룹 이름을 사용하여 애플리케이션은 QMGR 복구 단위 속성 지정 또는 GROUP 복구 단위 속성 지정에 대한 트랜잭션에 참여할지 여부를 제어합니다. GROUP 복구 단위 속성 지정을 사용하여 트랜잭션 복구가 QSG의 모든 멤버에서 처리될 수 있습니다. GROUP 복구 단위를 사용하려면 **GROUPUR** 큐 관리자 속성을 사용으로 설정해야 합니다.

z/OS GROUP 복구 단위 사용에 대한 추가 정보는 큐 공유 그룹에서 복구 단위 속성 지정을 참조하십시오.

TPM

사용 중인 트랜잭션 관리자입니다. 올바른 값은 CICS 및 TUXEDO입니다.

확장된 트랜잭션 클라이언트는 이 매개변수 및 AXLIB 매개변수를 같은 목적으로 사용합니다. 이러한 매개변수에 대한 자세한 정보는 TPM 및 AXLIB 매개변수를 참조하십시오.

이 매개변수는 선택적 매개변수입니다. 이 매개변수의 값은 대소문자를 구분하지 않습니다.

AXLIB

트랜잭션 관리자의 ax_reg 및 ax_unreg 함수를 포함하는 라이브러리의 이름입니다.

이 매개변수는 선택적 매개변수입니다.

UID

인증을 위해 큐 관리자에 제공되는 사용자 ID입니다. 이 매개변수가 제공되면 **PWD** 매개변수도 제공되어야 합니다. 제공된 사용자 ID 및 비밀번호가 인증되면 이 사용자 ID는 트랜잭션 관리자의 연결 식별에 사용됩니다. 사용자 ID 및 비밀번호는 MQCONNX 호출 시 MQCSP 오브젝트를 채웁니다.

UID 및 **PWD** 매개변수는 클라이언트 및 서버 바인딩 둘 다에 유효합니다.

PWD

인증을 위해 큐 관리자에 제공되는 비밀번호입니다. 이 매개변수가 제공되면 **UID** 매개변수도 제공되어야 합니다.

경고: 일부 경우에 클라이언트 애플리케이션에 대한 MQCSP 구조의 비밀번호는 일반 텍스트로 네트워크 전체에 전송됩니다. 클라이언트 애플리케이션 비밀번호가 적절하게 보호될 수 있도록 하려면 IBM MQCSP 비밀번호 보호를 참조하십시오.

다음은 xa_open 문자열의 예입니다.

```
channel=MARS.SVR,trptype=tcp,conname=MARS(1415),qmname=MARS,tpm=cics
```

xa_open 문자열의 *CHANNEL, TRPTYPE, CONNAME* 및 *QMNAME* 매개변수

이 정보를 사용하여 확장된 트랜잭션 클라이언트가 연결할 큐 관리자를 판별하는 데 이러한 매개변수를 사용하는 방법에 대해 이해할 수 있습니다.

CHANNEL 및 CONNAME 매개변수가 xa_open 문자열에 제공되면 확장된 트랜잭션 클라이언트가 이러한 매개변수 및 TRPTYPE 매개변수를 사용하여 서버 큐 관리자에 대한 MQI 채널을 시작합니다.

CHANNEL 및 CONNAME 매개변수가 xa_open 문자열에 제공되지 않으면 확장된 트랜잭션 클라이언트가 MQSERVER 환경 변수의 값을 사용하여 MQI 채널을 시작합니다. MQSERVER 환경 변수가 정의되지 않으면 확장된 트랜잭션 클라이언트가 QMNAME 매개변수로 식별되는 클라이언트 채널 정의의 입력 항목을 사용합니다.

이러한 각각의 경우에서 확장된 트랜잭션 클라이언트는 QMNAME 매개변수의 값이 MQI 채널의 서버 측 큐 관리자 이름인지 확인합니다. 그렇지 않은 경우 xa_open 호출이 실패하고 트랜잭션 관리자가 애플리케이션에 실패를 보고합니다.

애플리케이션이 7.0.1보다 이전 버전의 큐 관리자에 연결하는 경우 xa_open 호출이 성공하지만 트랜잭션에 QMGR 복구 속성이 있습니다. z/OS 복구의 GROUP 단위 처리가 필요한 응용프로그램은 7.0.1 이상의 큐 관리자에만 연결되는지 확인하십시오.

z/OS 애플리케이션이 QMNAME 매개변수 필드에서 큐 공유 그룹 이름을 사용하고 GROUPUR 특성이 연결되는 큐 관리자에서 사용 안함으로 설정되면 xa_open 호출은 실패합니다.

z/OS 응용프로그램 클라이언트가 7.0.1 이상의 z/OS 대기열 관리자에 연결 중인 경우 QMNAME 매개변수에 QSG (대기열 공유 그룹) 이름을 지정할 수 있습니다. 이 경우 애플리케이션 클라이언트는 GROUP 복구 단위 속성 지정과의 트랜잭션에 참여할 수 있습니다. GROUP 복구 단위 처리에 대한 자세한 정보는 복구 단위 처리의 내용을 참조하십시오.

트랜잭션 관리자가 xa_open 호출을 발행하는 데 사용한 것과 같은 스레드에서 클라이언트 애플리케이션이 나중에 MQCONN 또는 MQCONNX를 호출하는 경우 이 애플리케이션은 xa_open 호출에 의해 시작된 MQI 채널에 대한 연결 핸들을 수신합니다. 두 번째 MQI 채널이 시작되지 않습니다. 확장된 트랜잭션 클라이언트가 MQCONN 또는 MQCONNX 호출 시 *QMgrName* 매개변수의 값이 MQI 채널의 서버 측 큐 관리자의 이름인지 확인합니다. 그렇지 않은 경우 MQCONN 또는 MQCONNX 호출이 MQRC_ANOTHER_Q_MGR_CONNECTED 이유 코드로 실패합니다. *QMgrName* 매개변수의 값이 공백이거나 단일 별표(*)이거나 별표로 시작되는 경우 MQCONN 또는 MQCONNX 호출이 MQRC_Q_MGR_NAME_ERROR 이유 코드로 실패합니다.

트랜잭션 관리자가 같은 스레드에서 xa_open을 호출하기 전에 클라이언트 애플리케이션이 MQCONN 또는 MQCONNX를 호출하여 MQI 채널을 이미 시작한 경우 트랜잭션 관리자는 이 MQI 채널을 대신 사용합니다. 두 번째 MQI 채널이 시작되지 않습니다. 확장된 트랜잭션 클라이언트가 xa_open 문자열의 QMNAME 매개변수 값이 서버 큐 관리자의 이름인지 확인합니다. 그렇지 않은 경우 xa_open 호출이 실패합니다.

클라이언트 애플리케이션이 MQI 채널을 먼저 시작하는 경우 MQCONN 또는 MQCONNX 호출 시 *QMgrName* 매개변수의 값은 공백이거나 단일 별표(*)이거나 별표로 시작할 수 있습니다. 그러나 이러한 상황에서는 트랜잭션 관리자가 같은 스레드에서 나중에 xa_open을 호출할 때 자원 관리자로 열려고 하는 큐 관리자와 애플리케이션

이 연결하는 큐 관리자가 같은지 확인해야 합니다. *QMgrName* 매개변수의 값이 이름에 따라 명시적으로 큐 관리자를 식별하는 경우 문제점이 더 적게 발생할 수 있습니다.

TPM 및 *AXLIB* 매개변수

확장된 트랜잭션 클라이언트는 TPM 및 AXLIB 매개변수를 사용하여 트랜잭션 관리자의 ax_reg 및 ax_unreg 함수를 찾습니다. 이러한 함수는 큐 관리자가 동적 등록을 사용하는 경우에만 사용됩니다.

TPM 매개변수가 xa_open 문자열에 제공되지만 AXLIB 매개변수는 제공되지 않는 경우 확장된 트랜잭션 클라이언트는 TPM 매개변수의 값을 기반으로 AXLIB 매개변수의 값을 가정합니다. AXLIB 매개변수의 가정 값에 대해서는 23 페이지의 표 3의 내용을 참조하십시오.

표 *3. AXLIB* 매개변수의 가정 값

TPM의 값	플랫폼	AXLIB의 가정 값
CICS	AIX	/usr/lpp/encina/lib/libEncServer.a(EncServer_shr.o)
CICS	HP-UX	/opt/encina/lib/libEncServer.sl
CICS	Solaris	/opt/encina/lib/libEncServer.so
CICS	Windows 시스템	libEncServer
Tuxedo	AIX	/usr/lpp/tuxedo/lib/libtux.a(libtux.so.60)
Tuxedo	HP-UX	/opt/tuxedo/lib/libtux.sl
Tuxedo	Solaris	/opt/tuxedo/lib/libtux.so.60
Tuxedo	Windows 시스템	libtux

AXLIB 매개변수가 xa_open 문자열에 제공되는 경우 확장된 트랜잭션 클라이언트는 그 값을 사용하여 TPM 매개변수의 값을 기반으로 하여 가정된 값을 대체합니다. AXLIB 매개변수는 TPM 매개변수가 지정된 값을 갖지 않는 트랜잭션 관리자에 대해서도 사용할 수 있습니다.

*xa_open*에 대한 추가 오류 처리

xa_open 호출은 특정 환경에서 실패합니다.

이 절의 주제에서는 xa_open 호출이 실패하는 상황에 대해 설명합니다. 이 호출은 다음 상황 중 어느 하나가 발생하는 경우에도 실패합니다.

- xa_open 문자열에 오류가 있습니다.
- MQI 채널을 시작하기에는 정보가 충분하지 않습니다.
- MQI 채널을 시작하려고 시도하는 중에 문제점이 발생했습니다(예: 서버 큐 관리자가 실행되지 않음).

확장된 트랜잭션 처리에서 실패 후 복구

실패 후 트랜잭션 관리자는 불완전한 작업 단위를 복구할 수 있어야 합니다. 이를 수행하려면 트랜잭션 관리자가 실패 시 불완전한 작업 단위에 참여했던 큐 관리자를 자원 관리자로 열 수 있어야 합니다.

따라서 임의의 구성 정보에 대한 변경을 작성하기 전에 모든 불완전한 작업 단위가 해결되었는지 확인해야 합니다.

또는 구성 변경이 트랜잭션 관리자가 열어야 하는 큐 관리자를 여는 기능에 영향을 주지 않는지 확인해야 합니다. 다음은 이러한 구성 변경의 예입니다.

- xa_open 문자열의 내용 변경
- MQSERVER 환경 변수의 값 변경
- 클라이언트 채널 정의 테이블(CCDT)의 입력 항목 변경
- 서버 연결 채널 정의 삭제

XA 스위치 구조

두 개의 XA 스위치 구조가 각 플랫폼의 확장된 트랜잭션 클라이언트에 제공됩니다.

이러한 스위치 구조는 다음과 같습니다.

MQRMIXASwitch
이 스위치 구조는 자원 관리자 역할을 하는 큐 관리자가 동적 등록을 사용하지 않는 경우에 트랜잭션 관리자가 사용합니다.

MQRMIXASwitchDynamic
이 스위치 구조는 자원 관리자 역할을 하는 큐 관리자가 동적 등록을 사용하는 경우에 트랜잭션 관리자가 사용합니다.

이러한 스위치 구조는 24 페이지의 표 4에 표시된 라이브러리에 있습니다.

표 *4. XA* 스위치 구조를 포함하는 *IBM MQ* 라이브러리

플랫폼	XA 스위치 구조를 포함하는 라이브러리
AIX HP-UX Linux Solaris	*MQ_INSTALLATION_PATH*/lib/libmqcxa
Windows 시스템	*MQ_INSTALLATION_PATH*\bin\mqcxa.dll [1]

*MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

각 스위치 구조에서 IBM MQ 자원 관리자의 이름은 MQSeries_XA_RMI지만 다수의 큐 관리자가 같은 스위치 구조를 공유할 수 있습니다.

관련 개념

24 페이지의 『동적 등록 및 확장된 트랜잭션 처리』
동적 등록 사용은 트랜잭션 관리자가 발행하는 xa_ 함수 호출의 수를 줄일 수 있기 때문에 최적화된 양식입니다.

동적 등록 및 확장된 트랜잭션 처리
동적 등록 사용은 트랜잭션 관리자가 발행하는 xa_ 함수 호출의 수를 줄일 수 있기 때문에 최적화된 양식입니다.

큐 관리자가 동적 등록을 사용하지 않는 경우 트랜잭션 관리자는 모든 작업 단위에 큐 관리자를 관여시킵니다. 트랜잭션 관리자는 큐 관리자에게 작업 단위 내에서 업데이트되는 자원이 없더라도 xa_start, xa_end 및 xa_prepare를 호출하여 이를 수행합니다.

큐 관리자가 동적 등록을 사용하는 경우 트랜잭션 관리자는 큐 관리자가 작업 단위에 관여하지 않는 것으로 가정하여 시작하고 xa_start를 호출하지 않습니다. 그러면 큐 관리자는 해당 자원이 동기점 제어 내에서 업데이트되는 경우에만 작업 단위에 관여하게 됩니다. 이런 경우가 발생하면 확장 트랜잭션 클라이언트가 ax_reg를 호출하여 큐 관리자의 관여를 등록합니다.

TLS 채널에서 확장된 트랜잭션 클라이언트 사용

xa_open 문자열을 사용하여 TLS 채널을 설정할 수 없습니다. 클라이언트 채널 정의 테이블(ccdt)을 사용하려면 다음 지시사항을 따르십시오.

이 태스크 정보

xa_open xa_info 문자열의 제한된 크기로 인해 큐 관리자에 연결하는 xa_open 문자열 메소드를 사용하여 TLS 채널을 설정하는 데 필요한 모든 정보를 전달하는 것은 불가능합니다. 따라서 클라이언트 채널 정의 테이블을 사용해야 합니다. 또는 트랜잭션 관리자가 허용하는 경우 xa_open 호출을 발행하기 전에 MQCONNX로 채널을 작성해야 합니다.

클라이언트 채널 정의 테이블을 사용하려면 다음 단계를 따르십시오.

프로시저

1. 필수 qmname(큐 관리자 이름) 매개변수만 포함하는 xa_open 문자열을 지정하십시오. 예를 들면, 다음과 같습니다. `XA_Open_String=qmname=MYQM`

2. 큐 관리자를 사용하여 필수 TLS 매개변수로 CLNTCONN(클라이언트 연결) 채널을 정의하십시오. 큐 관리자 이름을 CLNTCONN 정의의 QMNAME 속성에 포함시키십시오. 이는 xa_open 문자열의 qmname과 일치됩니다.
3. 클라이언트 채널 정의 테이블(CCDT)에서 또는 Windows의 경우 Active Directory에서 클라이언트 시스템이 CLNTCONN 정의를 사용할 수 있도록 하십시오.
4. CCDT를 사용 중인 경우 환경 변수 MQCHLLIB 및 MQCHLTAB를 사용하여 CLNTCONN 채널의 정의가 포함된 CCDT를 식별하십시오. 클라이언트 애플리케이션 및 트랜잭션 관리자 둘 다 사용하는 환경에서 이러한 변수를 설정하십시오.

결과

이는 SSLCIPH, CipherSpec 등의 올바르게 인증되어야 하는 TLS 속성이 있는 해당 큐 관리자에 대한 채널 정의를 트랜잭션 관리자에게 제공합니다.

*CICS*를 위한 확장된 트랜잭션 클라이언트 구성

XAD 자원 정의를 CICS 리젼에 추가하여 CICS에서 사용할 확장된 트랜잭션 클라이언트를 구성합니다.

CICS 자원 정의 온라인(RDO) 명령 **cicsadd**를 사용하여 XAD 자원 정의를 추가하십시오. 이 XAD 자원 정의는 다음 정보를 지정합니다.

- xa_open 문자열
- 스위치 로드 파일의 완전한 경로 이름

다음 플랫폼 각각의 CICS에서 사용할 한 개의 스위치 로드 파일이 제공됩니다. AIX, HP-UX, Solaris 및 Windows 시스템.각 스위치 로드 파일은 동적 등록에 사용되는 XA 스위치 구조 MQRMIXASwitchDynamic에 대한 포인터를 리턴하는 기능을 포함합니다. 각 스위치 로드 파일의 완전한 경로 이름은 25 페이지의 표 5의 내용을 참조하십시오.

표 *5*. 스위치 로드 파일

플랫폼	스위치 로드 파일
AIX HP-UX Linux Solaris	*MQ_INSTALLATION_PATH*/lib/amqczsc
Windows 시스템	*MQ_INSTALLATION_PATH*\bin\mqcc4swi.dll [1]

*MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

다음은 Windows 시스템의 XAD 자원 정의의 예입니다.

```
cicsadd -c xad -r REGION1 WMQXA \
    ResourceDescription="IBM MQ queue manager MARS" \
    XAOpen="channel=MARS.SVR,trptype=tcp,conname=MARS(1415),qmname=MARS,tpm=cics" \
    SwitchLoadFile="C:\Program Files\IBM\MQ\bin\mqcc4swi.dll"
```

CICS 리젼에 XAD 자원 정의를 추가하는 것에 대한 자세한 정보는 사용자 플랫폼에 해당하는 *CICS* 관리 참조 및 *CICS* 관리 안내서를 참조하십시오.

확장된 트랜잭션 클라이언트에서 CICS를 사용하는 것에 대해 다음 정보를 참고하십시오.

- IBM MQ에 대한 한 개의 XAD 정의만 CICS 리젼에 추가할 수 있습니다. 이는 한 큐 관리자만 리젼과 연관될 수 있고 리젼에서 실행되는 모든 CICS 애플리케이션은 해당 큐 관리자에만 연결할 수 있음을 의미합니다. 다른 큐 관리자에 연결하는 CICS 애플리케이션을 실행하려는 경우 이 애플리케이션을 다른 리젼에서 실행해야 합니다.
- 리젼 내 각 애플리케이션 서버는 초기화 중 xa_open을 호출하고 리젼과 연관된 큐 관리자에 대한 MQI 채널을 시작합니다. 이는 애플리케이션 서버가 시작되기 전에 큐 관리자를 시작해야 하고 그렇지 않은 경우 xa_open

호출이 실패함을 의미합니다. 이 애플리케이션 서버가 이후에 처리하는 모든 IBM MQ MQI client 애플리케이션은 같은 MQI 채널을 사용합니다.

- MQI 채널이 시작되고 채널의 클라이언트 측에 보안 엑시트가 없는 경우 클라이언트 시스템에서 서버 연결 MCA로 플로우하는 사용자 ID는 `cics`입니다. 특정 상황에서 서버 연결 MCA가 이후에 클라이언트 애플리케이션을 대신해서 큐 관리자 자원에 액세스하려고 시도하면 큐 관리자는 권한 검사를 위해 이 사용자 ID를 사용합니다. 이 사용자 ID가 권한 검사에 사용되는 경우 이 사용자 ID가 액세스해야 하는 모든 자원에 대한 액세스 권한을 가지고 있는지 확인해야 합니다.

  큐 관리자가 권한 검사를 위해 이 사용자 ID를 사용하는 경우에 대한 정보는 보안을 참조하십시오.
- IBM MQ 클라이언트 시스템에서의 사용을 위해 제공되는 CICS 태스크 엑시트는 26 페이지의 표 6에 나열됩니다. 이러한 엑시트는 IBM MQ 서버 시스템에 대한 해당 엑시트를 구성한 것과 같은 방식으로 구성합니다. 따라서 이 정보는 CICS 사용자 엑시트 사용을 참조하십시오.

표 *6. CICS* 태스크 종료 엑시트

플랫폼	소스	라이브러리
AIX HP-UX Linux Solaris	amqzscgx.c	amqczscg
Windows 시스템	amqzscgn.c	mqcc1415.dll

*Tuxedo*를 위한 확장된 트랜잭션 클라이언트 구성

Tuxedo에서 사용하도록 XAD 자원 정의를 구성하려면 UBBCONFIG 파일 및 자원 관리자 테이블을 업데이트하십시오.

Tuxedo에서 사용할 XAD 자원 정의를 구성하려면 다음 조치를 수행하십시오.

- 애플리케이션의 Tuxedo UBBCONFIG 파일 GROUPS 섹션에서 OPENINFO 매개변수를 사용하여 xa_open 문자열을 지정하십시오.

  이를 수행하는 방법에 대한 예는 Tuxedo 샘플 프로그램에서 사용하도록 제공된 샘플 UBBCONFIG 파일을 참조하십시오. AIX, HP-UX 및 Solaris에서 이 파일의 이름은 ubbstxcx.cfg이고 Windows 시스템에서 이 파일의 이름은 `ubbstxcn.cfg`입니다.
- Tuxedo 자원 관리자 테이블의 큐 관리자에 대한 항목에서:
  - udataobj/RM(AIX, HP-UX 및 Solaris)
  - udataobj\rm(Windows 시스템)

  XA 스위치 구조의 이름 및 이 구조를 포함하는 라이브러리의 완전한 경로 이름을 지정하십시오. 각 플랫폼에 대해 이를 수행하는 방법의 예는 TUXEDO 샘플을 참조하십시오. Tuxedo는 자원 관리자의 동적 등록을 지원하므로 MQRMIXASwitch 또는 MQRMIXASwitchDynamic을 사용할 수 있습니다.

MTS(Microsoft Transaction Server)

MTS를 트랜잭션 관리자로 사용하기 전에 필요한 추가 구성은 없습니다. 그러나 몇 가지 참고해야 할 점이 있습니다.

확장된 트랜잭션 클라이언트에서 MTS를 사용하는 것에 대해 다음 정보를 참고하십시오.

- MTS 애플리케이션은 서버 큐 관리자에 연결할 때 항상 MQI 채널을 시작합니다. 트랜잭션 관리자 역할을 할 때 MTS는 같은 MQI 채널을 사용하여 큐 관리자와 통신합니다.
- 실패 후 MTS는 불완전한 작업 단위를 복구할 수 있어야 합니다. 이를 수행하려면 MTS가 실패 시 불완전한 작업 단위에 참여했던 큐 관리자와 통신할 수 있어야 합니다.

  MTS 애플리케이션이 서버 큐 관리자에 연결하고 MQI 채널을 시작하는 경우 확장된 트랜잭션 클라이언트는 필요한 경우에 실패 후 채널이 재시작될 수 있도록 MQCONN 또는 MQCONNX 호출의 매개변수에서 충분한 정보를 추출합니다. 확장 트랜잭션 클라이언트는 이 정보를 MTS에 전달하고 MTS는 이 정보를 그 로그에 기록합니다.

MTS 애플리케이션이 MQCONN 호출을 발행하는 경우 이 정보는 단순히 큐 관리자의 이름입니다. MTS 애플리케이션이 MQCONNX 호출을 발행하고 채널 정의 구조인 MQCD를 제공하는 경우 이 정보에는 MQI 채널의 이름, 서버 큐 관리자의 네트워크 주소 및 채널의 통신 프로토콜이 포함됩니다.

복구 상황에서 MTS는 이 정보를 확장된 트랜잭션 클라이언트에 다시 전달하고 확장된 트랜잭션 클라이언트는 이 정보를 사용하여 MQI 채널을 재시작합니다.

구성 정보를 변경해야 하는 경우 변경을 작성하기 전에 불완전한 작업 단위가 모두 해결되었는지 확인하십시오. 또는 구성 변경이 확장된 트랜잭션 클라이언트가 MTS에서 기록된 정보를 사용하여 MQI 채널을 재시작하는 기능에 영향을 주지 않는지 확인하십시오. 다음은 이러한 구성 변경의 예입니다.

- MQSERVER 환경 변수의 값 변경
- 클라이언트 채널 정의 테이블(CCDT)의 입력 항목 변경
- 서버 연결 채널 정의 삭제

• 확장된 트랜잭션 클라이언트를 MTS와 함께 사용할 때 다음 조건을 참조하십시오.
  - 단일 스레드 내에서 클라이언트 애플리케이션은 한 번에 한 큐 관리자에만 연결할 수 있습니다.
  - 클라이언트 애플리케이션의 각 스레드는 서로 다른 큐 관리자에 연결할 수 있습니다.
  - 클라이언트 애플리케이션은 공유 연결 핸들을 사용할 수 없습니다.

MQI 채널 정의

새 채널 정의를 작성하려면 같은 채널 이름과 호환 가능한 채널 유형을 사용하여 연결의 양 끝에 하나씩 **두 개** 채널 정의를 작성해야 합니다. 이 경우 채널 유형은 *server-connection* 및 *client-connection*입니다.

사용자 정의 채널

서버가 채널을 자동으로 정의하지 않는 경우 채널 정의를 작성하고 IBM MQ MQI client 시스템의 IBM MQ 애플리케이션에 이 채널에 대한 액세스를 제공하는 두 가지 방법이 있습니다.

이러한 두 가지 방법은 다음에 자세히 설명되어 있습니다.

1. IBM MQ 클라이언트에서 한 개의 채널 정의를 작성하고 서버에서 다른 한 개의 채널 정의를 작성합니다.

   이는 IBM MQ MQI client와 서버 플랫폼의 모든 조합에 적용됩니다. 시스템에서 시작하거나 설정을 테스트할 때 이 방법을 사용하십시오.

   이 방법 사용에 대한 자세한 내용은 33 페이지의 『다른 플랫폼에서 서버 연결 및 클라이언트 연결 정의 작성』의 내용을 참조하십시오.

2. 서버 시스템에서 두 채널 정의를 모두 작성합니다.

   다중 채널 및 IBM MQ MQI client 시스템을 동시에 설정하는 경우에 이 방법을 사용하십시오.

   이 방법 사용에 대한 자세한 내용은 36 페이지의 『서버에서 서버 연결 및 클라이언트 연결 정의 작성』의 내용을 참조하십시오.

자동으로 정의되는 채널

z/OS 외의 플랫폼에서 IBM MQ 제품은 채널 정의가 없는 경우 서버에서 자동으로 채널 정의를 생성할 수 있는 기능을 포함합니다.

클라이언트로부터 인바운드 연결 요청을 수신하고 해당 큐 관리자에서 적절한 서버 연결 정의를 찾을 수 없는 경우에 IBM MQ는 자동으로 정의를 작성하고 이 정의를 큐 관리자에 추가합니다. 자동 정의는 기본 서버 연결 채널 정의(SYSTEM.AUTO.SVRCONN)를 기반으로 합니다. CHAD 매개변수와 함께 ALTER QMGR 명령을 사용하여(또는 ChannelAutoDef 매개변수와 함께 PCF 명령 Change Queue Manager 사용) 큐 관리자 오브젝트를 업데이트함으로써 서버 연결 정의의 자동 정의를 사용할 수 있습니다.

관련 개념

154 페이지의 『채널 제어 기능』
채널 제어 기능은 채널을 정의, 모니터, 제어할 수 있는 기능을 제공합니다.

ULW AMQP 채널 작성 및 사용

IBM MQ 설치에 MQ Light API에 대한 IBM MQ 지원을 설치하는 경우 IBM MQ MQSC 명령(**runmqsc**)을 실행하여 채널을 정의, 대체, 삭제, 시작 및 중지할 수 있습니다. 채널의 상태를 볼 수도 있습니다.

시작하기 전에

이 태스크에서는 사용자가 AMQP 채널을 설치한 것으로 간주합니다. IBM MQ를 설치할 때 AMQP 서비스 컴포넌트를 선택하여 수행합니다. 자세한 정보는 사용자 플랫폼에 적합한 링크에 따라 "AMQP 서비스"의 테이블 행을 찾으십시오.

- AIX AIX 시스템을 위한 IBM MQ 컴포넌트
- HP-UX HP-UX 시스템을 위한 IBM MQ 컴포넌트
- Linux Linux 시스템을 위한 IBM MQ rpm 컴포넌트
- Linux Linux Ubuntu 시스템을 위한 IBM MQ Debian 컴포넌트
- Solaris Solaris 시스템을 위한 IBM MQ 컴포넌트
- Windows Windows 시스템을 위한 IBM MQ 기능

큐 관리자에 대한 테스트 연결을 작성하려면 MQ Light 클라이언트가 있어야 합니다. Node.js, Ruby, Java 및 Python에 대해 MQ Light 클라이언트를 사용할 수 있습니다. 사용 가능한 클라이언트에 대한 자세한 정보는 IBM MQ Light 커뮤니티 웹 사이트를 참조하십시오.

이 태스크는 MQ Light Node.js 클라이언트를 기반으로 합니다. 그러나 IBM MQ 큐 관리자와 관련된 단계는 임의의 클라이언트에 대해 동일합니다.

이 태스크 정보

다음 프로시저에서는 사용자가 기존의 큐 관리자를 가지고 있다고 간주합니다.

새 큐 관리자가 필요한 경우 `*mqinstall*/amqp/samples` 디렉토리에 있는 샘플 스크립트가 포함됩니다. 스크립트는 새 큐 관리자를 시작하고, AMQP 서비스를 시작하고, `SAMPLE.AMQP.CHANNEL`이라는 새 채널을 작성한 후 채널을 시작합니다.

참고: AMQP 채널은 사용자 정의 AMQP 서비스를 지원하지 않습니다. AMQP 채널은 시스템 기본 SYSTEM.AMQP.SERVICE 서비스만 지원합니다.

Windows Linux 샘플 스크립트(Linux에서는 `SampleMQM.sh`)를 실행하거나 Windows에서 `SampleMQM.bat`을(를) 실행하는 경우 29 페이지의 『6』에서 다음 프로시저를 시작할 수 있습니다.

기본 채널인 SYSTEM.DEF.AMQP를 사용하여 큐 관리자에 대한 MQ Light 연결을 테스트하거나 새 채널을 작성할 수 있습니다.

다음 프로시저는 기본 채널을 사용합니다.

프로시저

1. `*mqinstall*/bin/` 디렉토리에서 **runmqsc**를 시작하십시오.

   ```
   runmqsc QMNAME
   ```

2. V 9.0.5
   (큐 관리자가 IBM MQ 9.0.4 이전인 경우에만 필요). AMQP 기능이 설치되었고 올바르게 작동 중인지 확인하십시오.

   START SERVICE 명령을 사용하여 JVM을 제어하는 IBM MQ 서비스를 시작하십시오.

   ```
   START SERVICE(SYSTEM.AMQP.SERVICE)
   ```

참고: IBM MQ 9.0.5에서 SYSTEM.AMQP.SERVICE는 **CONTROL** 속성은 *QMGR*로 설정됩니다. 이로 인해 큐 관리자가 시작되면 서비스가 자동으로 시작됩니다. **CONTROL** 속성을 *MANUAL*로 설정하면 큐 관리자가 시작될 때 서비스가 시작되지 않도록 할 수 있습니다.

큐 관리자가 시작되면 AMQP 서비스 및 AMQP 채널이 정의된 경우 자동으로 시작됩니다.

3. MCAUSER 사용자 ID를 설정하십시오.

   AMQP 클라이언트가 채널에 연결되면 큐 관리자에 연결할 때 사용되는 MCAUSER 사용자 ID를 채널에서 지정합니다. MCAUSER의 기본값은 공백입니다. AMQP 클라이언트를 큐 관리자에 연결하려면 MCAUSER 값을 지정해야 하며, 이는 IBM MQ 토픽에 대한 발행/구독 권한을 가진 올바른 IBM MQ 사용자여야 합니다.

   참고: Windows Windows에서 MCAUSER 사용자 ID 설정은 최대 12자 길이의 사용자 ID에 대해서만 지원됩니다.

   a) **ALTER CHANNEL** 명령을 사용하여 MCAUSER 사용자 ID를 설정하십시오.

   ```
   ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) MCAUSER(User ID)
   ```

   b) 다음 두 **setmqaut** 명령을 사용하여 MCAUSER 사용자 ID에 토픽을 발행하고 구독할 권한을 부여하십시오.

   ```
   setmqaut -m QMNAME -t topic -n SYSTEM.BASE.TOPIC -p MCAUSER
    -all +pub +sub
   ```

   및

   ```
   setmqaut -m QMNAME -t qmgr -p MCAUSER -all +connect
   ```

   MCAUSER 사용자 ID가 추가 또는 대체되는 중에 채널이 실행 중인 경우 채널을 중지하고 다시 시작해야 합니다.

   참고: MCAUSER 사용자 ID가 설정되지 않거나 MCAUSER 사용자 ID에 IBM MQ 토픽을 발행하고 구독할 권한이 부여되지 않은 경우 AMQP 클라이언트에서 오류 메시지가 수신됩니다.

4. **START CHANNEL** 명령을 사용하여 기본 SYSTEM.DEF.AMQP 채널을 시작하십시오.

   ```
   START CHANNEL(SYSTEM.DEF.AMQP)
   ```

5. 채널 상태를 검사하려는 경우 **DISPLAY CHSTATUS** 명령을 사용하십시오.

   ```
   DISPLAY CHSTATUS(SYSTEM.DEF.AMQP) CHLTYPE(AMQP)
   ```

   채널이 올바르게 실행 중인 경우 STATUS(RUNNING)가 명령 출력에 표시됩니다.

6. 기본 포트를 변경하십시오.

   AMQP 1.0 연결의 기본 포트는 5672입니다. 이전에 MQ Light를 설치한 경우 가능한 포트 5672를 이미 사용 중이면 AMQP 채널이 사용하는 포트를 변경해야 합니다. **ALTER CHANNEL** 명령을 사용하여 포트를 변경하십시오.

   ```
   ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) PORT(NEW PORT NUMBER)
   ```

7. 채널 인증(CHLAUTH) 규칙을 사용하여 AMQP 채널에 대한 연결을 필터링하거나 차단하지 않으려는 경우 다음과 같이 큐 관리자에서 채널 인증을 사용 안함으로 설정하십시오.

   ```
   alter qmgr chlauth(disabled)
   ```

   프로덕션 큐 관리자에서 연결 인증을 사용 안함으로 설정하지 않는 것이 좋습니다. 개발 환경에서 연결 인증만을 사용 안함으로 설정해야 합니다.

   또는 AMQP 채널에 대한 특정 연결을 허용하도록 큐 관리자 채널 인증 규칙을 구성하십시오.

8. 옵션: 큐 관리자에 대해 구성된 키 저장소를 사용하여 채널에서 SSL/TLS 암호화를 사용으로 설정하려는 경우 채널의 SSLCIPH 속성을 적절한 암호 스펙으로 설정해야 합니다. 기본적으로 암호 스펙은 공백이며, 이는 SSL/TLS 암호화가 채널에서 사용되지 않음을 의미합니다. **ALTER CHANNEL** 명령을 사용하여 암호 스펙을 설정하십시오. 예를 들면, 다음과 같습니다.

   ```
   ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) SSLCIPH(CIPHER SPECIFICATION)
   ```

   또한 다음과 같이 설정할 수 있는 SSL/TLS 암호화와 연관된 다수의 기타 채널 구성 옵션이 있습니다.

   - 기본적으로 큐 관리자 CERTLABL 속성에 해당한 레이블을 가진 큐 관리자 키 저장소에 있는 인증서는 채널을 위한 SSL/TLS 암호화에 사용되는 이름입니다. CERTLABL을 설정하여 다른 인증을 선택할 수 있습니다. **ALTER CHANNEL** 명령을 사용하여 필수 인증을 위한 레이블을 지정하십시오.

     ```
     ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) CERTLABL(CERTIFICATE LABEL)
     ```

   - SSL/TLS 클라이언트 연결에서 인증이 필요하도록 채널을 설정할 수 있습니다. SSLCAUTH를 설정하여 SSL/TLS 클라이언트 연결에서 인증이 필요한지 여부를 선택할 수 있습니다. **ALTER CHANNEL** 명령을 사용하여 SSL/TLS 클라이언트 연결에서 인증이 필요한지 여부를 설정하십시오. 예를 들면, 다음과 같습니다.

     ```
     ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) SSLCAUTH(REQUIRED or OPTIONAL)
     ```

   - V 9.0.0.10 SSLCAUTH 속성을 REQUIRED로 설정하는 경우 클라이언트에서 인증서의 식별 이름(DN)을 검사할 수 있습니다. 클라이언트에서 인증서의 식별 이름을 검사하려면 SSLPEER 속성을 설정하십시오. **ALTER CHANNEL** 명령을 사용하여 클라이언트에서 인증서의 식별 이름을 확인하십시오. 예를 들면, 다음과 같습니다.

     ```
     ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) SSLPEER (DN SPECIFICATION)
     ```

     또는 이 메소드는 SSLPEER 속성 사용보다 더 큰 단위를 제공하므로 채널 인증 레코드를 사용하여 연결을 허용하거나 차단할 수 있습니다. SSLPEER 설정 및 채널 인증 레코드를 대안으로 사용에 대한 자세한 정보는 SSL 피어를 참조하십시오.

9. 다음 명령을 실행하여 MQ Light Node.js 클라이언트를 설치하십시오.

   ```
   npm install mqlight
   ```

10. node_modules/mqlight/samples 디렉토리를 탐색하고 샘플 수신자 애플리케이션을 실행하십시오.

    - 기본 포트 번호를 사용하는 경우 샘플 수신자 애플리케이션을 실행할 수 있습니다.

      ```
      node recv.js
      ```

    - 다른 포트 번호를 사용하도록 AMQP 채널을 구성한 경우 매개변수와 함께 샘플 수신자 애플리케이션을 실행하여 새 포트 번호를 지정할 수 있습니다.

      ```
      node recv.js -s amqp://localhost:6789
      ```

    기본 채널에 연결되면 다음 메시지가 표시됩니다.

    ```
    Connected to amqp://localhost:5672 using client-id recv_e79c55d
    Subscribed to pattern: public
    ```

    이제 애플리케이션이 큐 관리자에 연결되고 메시지를 수신하려고 대기 중입니다. 이는 토픽 public에 등록됩니다.

    참고: -i 매개변수를 사용하여 지정하지 않는 경우 client-id가 자동으로 생성됩니다.

11. 새 명령 창에서 node_modules/mqlight/samples 디렉토리를 탐색하고 다음 명령을 실행하여 샘플 송신자 애플리케이션을 실행하십시오.

```
node send.js
```

수신자 애플리케이션의 명령 창에서 Hello World 메시지가 표시됩니다.

12. **AMQSSUB** IBM MQ 샘플을 사용하여 MQ Light 샘플 메시지를 수신하십시오.

    Linux 및 Windows에서는 다음 위치에서 샘플을 찾을 수 있습니다.

    - Linux Linux에서 *mqinstall*/samp/bin 디렉토리
    - Windows Windows에서 *mqinstall*/Tools\c\Samples\Bin 디렉토리

    a) 다음 명령을 실행하여 샘플을 실행하십시오.

    ```
    amqssub public QM-name.
    ```

    b) 다음 명령을 재실행하여 IBM MQ 애플리케이션에 메시지를 전송하십시오.

    ```
    node send.js
    ```

13. **DEFINE CHANNEL** 명령을 사용하여 추가 AMQP 채널을 작성하십시오.

```
DEFINE CHANNEL(MY.AMQP.CHANNEL) CHLTYPE(AMQP) PORT(2345)
```

채널을 정의하는 경우 **START CHANNEL** 명령을 사용하여 수동으로 시작해야 합니다.

```
START CHANNEL(MY.AMQP.CHANNEL)
```

채널이 올바르게 실행 중인지 확인하기 위해 새 채널의 포트를 지정하여 샘플 수신자 애플리케이션을 실행할 수 있습니다.

```
node recv.js -s amqp://localhost:2345
```

다음에 수행할 작업

다음 명령을 사용하여 IBM MQ 연결을 표시하고 채널을 중지하고 채널을 삭제하십시오.

DISPLAY CONN(*) TYPE(*CONN*) WHERE (*CHANNEL EQ SYSTEM.DEF.AMQP*)
AMQP 채널이 큐 관리자에서 작성된 IBM MQ 연결을 표시합니다.

DISPLAY CHSTATUS(*) CHLTYPE(*AMQP*) CLIENTID(*) ALL
지정된 채널에 연결된 AMQP 클라이언트 목록을 표시합니다.

STOP CHANNEL (*MY.AMQP.CHANNEL*)
AMQP 채널을 중지하고 청취 중인 포트를 닫습니다.

DELETE CHANNEL (*MY.AMQP.CHANNEL*)
사용자가 작성한 채널을 삭제합니다.

참고: 기본 채널 SYSTEM.DEF.AMQP를 삭제하지 마십시오.

runmqsc 또는 PCF를 사용하여 연관된 큐 관리자가 있는지 여부 및 AMQP 기능이 IBM MQ 설치에 설치되는지 여부를 판별할 수 있습니다.

- **runmqsc**를 사용하여 큐 관리자의 속성을 표시하고 AMQPCAP(YES)를 검사하십시오.
- PCF를 사용하여 **MQCMD_INQUIRE_Q_MGR** 명령을 사용하고 MQIA_AMQP_CAPABILITY의 값을 확인하십시오.

관련 정보
strmqm
AMQP 클라이언트 애플리케이션 개발
AMQP 클라이언트 보안 설정

ULW 큐 관리자에서 AMQP 채널 제거

설치 디렉토리에서 폴더를 제거하여 큐 관리자에서 AMQP 채널을 제거할 수 있습니다.

프로시저

1. 큐 관리자를 중지합니다.
2. MQ Light API에 대한 IBM MQ 지원을 제거하십시오.
   - AIX AIX의 경우 다음 명령을 실행하십시오.

     ```
     installp -u mqm.amqp.rte
     ```

   - Linux Linux의 경우 AMQP RPM을 제거하십시오. 설치하기 전에 RPM을 재패키징한 경우 재패키징한 RPM의 이름을 지정하십시오.

     ```
     rpm -e MQSeriesAMQP
     ```

   - Windows Windows의 경우 IBM MQ 설치에서 amqp 폴더를 제거하십시오. IBM MQ 설치 경로에서 기타 파일 또는 폴더가 제거되지 않도록 하십시오.
3. 큐 관리자를 재시작하십시오.

관련 정보
AMQP 클라이언트 애플리케이션 개발
AMQP 클라이언트 보안 설정

ULW AMQP 채널 로그 파일

AMQP 채널의 로그 파일은 동일한 IBM MQ 데이터 디렉토리에 IBM MQ 로그 파일로 저장됩니다.

Windows의 기본 데이터 디렉토리는 `C:\ProgramData\IBM\MQ`입니다.

Linux의 기본 데이터 디렉토리는 `/var/mqm`입니다.

AMQP 채널은 IBM MQ 데이터 디렉토리에 있는 다음 로그인 파일에 로그 정보를 기록합니다.

- `amqp.stdout`은 qmgrs/*QM-name* 폴더에 기록됩니다.
- `amqp.stderr`은 qmgrs/*QM-name* 폴더에 기록됩니다.
- `amqp_*.log` 은 qmgrs/*QM-name*/`errors` 폴더에 기록됩니다.

MQ Light 클라이언트가 인증 또는 권한 부여 오류를 수신하면 관리자가 amqp_`0.log` 파일 및 MQ AMQERR`*.log` 파일에서 보안 실패 이유에 대한 자세한 정보를 찾을 수 있습니다.

FDC 파일은 *data-directory*/`errors` 폴더에 기록되는 AMQP`*.FDC` 파일로서 작성됩니다.

일부 구성 파일은 qmgrs/*QM-name*/amqp 디렉토리에 기록됩니다. 이 디렉토리의 파일을 편집할 필요가 없습니다.

관련 정보
UNIX, Linux, and Windows의 오류 로그
AMQP 클라이언트 애플리케이션 개발
AMQP 클라이언트 보안 설정

다른 플랫폼에서 서버 연결 및 클라이언트 연결 정의 작성

적용할 컴퓨터에서 각 채널 정의를 작성할 수 있습니다. 그러나 클라이언트 컴퓨터에서 채널 정의를 작성하는 방법에는 제한이 있습니다.

이 태스크 정보

모든 플랫폼에서 프로그래밍 가능 명령 형식(PCF) 명령인 MQSC(IBM MQ Script) 명령 또는 IBM MQ Explorer를 사용하여 서버 시스템에서 서버 연결 채널을 정의할 수 있습니다.

z/OS z/OS에서는 조작 및 제어 패널을 사용할 수도 있습니다.

IBM i IBM i에서는 패널 인터페이스를 사용할 수도 있습니다.

MQSC 명령은 IBM MQ가 IBM MQ MQI client 전용으로 설치된 시스템에서 사용 불가능하기 때문에 클라이언트 시스템에서는 다른 방식의 클라이언트 연결 채널 정의를 사용해야 합니다.

runmqsc를 사용할 경우 다음 고려사항이 적용됩니다.

- **-c** 매개변수를 지정하고 선택적으로 **-u** 매개변수를 지정하여 **runmqsc**를 클라이언트로 관리할 큐 관리자에 연결할 수 있습니다.
- **-u** 매개변수를 사용하여 사용자 ID를 제공하는 경우 일치하는 비밀번호를 입력하도록 프롬프트됩니다.
- `CHCKLOCL(REQUIRED)` 또는 `CHCKLOCL(REQDADM)`으로 `CONNAUTH AUTHINFO` 레코드를 구성한 경우 **-u** 매개변수를 사용해야 합니다. 그렇지 않으면 **runmqsc**로 큐 관리자를 관리할 수 없습니다.

프로시저

- 서버에서 서버 연결 채널을 정의하려면 33 페이지의 『서버에서 서버 연결 채널 정의』의 내용을 참조하십시오.
- IBM MQ MQI client에서 클라이언트 연결 채널을 작성하려면 34 페이지의 『IBM MQ MQI client에서 클라이언트 연결 채널 작성』의 내용을 참조하십시오.

서버에서 서버 연결 채널 정의

필요한 경우 MQSC를 시작한 다음 서버 연결 채널을 정의합니다.

프로시저

1. 옵션: 서버 플랫폼이 z/OS가 아닌 경우 먼저 큐 관리자를 작성하고 시작한 다음 MQSC 명령을 시작하십시오.
   a) 예를 들어, QM1이라고 하는 큐 관리자를 작성하십시오.

   ```
   crtmqm QM1
   ```

   b) 큐 관리자를 시작하십시오.

   ```
   strmqm QM1
   ```

   c) MQSC 명령을 시작하십시오.

   ```
   runmqsc QM1
   ```

2. 선택한 이름 및 채널 유형 서버 연결을 사용하여 채널을 정의하십시오.

   ```
   DEFINE CHANNEL(CHAN1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
   DESCR('Server-connection to Client_1')
   ```

   이 채널 정의는 서버에서 실행되는 큐 관리자와 연관됩니다.
3. 다음 명령을 사용하여 큐 관리자에 대한 인바운드 연결 액세스를 허용하십시오.

```
SET CHLAUTH(CHAN1) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- 여기서, SET CHLAUTH는 이전 단계에서 정의된 채널의 이름을 사용합니다.
- 여기서, *'IP address'*는 클라이언트의 IP 주소입니다.
- 여기서, *'userid'*는 대상 큐에 대한 액세스 제어를 위해 채널에 제공할 ID입니다. 이 필드는 대소문자를 구분합니다.

다양한 속성을 사용하여 인바운드 연결을 식별하도록 선택할 수 있습니다. 이 예는 IP 주소를 사용합니다. 대체 속성에는 클라이언트 사용자 ID 및 TLS 주제 식별 이름 등이 있습니다. 자세한 정보는 채널 인증 레코드의 내용을 참조하십시오.

IBM MQ MQI client에서 클라이언트 연결 채널 작성

MQCONNX 호출 시 MQSERVER 또는 MQCNO 구조를 사용하여 클라이언트 워크스테이션에서 클라이언트 연결 채널을 정의할 수 있습니다.

MQSERVER 사용

MQSERVER 환경 변수를 사용하여 클라이언트 연결 채널의 단순 정의를 지정할 수 있습니다. 이 방법을 사용하여 적은 수의 채널 속성만 지정할 수 있다는 면에서 단순 정의입니다.

- Windows에서 다음과 같이 단순 채널 정의를 지정하십시오.

```
SET MQSERVER=ChannelName/TransportType/ConnectionName
```

- UNIX and Linux 시스템에서 다음과 같이 단순 채널 정의를 지정하십시오.

```
export MQSERVER=ChannelName/TransportType/ConnectionName
```

- IBM i 시스템에서 다음과 같이 단순 채널 정의를 지정하십시오.

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('ChannelName/TransportType/ConnectionName')
```

설명:

- `ChannelName`은 서버에서 정의되는 이름과 같은 이름이어야 합니다. 슬래시는 포함시킬 수 없습니다.
- `TransportType`은 IBM MQ MQI client 플랫폼에 따라 다음 값 중 하나일 수 있습니다.
  - LU62
  - TCP
  - NETBIOS
  - SPX

  참고: UNIX and Linux 시스템에서 TransportType은 대소문자를 구분하고 대문자여야 합니다. TransportType이 인식되지 않는 경우 `MQCONN` 또는 `MQCONNX` 호출이 2058을 리턴합니다.
- `ConnectionName`은 통신 프로토콜(TransportType)에 정의된 서버의 이름입니다.

예를 들어, Windows의 경우는 다음과 같습니다.

```
SET MQSERVER=CHANNEL1/TCP/MCID66499
```

또는 UNIX and Linux 시스템의 경우

```
export MQSERVER=CHANNEL1/TCP/'MCID66499'
```

참고: TCP/IP 포트 번호를 변경하려면 74 페이지의 『MQSERVER』의 내용을 참조하십시오.

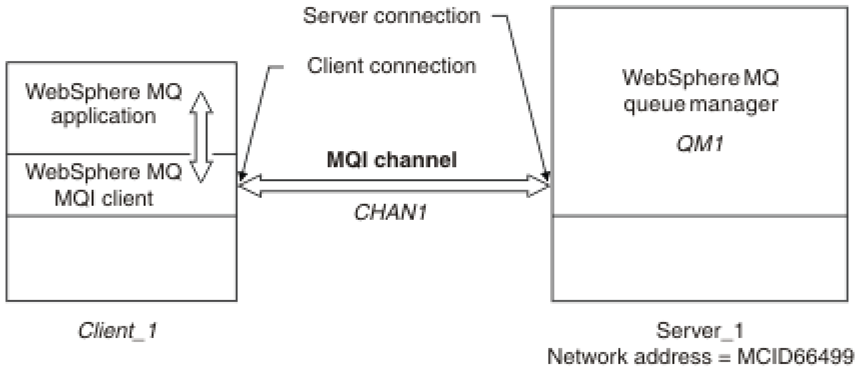


그림 *1.* 단순 채널 정의

단순 채널 정의의 몇 가지 다른 예는 다음과 같습니다.

- Windows:

```
SET MQSERVER=CHANNEL1/TCP/9.20.4.56
SET MQSERVER=CHANNEL1/NETBIOS/BOX643
```

- UNIX and Linux 시스템의 경우:

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56'
export MQSERVER=CHANNEL1/LU62/BOX99
```

- IBM i IBM i:

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('CHANNEL1/TCP/9.20.4.56(1416)')
```

여기서, BOX99는 LU 6.2 ConnectionName입니다.

IBM MQ MQI client에서 모든 **MQCONN** 또는 **MQCONNX** 요청은 **MQCONNX**에 제공된 MQCNO 구조에서 참조된 MQCD 구조에서 채널이 대체되지 않는 한 사용자가 정의한 채널을 사용하려고 시도합니다.

참고: *MQSERVER* 환경 변수에 대한 자세한 정보는 74 페이지의 『MQSERVER』의 내용을 참조하십시오.

MQCONNX 호출 시 MQCNO 구조 사용

IBM MQ MQI client 애플리케이션은 **MQCONNX** 호출에서 연결 옵션 구조인 MQCNO를 사용하여 클라이언트 연결 채널의 정의를 포함하는 채널 정의 구조 MQCD를 참조할 수 있습니다.

이 방식으로 클라이언트 애플리케이션은 클라이언트 애플리케이션이 다중 서버 큐 관리자에 동시에 연결할 수 있도록 하는 채널의 **ChannelName**, **TransportType** 및 **ConnectionName** 속성을 런타임 시 지정할 수 있습니다.

MQSERVER 환경 변수를 사용하여 채널을 정의하는 경우 **ChannelName**, **TransportType** 및 **ConnectionName** 속성을 런타임 시 지정할 수 없음을 참고하십시오.

클라이언트 애플리케이션은 **MaxMsgLength** 및 **SecurityExit**와 같은 채널 속성을 지정할 수도 있습니다. 그러한 속성을 지정하면 클라이언트 애플리케이션이 속성에 기본값이 아닌 값을 지정할 수 있으며 채널 엑시트 프로그램을 MQI 채널의 클라이언트 측에서 호출할 수 있습니다.

채널이 TLS(Transport Layer Security)를 사용할 경우 클라이언트 애플리케이션은 MQCD 구조에 있는 TLS와 관련된 정보도 제공할 수 있습니다.TLS와 관련된 추가 정보는 TLS 구성 옵션 구조인 MQSCO에서 제공될 수 있으며 이 구조는 **MQCONNX** 호출 시 MQCNO 구조라고도 합니다.

MQCNO, MQCD 및 MQSCO 구조에 대하 자세한 정보는 MQCNO, MQCD 및 MQSCO를 참조하십시오.

참고: MQCONNX에 대한 샘플 프로그램은 **amqscnxc**라고 합니다. **amqssslc**라고 하는 다른 샘플 프로그램은 MQSCO 구조의 사용을 보여줍니다.

서버에서 서버 연결 및 클라이언트 연결 정의 작성

서버에서 두 정의 모두 작성한 다음 클라이언트가 클라이언트 연결 정의를 사용할 수 있도록 설정할 수 있습니다.

이 태스크 정보

먼저 서버 연결 채널을 정의한 다음 클라이언트 연결 채널을 정의하십시오.

- 모든 플랫폼에서 프로그래밍 가능 명령 형식(PCF) 명령인 MQSC(IBM MQ Script) 명령을 사용하여 서버 시스템에서 서버 연결 채널을 정의할 수 있습니다.
- Windows Linux Linux 및 Windows에서는 IBM MQ Explorer를 사용할 수도 있습니다.
- z/OS z/OS에서는 조작 및 제어 패널을 사용할 수도 있습니다.
- IBM i IBM i에서는 패널 인터페이스를 사용할 수도 있습니다.

서버에서 작성된 클라이언트 연결 채널 정의를 클라이언트 채널 정의 테이블(CCDT)을 사용하여 클라이언트가 사용할 수 있도록 설정합니다.

프로시저

1. 서버 연결 채널을 정의하려면 40 페이지의 『서버에서 서버 연결 채널 정의』의 내용을 참조하십시오.
2. 클라이언트 연결 채널을 정의하려면 41 페이지의 『서버에서 클라이언트 연결 채널 정의』의 내용을 참조하십시오.

관련 개념

36 페이지의 『클라이언트 채널 정의 테이블』
클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

관련 태스크

40 페이지의 『서버에서 서버 연결 채널 정의』
큐 관리자에 대한 서버 연결 채널 정의를 작성하십시오.

41 페이지의 『서버에서 클라이언트 연결 채널 정의』
서버 연결 채널을 정의했으면 이제 해당 클라이언트 연결 채널을 정의합니다.

42 페이지의 『클라이언트 연결 채널 정의 액세스』
클라이언트 채널 정의 테이블(CCDT)을 복사하거나 공유하여 클라이언트 애플리케이션에게 사용 가능하게 한 후, 클라이언트 컴퓨터에 해당 위치 및 이름을 지정할 수 있습니다. V 9.0.0 IBM MQ 9.0에서 제품은 URL 을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾는 기능도 제공합니다.

클라이언트 채널 정의 테이블

클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

클라이언트 채널 정의 테이블(CCDT)의 용도는 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 채널 정의를 판별하는 것입니다. 채널 정의는 연결에 적용되는 인증 정보도 지정합니다.

CCDT는 2진 파일입니다. CCDT는 큐 관리자에 의해 생성됩니다. 큐 관리자는 CCDT 파일을 읽지 않습니다.

Multi 멀티플랫폼에서 큐 관리자가 작성될 때 CCDT가 작성됩니다. 큐 관리자와 연관된 CCDT는 오브젝트 정의와 동기화되어 있습니다. 따라서 클라이언트 채널 오브젝트를 정의, 변경 또는 삭제할 경우 큐 관리자 오브젝트 정의 및 CCDT의 항목이 동일한 작업 중에 업데이트됩니다.

참고사항:

- IBM MQ CCDT 파일의 디자인은 사용자가 정의하는 모든 클라이언트 연결 채널이 실제로 정의된 후에만 CCDT 파일을 축소하는 것입니다. 클라이언트 연결 채널이 삭제되면 이는 CCDT 파일에서 삭제됨으로 표시되지만 실제로 제거되지는 않습니다.
- CCDT 파일을 강제로 축소시키려면 하나 이상의 클라이언트 연결 채널을 삭제한 후 다음 명령을 실행하십시오.

```
rcrmqobj -m QM80 -t clchltab
```

CCDT를 사용하여 클라이언트에 TLS 인증서 폐기를 검사하기 위한 인증 정보를 제공할 수 있습니다. 인증 정보 오브젝트를 포함하는 이름 목록을 정의하고 큐 관리자 속성 **SSLCRLNameList**를 이름 목록의 이름으로 설정하십시오.

Default CCDT AMQCLCHL.TAB

Multi

멀티플랫폼에서 큐 관리자를 작성할 때 AMQCLCHL.TAB이라는 기본 CCDT가 작성됩니다.

기본적으로 AMQCLCHL.TAB는 서버의 다음 디렉토리에 위치합니다.

- IBM i 통합 파일 시스템에서 IBM i의 경우:

  ```
  /QIBM/UserData/mqm/qmgrs/QUEUEMANAGERNAME/&ipcc
  ```

- Linux UNIX UNIX and Linux 시스템의 경우:

  ```
  /prefix/qmgrs/QUEUEMANAGERNAME/@ipcc
  ```

  *QUEUEMANAGERNAME*으로 참조되는 디렉토리의 이름은 UNIX and Linux 시스템에서 대소문자를 구분합니다. 이 디렉토리 이름은 큐 관리자 이름에 특수 문자가 있을 경우 큐 관리자 이름과 같지 않을 수 있습니다.

- Windows Windows:

  ```
  MQ_INSTALLATION_PATH\data\qmgrs\QUEUEMANAGERNAME\@ipcc
  ```

*MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

그러나 큐 관리자 데이터에 대해 다른 디렉토리 사용을 선택했을 수 있습니다. **crtmqm** 명령을 사용할 때 매개변수 **-md** *DataPath*를 지정할 수 있습니다. 이 경우 AMQCLCHL.TAB은 사용자가 지정한 *DataPath*의 @ipcc 디렉토리에 있습니다.

CCDT에 대한 경로는 MQCHLLIB를 설정하여 변경할 수 있습니다. MQCHLLIB를 설정하지 않는 경우 동일한 서버에 여러 큐 관리자가 있으면 이러한 큐 관리자가 동일한 CCDT 위치를 공유함을 알아야 합니다.

CCDT는 큐 관리자가 작성될 때 작성됩니다. CCDT의 각 입력 항목은 특정 큐 관리자에 대한 클라이언트 연결을 나타냅니다. **DEFINE CHANNEL** 명령을 사용하여 클라이언트 연결 채널을 정의할 때 새 입력 항목이 추가되며 이 입력 항목은 **ALTER CHANNEL** 명령을 사용하여 클라이언트 연결 채널을 대체할 때 업데이트됩니다.

클라이언트 채널 정의 테이블 위치

클라이언트 애플리케이션에서 CCDT를 사용하는 여러 가지 방법이 있습니다. CCDT를 클라이언트 컴퓨터에 복사할 수 있습니다. CCDT를 둘 이상의 클라이언트가 공유하는 위치에 복사할 수 있습니다. CCDT가 서버에 있는 상태에서 클라이언트가 공유 파일로서 CCDT에 액세스할 수 있게 설정할 수 있습니다.

파일을 복사하는 데 FTP를 사용하는 경우 bin 옵션을 사용하여 2진 모드를 설정하고 기본 ASCII 모드를 사용하지 마십시오. CCDT를 사용할 수 있도록 설정하기 위해 선택하는 메소드에 관계없이 채널을 권한 없이 변경하지 못하도록 그 위치를 보안 설정해야 합니다.

V 9.0.0 IBM MQ 9.0부터 URI를 통해 액세스할 수 있는 중심 위치에서 CCDT를 호스팅할 수 있어 배치된 클라이언트마다 CCDT를 개별적으로 업데이트하지 않아도 됩니다. IBM MQ 9.0에서는 로컬 파일, ftp 또는 http

자원인지 여부와 상관없이 URL에서 CCDT를 가져오는 기본(C/C++, COBOL 및 RPG) 및 관리되지 않는 .NET 애플리케이션에 대한 기능을 추가합니다.

주의: IBM MQ는 파일, ftp 또는 http URL에서 CCDT 검색을 지원합니다.

V 9.0.0 IBM MQ 클라이언트의 기본 캐싱 동작은 파일 수정 시간이 파일을 마지막으로 검색한 시간과 다를 경우 CCDT 파일만 가져오는 것입니다. 대부분의 클라이언트 구성 옵션처럼 URL 위치를 제공할 수 있는 다양한 방법이 있습니다.

- MQCONNX MQI 호출에 전달되는 MQCNO 구조를 통한 CCDTUrlPtr/CCDTUrlOffset
- MQCCDTURL 환경 변수
- `mqclient.ini`의 채널 스탠자에 있는 ChannelDefinitionDirectory 속성

V 9.0.0 인증되거나 인증되지 않은 URL 모두가 지원됩니다. 다음은 일부 예입니다.

```
export MQCCDTURL=ftp://myuser:password@myhost.sample.com//var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
```

```
export MQCCDTURL=http://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
```

V 9.0.0 이 지원을 ftp 또는 http와 함께 사용할 경우 CCDT 파일을 여전히 서버에서 호스팅해야 함을 의미하지만, IBM MQ 9.0에서 추가된 지원을 사용하면 업데이트를 수동으로 푸시하거나 각 클라이언트에서 네트워크된 파일 시스템을 마운트하지 않고도 채널 정의에 대한 변경사항을 모든 클라이언트 애플리케이션이 자동으로 선택할 수 있습니다. 추가 정보는 39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』의 내용을 참조하십시오.

runmqsc를 사용하여 클라이언트 시스템에서 직접 CCDT를 작성하는 방법

IBM MQ 8.0부터는 runmqsc 명령을 **-n** 매개변수와 함께 사용하여 클라이언트 시스템에서 직접 CCDT를 작성할 수 있습니다. 이 CCDT는 MQCHLLIB에 의해 표시되는 위치에서 MQCHLTAB에 의해 표시되는 파일 이름으로 작성되며 기본적으로 AMQCLCHL.TAB입니다.

중요사항: -n 매개변수를 지정하는 경우 다른 매개변수를 지정하지 않아야 합니다.

CCDT의 각 입력 항목은 특정 큐 관리자에 대한 클라이언트 연결을 나타냅니다. **DEFINE CHANNEL** 명령을 사용하여 클라이언트 연결 채널을 정의할 때 새 입력 항목이 추가되며 이 입력 항목은 **ALTER CHANNEL** 명령을 사용하여 클라이언트 연결 채널을 대체할 때 업데이트됩니다.

클라이언트에서 CCDT의 위치를 지정하는 방법

클라이언트 시스템에서 다음 방법으로 CCDT의 위치를 지정할 수 있습니다.

- 환경 변수 MQCHLLIB를 사용하여 테이블이 위치하는 디렉토리를 지정하고 MQCHLTAB를 사용하여 테이블의 파일 이름을 지정합니다.
- 클라이언트 구성 파일 사용. CHANNELS 스탠자에서 속성 ChannelDefinitionDirectory를 사용하여 테이블이 위치하는 디렉토리를 지정하고 ChannelDefinitionFile을 사용하여 파일 이름을 지정합니다.
- V 9.0.0 중앙 위치에서 호스팅되는 CCDT에 대한 URL(파일, ftp 또는 http) 제공(37 페이지의 『클라이언트 채널 정의 테이블 위치』 참조).

위치가 클라이언트 구성 파일에 지정되고 동시에 환경 변수를 사용하여 지정된 경우, 환경 변수가 우선순위를 가집니다. 이 기능을 사용하여 클라이언트 구성 파일에 표준 위치를 지정하고 필요하면 환경 변수를 사용하여 이를 대체할 수 있습니다.

V 9.0.0 URL을 사용하여 CCDT 위치를 제공하는 경우 기본 클라이언트 애플리케이션이 클라이언트 채널 정의를 찾는 우선 순위는 39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』에 설명되어 있습니다.

관련 참조

70 페이지의 『MQCHLLIB』

MQCHLLIB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일에 대한 디렉토리 경로를 지정합니다. 파일은 서버에서 작성되지만, IBM MQ MQI client 워크스테이션에서 복사될 수 있습니다.

관련 정보

폐기된 인증서에 대한 작업

V 9.0.0 클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스

IBM MQ 9.0에서 제품은 MQCNO를 사용하여 프로그래밍하거나 환경 변수를 사용하거나 mqclient.ini 파일 스탠자를 사용하여 URL을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾을 수 있는 기능을 제공합니다.

주의: 클라이언트로 연결되는 고유 프로그램(즉, C, COBOL 또는 C++ 애플리케이션)에 대해서만 환경 변수 옵션을 사용할 수 있습니다. 환경 변수는 Java, JMS 또는 관리 .NET 애플리케이션에 영향을 주지 않습니다.

IBM MQ는 파일, ftp 또는 http URL에서 CCDT 검색을 지원합니다.

환경 변수 69 페이지의 『MQCCDTURL』에서는 클라이언트 채널 정의 테이블을 확보할 수 있는 단일 값으로 파일, ftp 또는 http URL을 제공할 수 있습니다.

또한 70 페이지의 『MQCHLLIB』(또는 56 페이지의 『클라이언트 구성 파일의 CHANNELS 스탠자』에서 **ChannelDefinitionDirectory**에 의해 지정된 값)를 사용하여 기존 로컬 파일 시스템 디렉토리(즉, /var/mqm) 외에도 파일, ftp 또는 http URL을 통해 CCDT 파일을 찾을 수 있습니다.

70 페이지의 『MQCHLLIB』 값은 디렉토리 시스템이며, 완전한 URL을 파생시키기 위해 72 페이지의 『MQCHLTAB』와 함께 사용합니다.

연결에서 기본 인증은 URL에 인코딩된 신임 정보를 통해 지원됩니다.

인증된 연결

```
export MQCHLLIB=ftp://myuser:password@myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLLIB=http://myuser:password@myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
```

인증되지 않은 연결

```
export MQCHLLIB=ftp://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLLIB=http://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLLIB=file:///var/mqm/qmgrs/QMGR/@ipcc
```

참고: 인증된 연결을 사용하려면 JMS의 경우와 마찬가지로 URL에 인코딩된 사용자 이름 및 비밀번호를 제공해야 합니다.

클라이언트 채널 정의를 찾기 위한 기본 클라이언트 애플리케이션의 우선순위 순서는 다음과 같습니다.

1. MQCNO에서 **ClientConnOffset** 및 **ClientConnPtr**에 의해 제공된 MQCD.
2. MQCNO에서 **CCDTUrlOffset** 및 **CCDTUrlPtr**에 의해 제공된 URL.
3. 74 페이지의 『MQSERVER』 환경 변수.
4. mqclient.ini 파일이 정의되어 있고 ServerConnectionParms를 포함하고 있는 경우에는 해당 파일이 정의하는 채널이 사용됩니다. 자세한 정보는 48 페이지의 『구성 파일을 사용하여 클라이언트 구성』 및 56 페이지의 『클라이언트 구성 파일의 CHANNELS 스탠자』의 내용을 참조하십시오.
5. 69 페이지의 『MQCCDTURL』 환경 변수.
6. 70 페이지의 『MQCHLLIB』 및 72 페이지의 『MQCHLTAB』 환경 변수.
7. 56 페이지의 『클라이언트 구성 파일의 CHANNELS 스탠자』의 **ChannelDefinitionDirectory**.

중요사항: URL을 사용하는 CCDT 파일에 대한 액세스는 항상, file:// 프로토콜을 사용할 때에도, 파일의 읽기 전용 사본을 엽니다.

예를 들어, 클라이언트에서 **runmqsc** DEFINE CHANNEL 명령을 사용하는 경우 쓰기 액세스를 위해 CCDT 파일을 열려고 하면 쓰기 액세스를 위해 파일을 열 수 없음을 나타내는 오류 메시지가 리턴됩니다.

그러나 **runmqsc**를 사용하여 채널 및 인증 정보 정의를 읽을 수 있습니다.

관련 개념

36 페이지의 『클라이언트 채널 정의 테이블』

클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

관련 태스크

42 페이지의 『클라이언트 연결 채널 정의 액세스』

클라이언트 채널 정의 테이블(CCDT)을 복사하거나 공유하여 클라이언트 애플리케이션에게 사용 가능하게 한 후, 클라이언트 컴퓨터에 해당 위치 및 이름을 지정할 수 있습니다. V 9.0.0 IBM MQ 9.0에서 제품은 URL을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾는 기능도 제공합니다.

관련 정보

CCDTURL

IBM MQ classes for JMS에서 CCDT 사용

XMSC_WMQ_CCDTURL

Windows Active Directory의 클라이언트 연결 채널

Active Directory를 지원하는 Windows 시스템에서, IBM MQ는 동적 클라이언트-서버 바인딩을 제공하기 위해 Active Directory에서 클라이언트 연결 채널을 공개합니다.

클라이언트 연결 채널 오브젝트가 정의되면 이는 기본적으로 클라이언트 채널 정의 파일 AMQCLCHL.TAB에 기록됩니다. 클라이언트 연결 채널이 TCP/IP 프로토콜을 사용하는 경우 IBM MQ 서버는 Active Directory에서 해당 프로토콜도 공개합니다. IBM MQ 클라이언트가 서버에 연결할 방법을 판별할 때 다음 검색 순서를 사용하여 관련 클라이언트 채널 오브젝트 정의를 찾습니다.

1. MQCONNX MQCD 데이터 구조
2. MQSERVER 환경 변수
3. 클라이언트 채널 정의 파일
4. Active Directory

이 순서는 모든 현재 애플리케이션은 어떤 변경의 영향도 받지 않음을 의미합니다. Active Directory의 이들 입력 항목을 클라이언트 채널 정의 파일의 레코드로 생각할 수 있으며, IBM MQ 클라이언트는 동일한 방법으로 이들을 처리합니다. Active Directory에서 클라이언트 연결 채널 정의의 공개를 위한 지원을 구성 및 관리하려면 setmqscp에서 설명하는 대로 `setmqscp` 명령을 사용하십시오.

서버에서 서버 연결 채널 정의

큐 관리자에 대한 서버 연결 채널 정의를 작성하십시오.

프로시저

1. 서버 시스템에서 선택한 이름과 서버 연결 채널 유형으로 채널을 정의하십시오.
   예를 들면 다음과 같습니다.

   ```
   DEFINE CHANNEL(CHAN2) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
   DESCR('Server-connection to Client_2')
   ```

2. 다음 명령을 사용하여 큐 관리자에 대한 인바운드 연결 액세스를 허용하십시오.

   ```
   SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
   ```

   - 여기서, SET CHLAUTH는 이전 단계에서 정의된 채널의 이름을 사용합니다.
   - 여기서, *'IP address'* IP 주소는 클라이언트의 IP 주소입니다.
   - 여기서, *'userid'*는 대상 큐에 대한 액세스 제어를 위해 채널에 제공할 ID입니다. 이 필드는 대소문자를 구분합니다.

다양한 속성을 사용하여 인바운드 연결을 식별하도록 선택할 수 있습니다. 이 예는 IP 주소를 사용합니다. 대체 속성에는 클라이언트 사용자 ID 및 TLS 주제 식별 이름 등이 있습니다.자세한 정보는 채널 인증 레코드의 내용을 참조하십시오.

이 채널 정의는 서버에서 실행되는 큐 관리자와 연관됩니다.

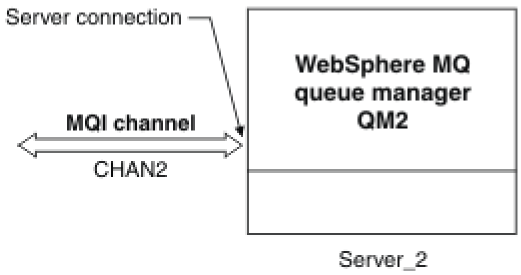


그림 *2.* 서버 연결 채널 정의

서버에서 클라이언트 연결 채널 정의

서버 연결 채널을 정의했으면 이제 해당 클라이언트 연결 채널을 정의합니다.

시작하기 전에

서버 연결 채널을 정의합니다.

프로시저

1. 서버 연결 채널과 이름은 같지만 채널 유형은 클라이언트 연결인 채널을 정의하십시오. 연결 이름(CONNAME)을 명시해야 합니다. TCP/IP의 경우 연결 이름은 서버 시스템의 네트워크 주소 또는 호스트 이름입니다. 클라이언트 환경에서 실행되는 IBM MQ 애플리케이션이 연결할 큐 관리자 이름(QMNAME)을 지정하는 것도 권장됩니다. 큐 관리자 이름을 변경하여 여러 큐 관리자에 연결하기 위한 일련의 채널을 정의할 수 있습니다.

```
DEFINE CHANNEL(CHAN2) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(9.20.4.26) QMNAME(QM2) DESCR('Client-connection to Server_2')
```

2. 다음 명령을 사용하여 큐 관리자에 대한 인바운드 연결 액세스를 허용하십시오.

```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP-address') MCAUSER('userid')
```

- 여기서, SET CHLAUTH는 이전 단계에서 정의된 채널의 이름을 사용합니다.
- 여기서, *'IP address'*는 클라이언트의 IP 주소입니다.
- 여기서, *'userid'*는 대상 큐에 대한 액세스 제어를 위해 채널에 제공할 ID입니다. 이 필드는 대소문자를 구분합니다.

다양한 속성을 사용하여 인바운드 연결을 식별하도록 선택할 수 있습니다. 이 예는 IP 주소를 사용합니다. 대체 속성에는 클라이언트 사용자 ID 및 TLS 주제 식별 이름 등이 있습니다.자세한 정보는 채널 인증 레코드의 내용을 참조하십시오.

결과

Multi 멀티플랫폼에서 이 채널 정의는 클라이언트 채널 정의 테이블(CCDT)이라는 파일에 저장되고 이 파일은 큐 관리자와 연관됩니다. 클라이언트 채널 정의 테이블에는 둘 이상의 클라이언트 연결 채널 정의가 포함될 수 있습니다. 클라이언트 채널 정의 테이블에 대한 자세한 정보 및 클라이언트 연결 채널 정의가 z/OS에서 저장되는 방식에 대한 해당 정보는 36 페이지의 『클라이언트 채널 정의 테이블』의 내용을 참조하십시오.

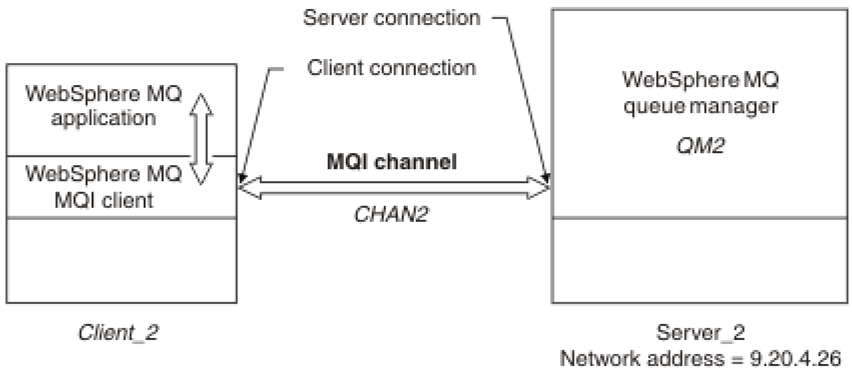


그림 3. 클라이언트 연결 채널 정의

클라이언트 연결 채널 정의 액세스

클라이언트 채널 정의 테이블(CCDT)을 복사하거나 공유하여 클라이언트 애플리케이션에게 사용 가능하게 한 후, 클라이언트 컴퓨터에 해당 위치 및 이름을 지정할 수 있습니다. V 9.0.0 IBM MQ 9.0에서 제품은 URL을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾는 기능도 제공합니다.

시작하기 전에

필요한 클라이언트 연결 채널을 정의했습니다.

z/OS z/OS에서 CCDT를 작성했습니다.

Multi 멀티플랫폼에서 CCDT가 자동으로 작성되고 업데이트됩니다.

이 태스크 정보

클라이언트 애플리케이션이 클라이언트 채널 정의 테이블(CCDT)를 사용하려면 CCDT를 클라이언트 애플리케이션이 사용할 수 있게 하고 해당 위치 및 이름을 지정해야 합니다. 이를 수행하는 방법에는 여러 가지가 있습니다.

- CCDT를 클라이언트 컴퓨터에 복사할 수 있습니다.
- CCDT를 둘 이상의 클라이언트가 공유하는 위치에 복사할 수 있습니다.
- CCDT가 서버에 있는 상태에서 클라이언트가 공유 파일로서 CCDT에 액세스할 수 있게 설정할 수 있습니다.

V 9.0.0 IBM MQ 9.0부터, IBM MQ, 기본(C/C++, COBOL 및 RPG) 및 비관리 .NET 애플리케이션은 로컬 파일, ftp 또는 http 자원인지 여부와 상관없이 중앙 위치에서 호스팅되는 CCDT를 URL에서 가져올 수 있습니다.

프로시저

1. 다음 방법 중 하나로 클라이언트 애플리케이션이 CCDT를 사용할 수 있도록 설정하십시오.
   a) 옵션: CCDT를 클라이언트 컴퓨터에 복사하십시오.
   b) 옵션: CCDT를 둘 이상의 클라이언트가 공유하는 위치에 복사하십시오.
   c) 옵션: 서버에 CCDT를 그대로 두고 클라이언트가 공유할 수 있도록 설정하십시오.
   d) V 9.0.0
   옵션: 기본(C/C++, COBOL 및 RPG) 및 관리되지 않는 .NET 애플리케이션이 이 URL에서 CCDT를 가져올 수 있도록 중앙 위치에서 호스팅되는 CCDT에 대한 로컬 파일, ftp 또는 http URL을 정의하십시오.

   CCDT에 대해 선택하는 위치에 관계없이 채널을 권한 없이 변경하지 못하도록 그 위치를 보안 설정해야 합니다.
2. 클라이언트에서 다음 세 가지 방식 중 하나로 CCDT를 포함하는 파일의 위치 및 이름을 지정하십시오.
   a) 옵션: 클라이언트 구성 파일의 CHANNELS 스탠자를 사용하십시오. 추가 정보는 56 페이지의 『클라이언트 구성 파일의 CHANNELS 스탠자』의 내용을 참조하십시오.

b) 옵션: 환경 변수 MQCHLLIB 및 MQCHLTAB를 사용하십시오.

예를 들어, 다음을 입력하여 환경 변수를 설정할 수 있습니다.

- UNIX and Linux 시스템에서:

```
export MQCHLLIB= MQ_INSTALLATION_PATH/qmgrs/ QUEUEMANAGERNAME /@ipcc
export MQCHLTAB=AMQCLCHL.TAB
```

- IBM i IBM i에서:

```
ADDENVVAR ENVVAR(MQCHLLIB) VALUE('/QIBM/UserData/mqm/qmgrs/QUEUEMANAGERNAME/@ipcc')
ADDENVVAR ENVVAR(MQCHLTAB) VALUE(AMQCLCHL.TAB)
```

여기서 *MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

c) 옵션: Windows에서만 **setmqscp** 제어 명령을 사용하여 클라이언트 연결 채널 정의를 Active Directory에서 발행하십시오.

d) V 9.0.0

MQCNO, 환경 변수 또는 mqclient.ini 파일 스탠자를 사용하여 프로그래밍하여 URL을 통해 중앙에서 호스팅되는 CCDT의 위치를 제공하십시오. 자세한 정보는 37 페이지의 『클라이언트 채널 정의 테이블 위치』 및 39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』의 내용을 참조하십시오.

MQSERVER 환경 변수가 설정되는 경우 IBM MQ 클라이언트는 클라이언트 채널 정의 테이블의 어떤 정의보다도 우선하여 MQSERVER에 의해 지정되는 클라이언트 연결 채널 정의를 사용합니다.

관련 개념

36 페이지의 『클라이언트 채널 정의 테이블』
클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』
IBM MQ 9.0에서 제품은 MQCNO를 사용하여 프로그래밍하거나 환경 변수를 사용하거나 mqclient.ini 파일 스탠자를 사용하여 URL을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾을 수 있는 기능을 제공합니다.

관련 정보

MQI 클라이언트: 클라이언트 채널 정의 테이블(CCDT)

ULW MQI 채널에 대한 채널 엑시트 프로그램

UNIX, Linux, and Windows의 IBM MQ MQI client 환경에서는 세 가지 유형의 채널 엑시트를 사용할 수 있습니다.

즉, 다음과 같습니다.

- 송신 엑시트
- 수신 엑시트
- 보안 엑시트

이러한 엑시트는 채널의 클라이언트 및 서버 측에서 사용 가능합니다. MQSERVER 환경 변수를 사용하는 경우에는 애플리케이션이 엑시트를 사용할 수 없습니다. 채널 엑시트는 메시징 채널의 채널 엑시트 프로그램에 설명되어 있습니다.

송신 및 수신 엑시트는 함께 작동합니다. 다음과 같이 엑시트를 사용하는 여러 가능한 방법이 있습니다.

- 메시지 분할 및 리어셈블링
- 메시지의 데이터 압축 및 압축 풀기(이 기능은 IBM MQ의 일부로 제공되지만 다른 압축 기술을 사용할 수도 있음)

- 사용자 데이터 암호화 및 복호화(이 기능은 IBM MQ의 일부로 제공되지만 다른 암호화 기술을 사용할 수도 있음)
- 송신 및 수신된 각 메시지 저널링

보안 엑시트를 사용하여 IBM MQ 클라이언트 및 서버가 올바르게 식별되는지 확인하고 액세스를 제어할 수 있습니다.

채널 인스턴스의 서버 연결 측 송신 또는 수신 엑시트가 연관된 연결에 대해 MQI 호출을 수행해야 하는 경우 해당 엑시트는 MQCXP Hconn 필드에 제공되는 연결 핸들을 사용합니다. 클라이언트 연결 송신 및 수신 엑시트가 MQI 호출을 수행할 수 없음을 알고 있어야 합니다.

관련 개념

44 페이지의 『클라이언트 연결의 보안 엑시트』
보안 엑시트 프로그램을 사용하여 채널의 다른 끝에 있는 파트너가 진짜인지 확인할 수 있습니다. 보안 엑시트가 클라이언트 연결에 적용될 때 특수 고려사항이 적용됩니다.

관련 참조

44 페이지의 『엑시트 경로』
채널 엑시트의 위치에 대한 기본 경로는 클라이언트 구성 파일에서 정의됩니다. 채널 엑시트는 채널이 초기화될 때 로드됩니다.

46 페이지의 『송신 또는 수신 엑시트 프로그램에서 API 호출 식별』
클라이언트에 MQI 채널을 사용하는 경우 에이전트 버퍼의 바이트 10은 송신 또는 수신 엑시트가 호출될 때 사용 중인 API 호출을 식별합니다. 이는 사용자 데이터를 포함하고 암호화 또는 디지털 서명과 같은 처리를 요구할 수 있는 채널 플로우를 식별하는 데 유용합니다.

관련 정보

큐 관리자 기능 확장
사용자 엑시트, API 엑시트 및 IBM MQ 설치 가능 서비스

ULW 엑시트 경로

채널 엑시트의 위치에 대한 기본 경로는 클라이언트 구성 파일에서 정의됩니다. 채널 엑시트는 채널이 초기화될 때 로드됩니다.

UNIX, Linux, and Windows 시스템에서 클라이언트 구성 파일은 IBM MQ MQI client 설치 중에 시스템에 추가됩니다. 클라이언트에서 채널 엑시트의 위치에 대한 기본 경로는 다음과 같은 스탠자를 사용하여 이 파일에서 정의됩니다.

```
ClientExitPath:
ExitsDefaultPath= string
ExitsDefaultPath64= string
```

여기서, *string*은 플랫폼에 적합한 형식의 파일 위치입니다.

채널이 초기화되면 MQCONN 또는 MQCONNX 호출 후에 클라이언트 구성 파일이 검색됩니다. ClientExitPath 스탠자를 읽고 채널 정의에서 지정되는 모든 채널 엑시트가 로드됩니다.

ULW 클라이언트 연결의 보안 엑시트

보안 엑시트 프로그램을 사용하여 채널의 다른 끝에 있는 파트너가 진짜인지 확인할 수 있습니다. 보안 엑시트가 클라이언트 연결에 적용될 때 특수 고려사항이 적용됩니다.

46 페이지의 그림 4은 클라이언트 연결에서 사용자를 인증하기 위해 IBM MQ 오브젝트 권한 관리자를 사용하는 보안 엑시트의 사용을 보여줍니다. SecurityParmsPtr 또는 SecurityParmsOffset이 클라이언트에서 MQCNO 구조로 설정되고 채널 양 끝에는 보안 엑시트가 있습니다. 정상 보안 메시지 교환이 종료되고 채널이 실행될 준비가 되고 나면 MQCXP SecurityParms 필드에서 액세스된 MQCSP 구조가 클라이언트의 보안 엑시트에 전달됩니다. 엑시트 유형은 MQXR_SEC_PARMS로 설정됩니다. 보안 엑시트는 사용자 ID 및 비밀번호를 그대로 두도록 선택하거나 이 둘 중 하나 또는 둘 다를 대체할 수 있습니다. 그러면 엑시트에서 리턴된 데이터가 채널의 서버 연결 측에 송신됩니다. MQCSP 구조가 채널의 서버 연결 측에서 다시 빌드되어 MQCXP SecurityParms 필드에서 액세스되는 서버 연결 보안 엑시트에 전달됩니다. 보안 엑시트는 이 데이터를 수신하고 처리합니다. 이 처리는 일반적으로 클라이언트 엑시트에서 큐 관리자 연결 권한 부여에 사용되는 사용자 ID 및 비밀번호 필드에 대해 수행된

변경을 되돌리는 것입니다. 결과로 생성되는 MQCSP 구조는 큐 관리자 시스템의 MQCNO 구조에서 SecurityParmsPtr을 사용하여 참조됩니다.

MQCXP SecurityParms 필드에 의해 다시 전달되는 메모리 주소는 MQXR_TERM까지 주소 지정 가능하고 변경되지 않은 상태를 유지해야 합니다. MQXR_TERM에 대해 엑시트가 호출되기 전에 엑시트에서는 메모리를 무효화하거나 다시 시스템에 사용 가능하게 해서는 안 됩니다.

SecurityParmsPtr 또는 SecurityParmsOffset이 MQCNO 구조에서 설정되고 채널의 한쪽 끝에만 보안 엑시트가 있는 경우 이 보안 엑시트는 MQCSP 구조를 수신하고 처리합니다. 암호화와 같은 조치는 보완하는 조치를 수행할 엑시트가 없는 단일 사용자 엑시트에는 적합하지 않습니다.

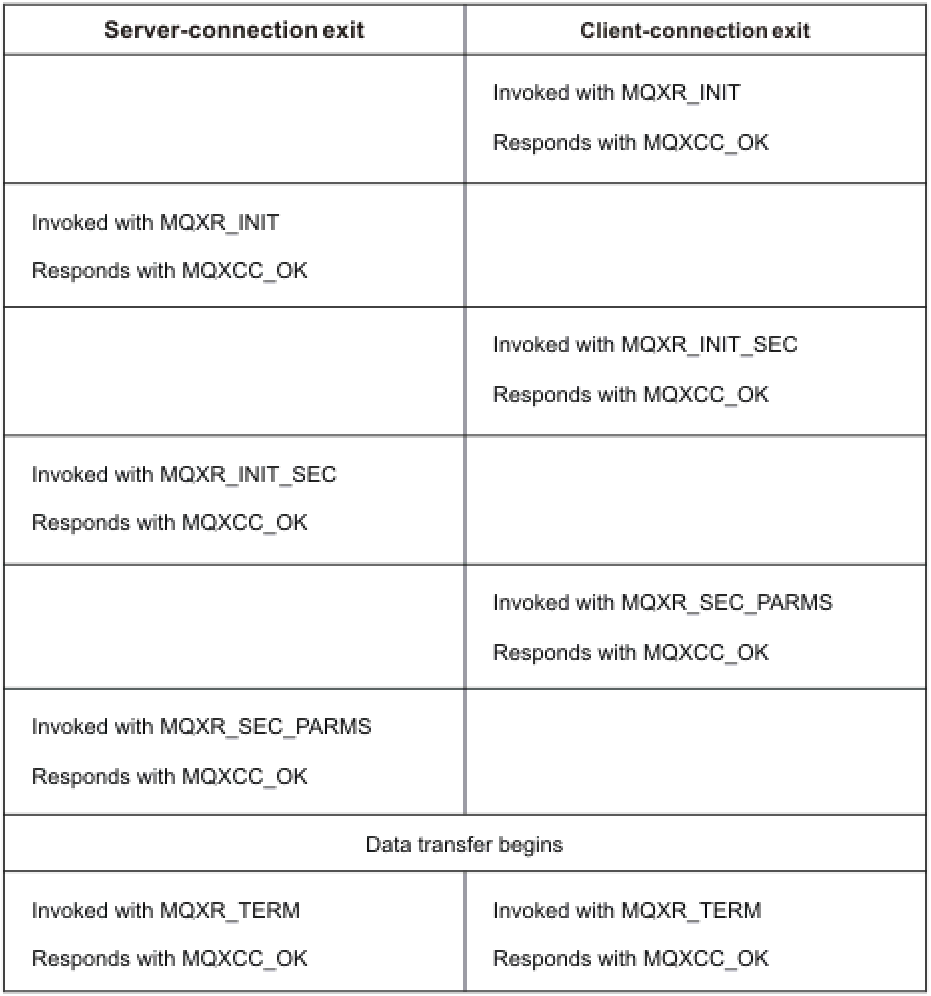
SecurityParmsPtr 및 SecurityParmsOffset이 MQCNO 구조에서 설정되지 않고 채널의 한 측 또는 양 측에 보안 엑시트가 있는 경우 이 보안 엑시트가 호출됩니다. 이 중 한 보안 엑시트는 SecurityParmsPtr을 통해 처리되는 자체 MQCSP 구조를 리턴할 수 있으며 이 보안 엑시트가 종료될 때까지(MQXR_TERM의 ExitReason) 보안 엑시트가 다시 호출되지 않습니다. 이 엑시트 기록기는 해당 단계에서 MQCSP에 사용되는 메모리를 비울 수 있습니다.

서버 연결 채널 인스턴스가 둘 이상의 대화를 공유하는 경우에는 보안 엑시트에 대한 호출 패턴이 두 번째 및 후속 대화에서 제한됩니다.

첫 번째 대화에서 이 패턴은 채널 인스턴스가 대화를 공유하지 않는 경우와 같습니다. 두 번째 및 후속 대화에서 보안 엑시트는 MQXR_INIT, MQXR_INIT_SEC 또는 MQXR_SEC_MSG로 호출되지 않습니다. MQXR_SEC_PARMS로 호출됩니다.

공유 대화가 있는 채널 인스턴스에서 MQXR_TERM은 실행 중인 마지막 대화에 대해서만 호출됩니다.

각 대화에는 MQCD를 대체하기 위한 엑시트의 MQXR_SEC_PARMS 호출 기회가 있으며 채널의 서버 연결 측에서는 이 기능이 큐 관리자에 연결되기 전에 값(예: MCAUserIdentifier 또는 LongMCAUserIdPtr 값)을 변경하는 데 유용할 수 있습니다.



Server-connection exit	Client-connection exit
	Invoked with MQXR_INIT Responds with MQXCC_OK
Invoked with MQXR_INIT Responds with MQXCC_OK	
	Invoked with MQXR_INIT_SEC Responds with MQXCC_OK
Invoked with MQXR_INIT_SEC Responds with MQXCC_OK	
	Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK
Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK	
Data transfer begins	
Invoked with MQXR_TERM Responds with MQXCC_OK	Invoked with MQXR_TERM Responds with MQXCC_OK

그림 *4.* 보안 매개변수를 사용한 클라이언트 연결 동의에 따른 클라이언트 연결 시작 교환

참고: IBM WebSphere MQ 7.1 릴리스 이전에 구성된 보안 엑시트 애플리케이션은 업데이트가 필요할 수 있습니다. 자세한 정보는 채널 보안 엑시트 프로그램을 참조하십시오.

ULW 송신 또는 수신 엑시트 프로그램에서 API 호출 식별

클라이언트에 MQI 채널을 사용하는 경우 에이전트 버퍼의 바이트 10은 송신 또는 수신 엑시트가 호출될 때 사용 중인 API 호출을 식별합니다. 이는 사용자 데이터를 포함하고 암호화 또는 디지털 서명과 같은 처리를 요구할 수 있는 채널 플로우를 식별하는 데 유용합니다.

다음 표는 API 호출을 처리 중일 때 채널 플로우의 바이트 10에 표시되는 데이터를 보여줍니다.

참고: 이러한 데이터가 이 바이트의 모든 값은 아닙니다. 다른 **예약** 값도 있습니다.

표 *7. API* 호출 식별

API 호출	**요청에 대한 바이트 10의 값**	**응답에 대한 바이트 10의 값**
MQCONN 47 페이지의 『1』, 47 페이지의 『2』	X'81'	X'91'
MQDISC 47 페이지의 『1』	X'82'	X'92'
MQOPEN 47 페이지의 『3』	X'83'	X'93'
MQCLOSE	X'84'	X'94'
MQGET 47 페이지의 『4』	X'85'	X'95'

표 7. *API 호출 식별 (계속)*

API 호출	요청에 대한 바이트 10의 값	응답에 대한 바이트 10의 값
MQPUT 47 페이지의 『4』	X'86'	X'96'
MQPUT1 요청 47 페이지의 『4』	X'87'	X'97'
MQSET 요청	X'88'	X'98'
MQINQ 요청	X'89'	X'99'
MQCMIT 요청	X'8A'	X'9A'
MQBACK 요청	X'8B'	X'9B'
MQSTAT 요청	X'8D'	X'9D'
MQSUB 요청	X'8E'	X'9E'
MQSUBRQ 요청	X'8F'	X'9F'
xa_start 요청	X'A1'	X'B1'
xa_end 요청	X'A2'	X'B2'
xa_open 요청	X'A3'	X'B3'
xa_close 요청	X'A4'	X'B4'
xa_prepare 요청	X'A5'	X'B5'
xa_commit 요청	X'A6'	X'B6'
xa_rollback 요청	X'A7'	X'B7'
xa_forget 요청	X'A8'	X'B8'
xa_recover 요청	X'A9'	X'B9'
xa_complete 요청	X'AA'	X'BA'

참고사항:

1. 클라이언트와 서버 간 연결이 MQCONN을 사용하는 클라이언트 애플리케이션에 의해 시작됩니다. 따라서 이 명령의 경우에는 특히 여러 개의 다른 네트워크 플로우가 있습니다. 이는 네트워크 연결을 종료하는 MQDISC에도 동일하게 적용됩니다.
2. MQCONNX는 클라이언트 서버 연결을 목적으로 MQCONN과 같은 방식으로 처리됩니다.
3. 대형 분배 목록이 열리는 경우 모든 필수 데이터를 SVRCONN MCA에 전달하기 위해 MQOPEN 호출당 두 개 이상의 네트워크 플로우가 있을 수도 있습니다.
4. 대형 메시지가 전송 세그먼트 크기를 초과할 수 있습니다. 이러한 상태가 발생하면 단일 호출 API 호출에서 다수의 네트워크 플로우가 발생할 수 있습니다.

z/OS 클라이언트를 큐 공유 그룹에 연결

큐 공유 그룹의 멤버인 서버에서 클라이언트와 큐 관리자 간 MQI 채널을 작성하여 클라이언트를 큐 공유 그룹에 연결할 수 있습니다.

이 태스크 정보

큐 공유 그룹은 같은 공유 큐 세트에 액세스할 수 있는 큐 관리자 세트로 구성됩니다. 공유 큐에 대한 자세한 정보는 공유 큐 및 큐 공유 그룹을 참조하십시오.

공유 큐에 있는 클라이언트는 큐 공유 그룹의 모든 멤버에 연결할 수 있습니다. 큐 공유 그룹 연결의 이점은 프론트 엔드 및 백엔드 가용성을 높이고 용량을 증가시킨다는 것입니다. 특정 큐 관리자 또는 일반 인터페이스에 연결할 수 있습니다.

큐 공유 그룹에 있는 큐 관리자에 직접 연결하면 메시지를 공유 대상 큐에 넣을 수 있는 이점으로 인해 백엔드 가용성을 증가시킬 수 있습니다.

큐 공유 그룹의 일반 인터페이스에 연결하면 그룹에 있는 큐 관리자 중 하나로 세션을 엽니다. 이는 클라이언트 큐 관리자가 그룹 내 모든 큐 관리자와 연결될 수 있기 때문에 프론트 엔드 가용성을 증가시킵니다. 큐 공유 그룹 내 특정 큐 관리자에게 연결하지 않으려는 경우 일반 인터페이스를 사용하여 그룹에 연결합니다.

일반 인터페이스는 Sysplex Distributor VIPA 주소 또는 VTAM 일반 자원 이름 또는 큐 공유 그룹에 대한 다른 공통 인터페이스일 수 있습니다. 일반 인터페이스 설정에 대한 자세한 정보는 큐 공유 그룹을 사용하여 IBM MQ for z/OS에 대한 통신 설정을 참조하십시오.

프로시저

큐 공유 그룹의 일반 인터페이스에 연결하려면 그룹 내 임의의 큐 관리자가 액세스할 수 있는 채널 정의를 작성해야 합니다. 이를 수행하려면 그룹 내 각 큐 관리자에 대해 동일한 정의를 가져야 합니다.

1. 다음 예에 표시된 것과 같이 SVRCONN 채널을 정의하십시오.

   ```
   DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
   QSGDISP(GROUP)
   ```

   서버에서의 채널 정의는 공유 Db2® 저장소에 저장됩니다. 큐 공유 그룹 내 각 큐 관리자는 정의의 로컬 사본을 작성하여 MQCONN이나 MQCONNX 호출을 발행할 때 항상 올바른 서버 연결 채널에 연결하도록 합니다.
2. 다음 예에 표시된 것과 같이 CLNTCONN 채널을 정의하십시오.

   ```
   DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
   CONNAME( VIPA address ) QMNAME(QSG1) +
   DESCR('Client-connection to Queue Sharing Group QSG1') QSGDISP(GROUP)
   ```

결과

큐 공유 그룹의 일반 인터페이스는 클라이언트 연결 채널의 CONNAME 필드에 저장되기 때문에 이제 그룹 내 어느 큐 관리자에도 연결할 수 있으며 해당 그룹이 소유하는 공유 큐에 넣을 수 있습니다.

구성 파일을 사용하여 클라이언트 구성

텍스트 파일의 속성을 사용하여 클라이언트를 구성할 수 있습니다. 이러한 속성은 환경 변수 또는 다른 플랫폼별 방식으로 대체될 수 있습니다.

이 태스크 정보

UNIX and Linux 플랫폼에서 사용되는 큐 관리자 구성 파일 `qm.ini`와 유사한 텍스트 파일을 사용하여 IBM MQ MQI clients을(를) 구성합니다. 이 파일에는 각각 **`attribute-name`** = *value* 형식의 행을 다수 포함하는 다수의 스탠자가 포함됩니다.

일반적으로 IBM MQ MQI client 구성 파일의 이름은 `mqclient.ini,`이지만 다른 이름을 지정하도록 선택할 수 있습니다. 이 파일의 구성 정보는 모든 플랫폼과 다음을 사용하는 클라이언트에 적용됩니다.

- MQI
- IBM MQ classes for Java
- IBM MQ classes for JMS
- IBM MQ classes for .NET
- XMS

속IBM MQ MQI client 구성 파일의 속성은 대부분의 IBM MQ 클라이언트에 적용할 수 있지만, 관리 .NET 및 XMS .NET 클라이언트나 IBM MQ classes for Java 또는 IBM MQ classes for JMS를 사용하는 클라이언트에서 읽을 수 없는 몇 가지 속성이 있습니다. 자세한 정보는 50 페이지의 『각 속성을 읽을 수 있는 IBM MQ 클라이언트』의 내용을 참조하십시오.

구성 기능은 한 큐 관리자에 대한 개별 연결에 특정한다기 보다는 클라이언트 애플리케이션이 임의의 큐 관리자에 대해 작성하는 모든 연결에 적용됩니다. 개별 큐 관리자 연결에 관련된 속성은 MQCD 구조를 사용하거나 CCDT(Client Channel Definition Table)를 사용하는 등의 프로그래밍 방식으로 구성할 수 있습니다.

다음은 클라이언트 구성 파일의 예입니다.

```
#* Module Name: mqclient.ini                                         *#
#* Type       : IBM MQ MQI client configuration file                 *#
#  Function   : Define the configuration of a client                 *#
#*                                                                   *#
#*******************************************************************#
#* Notes      :                                                      *#
#* 1) This file defines the configuration of a client                *#
#*                                                                   *#
#*******************************************************************#

ClientExitPath:
   ExitsDefaultPath=/var/mqm/exits
   ExitsDefaultPath64=/var/mqm/exits64

TCP:
   Library1=DLLName1
   KeepAlive = Yes
   ClntSndBuffSize=32768
   ClntRcvBuffSize=32768
   Connect_Timeout=0

MessageBuffer:
   MaximumSize=-1
   Updatepercentage=-1
   PurgeTime=0

LU62:
   TPName
   Library1=DLLName1
   Library2=DLLName2

PreConnect:
    Module=myMod
    Function=myFunc
    Data=ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
    Sequence=1

CHANNELS:
DefRecon=YES
ServerConnectionParms=SALES.SVRCONN/TCP/hostname.x.com(1414)
```

클라이언트 구성 파일을 사용하여 채널 연결을 여러 개 설정할 수는 없습니다.

IBM WebSphere MQ 7.0 이전 릴리스에서 지원되는 환경 변수는 이후 릴리스에서도 계속 지원되며, 이러한 환경 변수는 클라이언트 구성 파일에 있는 해당 값과 일치하고 환경 변수는 클라이언트 구성 파일 값을 대체합니다.

IBM MQ classes for JMS를 사용하는 클라이언트 애플리케이션의 경우 다음과 같은 방법으로 클라이언트 구성 파일을 대체할 수도 있습니다.

- JMS 구성 파일에 특성 설정
- Java 시스템 특성 설정. 이 경우 JMS 구성 파일도 대체됩니다.

.NET 클라이언트의 경우 .NET 애플리케이션 구성 파일을 사용하여 클라이언트 구성 파일과 해당 환경 변수를 대체할 수도 있습니다.

Linux UNIX 구성 파일의 주석

세미콜론 ';' 및 해시 '#' 문자를 사용하여 구성 파일 내에 주석의 시작을 표시할 수 있습니다. 이는 전체 행을 주석으로 표시하거나 설정 값에 포함되지 않는 행의 끝에 주석을 표시할 수 있습니다.

값에 이러한 문자 중 하나가 필요한 경우, 백슬래시 문자 '\'를 사용하여 이 문자를 이스케이프 처리해야 합니다.

다음 예제에서는 구성 파일 내 주석 사용을 보여줍니다.

```
# Example of an SSL stanza with comments
SSL:
    ClientRevocationChecks=REQUIRED ; Example of an end of line comment
    SSLCryptoHardware=GSK_PKCS11=/driver\;label\;password\;SYMMETRIC_CIPHER_ON # Example of
escaped comment characters.
```

프로시저

- 다음 주제의 정보를 활용하면 클라이언트를 구성하는 데 도움이 됩니다.
  - 50 페이지의 『클라이언트 구성 파일의 위치』
  - 50 페이지의 『각 속성을 읽을 수 있는 IBM MQ 클라이언트』

클라이언트 구성 파일의 위치

IBM MQ MQI client 구성 파일은 여러 위치에 보유할 수 있습니다.

클라이언트 애플리케이션은 다음 검색 경로를 사용하여 IBM MQ MQI client 구성 파일을 찾습니다.

1. 환경 변수 MQCLNTCF에 의해 지정되는 위치.

   이 환경 변수의 형식은 전체 URL입니다. 이는 파일 이름이 반드시 `mqclient.ini`일 필요는 없음을 의미하고 네트워크 접속 파일 시스템에 파일 배치를 용이하게 합니다.

   다음에 유의하십시오.

   - C, .NET 및 XMS 클라이언트는 `file:` 프로토콜만 지원합니다. URL 문자열이 `protocol:`로 시작되지 않는 경우 `file:` 프로토콜로 가정합니다.
   - 환경 변수 읽기를 지원하지 않는 Java 1.4.2 JRE를 감안하여 MQCLNTCF 환경 변수를 MQCLNTCF Java 시스템 특성으로 대체할 수 있습니다.
2. 애플리케이션의 현재 작업 디렉토리에 있는 `mqclient.ini`라는 파일.
3. Windows, UNIX and Linux 시스템의 IBM MQ 데이터 디렉토리에 있는 `mqclient.ini` 파일입니다.

   다음에 유의하십시오.

   - IBM MQ 데이터 디렉토리가 특정 플랫폼(예: IBM i 및 z/OS)에 존재하지 않거나, 클라이언트가 다른 제품과 함께 제공된 경우입니다.
   - UNIX and Linux 시스템에서 디렉토리는 `/var/mqm`입니다.
   - Windows 플랫폼에서는 설치 중에 MQ_DATA_PATH 환경 변수를 구성하여 데이터 디렉토리를 지시합니다. 일반적으로 `C:\ProgramData\IBM\MQ`입니다.

     참고: 클라이언트만 설치하는 경우, 환경 변수는 MQ_FILE_PATH일 수 있다.
   - 읽기 환경 변수를 지원하지 않는 Java 1.4.2 JRE를 허용하려면 MQ_DATA_PATH Java 시스템 특성으로 MQ_DATA_PATH 환경 변수를 수동으로 대체할 수 있습니다.
4. 플랫폼에 해당하고 사용자가 액세스할 수 있는 표준 디렉토리의 `mqclient.ini`라는 파일:
   - 모든 Java 클라이언트에서 이는 `user.home` Java 시스템 특성의 값입니다.
   - UNIX and Linux 플랫폼에서 C 클라이언트의 경우 이는 HOME 환경 변수의 값입니다.
   - Windows에서 C 클라이언트의 경우 이는 HOMEDRIVE 및 HOMEPATH 환경 변수의 병합된 값입니다.

각 속성을 읽을 수 있는 IBM MQ 클라이언트

IBM MQ MQI client 구성 파일에 있는 대부분의 속성은 C 클라이언트 및 비관리 .NET 클라이언트에서 사용할 수 있습니다. 그러나 관리 .NET 및 XMS .NET 클라이언트나 IBM MQ classes for Java 또는 IBM MQ classes for JMS를 사용하는 클라이언트에서 읽을 수 없는 몇 가지 속성이 있습니다.

표 8. 클라이언트의 각 유형에 적용할 수 있는 속성						
mqclient.ini 스탠자 이름 및 속성	**설명**	**C 및 비관리 .NET**	**Java**	**JMS**	**관리 .NET**	**관리 XMS .NET**
CHANNELS 스탠자						
CCSID	사용할 코드화 문자 세트 번호입니다.	예	아니오	아니오	예	예
ChannelDefinitionDirectory	클라이언트 채널 정의 테이블이 들어 있는 파일에 대한 디렉토리 경로입니다.	예	아니오	아니오	예	예
ChannelDefinitionFile	클라이언트 채널 정의 테이블이 들어 있는 파일의 이름입니다.	예	아니오	아니오	예	예
ReconDelay	자동으로 다시 연결할 수 있는 클라이언트 프로그램에 대해 다시 연결 지연을 구성하는 관리 옵션을 제공합니다.	예	아니오	예	예	예
DefRecon	클라이언트 프로그램이 자동으로 다시 연결되도록 설정하거나, 자동으로 다시 연결되도록 작성된 클라이언트 프로그램을 자동으로 다시 연결할 수 없도록 설정하는 관리 옵션을 제공합니다.	예	아니오	예	예	예
MQReconnectTimeout	클라이언트에 다시 연결되기까지의 제한시간(초)입니다.	예	아니오	아니오	예	아니오
ServerConnectionParms	IBM MQ 서버의 위치와 사용할 통신 방법입니다.	예	아니오	아니오	예	예

표 8. 클라이언트의 각 유형에 적용할 수 있는 속성 *(계속)*

mqclient.ini 스탠자 이름 및 속성	설명	C 및 비관리 .NET	Java	JMS	관리 .NET	관리 XMS .NET
Put1DefaultAlwaysSync	MQPMO_RESPONSE_AS_Q_DEF 옵션을 사용하여 MQPUT1 함수 호출의 작동을 제어합니다.	예	예	예	예	예
PasswordProtection	SSL 또는 TLS를 사용하는 대신 보호된 비밀번호를 MQCSP 구조에 설정할 수 있습니다.	예	예	예	예	예
ClientExitPath 스탠자						
ExitsDefaultPath	클라이언트에 대한 32비트 채널 엑시트의 위치를 지정합니다.	예	예	예	예	예
ExitsDefaultPath64	클라이언트에 대해 64비트 채널 엑시트의 위치를 지정합니다.	예	예	예	예	예
JavaExitsClassPath	Java 엑시트가 실행될 때 클래스 경로에 추가될 값입니다.	아니오	예	예	아니오	아니오
JMQI 스탠자						
useMQCSPauthentication	큐 관리자로 인증할 때 IBM MQ classes for Java 및 IBM MQ classes for JMS 애플리케이션이 호환 모드여야 하는지 또는 MQCSP 인증 모드여야 하는지 여부를 제어합니다.	아니오	예	예	아니오	아니오
MessageBuffer 스탠자						

표 8. 클라이언트의 각 유형에 적용할 수 있는 속성 *(계속)*						
mqclient.ini 스탠자 이름 및 속성	**설명**	**C 및 비관리 .NET**	**Java**	**JMS**	**관리 .NET**	**관리 XMS .NET**
MaximumSize	미리 읽기 버퍼의 크기로, 값은 1 - 999 999KB 범위입니다.	예	예	예	예	예
PurgeTime	미리 읽기 버퍼에 남겨진 메시지가 영구 제거되기 전까지의 간격(초)입니다.	예	예	예	예	예
UpdatePercentage	클라이언트 애플리케이션이 서버에 대한 새 요청을 작성하는 시간을 판별하기 위한 임계값 계산에서 사용되는 1 - 100 범위의 업데이트 백분율 값입니다.	예	예	예	예	예
PreConnect 스탠자						
데이터	연결 정의가 저장되어 있는 저장소의 URL입니다.	예	아니오	아니오	아니오	아니오
함수	PreConnect 엑시트 코드가 포함된 라이브러리로 들어가는 기능 시작점의 이름입니다.	예	아니오	아니오	아니오	아니오
모듈	API 엑시트 코드가 포함된 모듈의 이름입니다.	예	아니오	아니오	아니오	아니오
순서	이 엑시트가 다른 엑시트에 상대적으로 호출되는 순서입니다.	예	아니오	아니오	아니오	아니오
보안 스탠자						

표 8. 클라이언트의 각 유형에 적용할 수 있는 속성 (계속)						
mqclient.ini 스탠자 이름 및 속성	**설명**	**C 및 비관리 .NET**	**Java**	**JMS**	**관리 .NET**	**관리 XMS .NET**
DisableClientAMS	큐 관리자에 대한 클라이언트 연결을 위해 AMS를 사용 안함 또는 사용으로 설정합니다.	예	예	예	아니오	아니오
SSL 스탠자						
CDPCheckExtensions	이 큐 관리자의 SSL 또는 TLS 채널이 CrlDistributionPoint 인증서 확장에 이름 지정된 CDP 서버를 검사하려고 시도할지 여부를 지정합니다.	예	아니오	아니오	아니오	아니오
CertificateLabel	채널 정의의 인증서 레이블입니다.	예	아니오	아니오	아니오	아니오
CertificateValPolicy	사용되는 인증서 유효성 검증의 유형을 판별합니다.	예	아니오	아니오	아니오	아니오
ClientRevocationChecks	클라이언트 연결 호출에 SSL/TLS 채널을 사용하는 경우 인증서 폐기 검사가 구성되는 방식을 판별합니다.	예	아니오	아니오	아니오	아니오
EncryptionPolicySuiteB	채널이 Suite-B 준수 암호화를 사용하는지 및 사용할 강도 레벨을 판별합니다.	예	아니오	아니오	아니오	아니오

표 8. 클라이언트의 각 유형에 적용할 수 있는 속성 (계속)						
mqclient.ini 스탠자 이름 및 속성	**설명**	**C 및 비관리 .NET**	**Java**	**JMS**	**관리 .NET**	**관리 XMS .NET**
OCSPAuthentication	OCSP가 사용 가능하고 OCSP 폐기 검사가 인증서 폐기 상태를 판별할 수 없는 경우에 IBM MQ의 작동을 정의합니다.	예	아니오	아니오	아니오	아니오
OCSPCheckExtensions	IBM MQ가 AuthorityInfoAccess 인증서 확장자에 대해 작동하는지 여부를 제어합니다.	예	아니오	아니오	아니오	아니오
SSLCryptoHardware	PKCS #11 암호화 하드웨어가 시스템에 표시되도록 구성하기 위해 필요한 매개변수 문자열을 설정합니다.	예	아니오	아니오	아니오	아니오
SSLFipsRequired	IBM MQ에서 암호화가 수행되는 경우 FIPS 인증 알고리즘만 사용할 것인지 여부를 지정합니다.	예	아니오	아니오	아니오	아니오
SSLHTTPProxyName	문자열은 OCSP 검사를 위해 GSKit에서 사용될 HTTP 프록시 서버의 호스트 이름 또는 네트워크 주소입니다.	예	아니오	아니오	아니오	아니오
SSLKeyRepository	사용자의 디지털 인증서를 보유하는 키 저장소의 위치이며 스템 형식입니다.	예	아니오	아니오	아니오	아니오

표 *8.* 클라이언트의 각 유형에 적용할 수 있는 속성 *(*계속*)*

mqclient.ini 스탠자 이름 및 속성	**설명**	**C 및 비관리 .NET**	**Java**	**JMS**	**관리 .NET**	**관리 XMS .NET**
SSLKeyResetCount	비밀 키를 재협상하기 전에 SSL 또는 TLS 채널에서 송신 및 수신된 암호화되지 않은 바이트의 수입니다.	예	아니오	아니오	아니오	아니오
TCP 스탠자						
ClntRcvBuffSize	클라이언트 연결 서버 연결 채널의 클라이언트 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다.	예	예	예	예	예
ClntSndBuffSize	클라이언트 연결 서버 연결 채널의 클라이언트 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다.	예	예	예	예	예
Connect_Timeout	소켓 연결 시도가 제한시간 초과하기 전의 시간(초)입니다.	예	예	예	아니오	아니오
IPAddressVersion	채널 연결에 사용할 IP 프로토콜을 지정합니다.	예	아니오	아니오	예	예
KeepAlive	KeepAlive 기능을 설정하거나 해제합니다.	예	예	예	예	예
Windows Library1	(Windows만 해당) TCP/IP 소켓 DLL의 이름입니다.	예	아니오	아니오	아니오	아니오

클라이언트 구성 파일의 CHANNELS 스탠자

CHANNELS 스탠자를 사용하여 클라이언트 채널에 대한 정보를 지정할 수 있습니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

다음 속성이 CHANNELS 스탠자에 포함될 수 있습니다.

CCSID = *number*

사용할 코드화 문자 세트 번호입니다.

이 속성은 C, 비관리 .NET, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

CCSID 번호는 MQCCSID 환경 매개변수와 같습니다.

ChannelDefinitionDirectory = *path*

클라이언트 채널 정의 테이블이 들어 있는 파일에 대한 디렉토리 경로입니다.

이 속성은 C, 비관리 .NET, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

Windows Windows 시스템에서 기본값은 IBM MQ 데이터 및 로그 파일 디렉토리(일반적으로 `C:\ProgramData\IBM\MQ`)입니다.

Linux UNIX UNIX and Linux 시스템에서 기본값은 `/var/mqm`입니다.

V 9.0.0 ChannelDefinitionDirectory는 ChannelDefinitionFile 속성과 함께 작동하는 URL을 포함할 수 있습니다(39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』 참조).

ChannelDefinitionDirectory 경로는 MQCHLLIB 환경 매개변수와 같습니다.

ChannelDefinitionFile = *filename*|AMQCLCHL.TAB

클라이언트 채널 정의 테이블이 들어 있는 파일의 이름입니다.

이 속성은 C, 비관리 .NET, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

클라이언트 채널 정의 테이블은 MQCHLTAB 환경 매개변수와 같습니다.

ReconDelay = (delay[,rand]) (delay[,rand])...

ReconDelay 속성은 클라이언트 프로그램에 대해 자동으로 다시 연결할 수 있는 다시 연결 지연을 구성하는 관리 옵션을 제공합니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

다음은 구성 예입니다.

```
ReconDelay=(1000,200)(2000,200)(4000,1000)
```

표시된 예는 1초에 최대 200밀리초의 임의 간격을 더한 초기 지연을 정의합니다. 다음 지연은 2초에 최대 200밀리초의 임의 간격을 더합니다. 모든 후속 지연은 4초에 최대 1000밀리초의 임의 간격을 더합니다.

DefRecon = NO|YES|QMGR |DISABLED

DefRecon 속성은 클라이언트 프로그램이 자동으로 다시 연결할 수 있도록 하거나, 자동으로 다시 연결하도록 작성된 클라이언트 프로그램의 자동으로 다시 연결을 사용할 수 없도록 하는 관리 옵션을 제공합니다. 프로그램이 다시 연결과 호환되지 않는 옵션(예: `MQPMO_LOGICAL_ORDER`)을 사용하는 경우 위의 옵션 중 후자를 설정하는 것으로 선택할 수 있습니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

Java용 IBM MQ 클래스에서는 자동 클라이언트 다시 연결이 지원되지 않습니다.

DefRecon 옵션의 해석은 `MQCNO_RECONNECT_*` 값이 클라이언트 프로그램에도 설정되어 있는지 여부와 설정된 값에 따라 다릅니다.

클라이언트 프로그램이 `MQCONN`을 사용하여 연결하거나 `MQCONNX`를 사용하여 `MQCNO_RECONNECT_AS_DEF` 옵션을 설정하는 경우 DefRecon에 의해 설정된 다시 연결 값이 적용됩니다.

프로그램에서 또는 DefRecon 옵션에 의해 다시 연결 값이 설정되지 않은 경우 이 클라이언트 프로그램은 자동으로 다시 연결되지 않습니다.

NO
: **MQCONNX**로 대체되지 않는 한, 클라이언트는 자동으로 다시 연결되지 않습니다.

YES
: **MQCONNX**로 대체되지 않으면 클라이언트는 자동으로 다시 연결됩니다.

QMGR
: **MQCONNX**로 대체되지 않으면 클라이언트는 자동으로 다시 연결되지만 동일한 큐 관리자에만 다시 연결됩니다. QMGR 옵션은 MQCNO_RECONNECT_Q_MGR와 동일한 효과를 갖습니다.

DISABLED
: **MQCONNX** MQI 호출을 사용하여 클라이언트 프로그램이 요청한 경우에도 다시 연결을 사용할 수 없습니다.

자동 클라이언트 다시 연결은 다음 두 가지 값에 따라 달라집니다.

- 애플리케이션에서 설정된 다시 연결 옵션
- mqclient.ini 파일의 DefRecon 값

표 9. 자동으로 다시 연결은 애플리케이션 및 *mqclient.ini* 파일에서 설정된 값에 따라 달라집니다.

mqclient.ini의 DefRecon 값	**애플리케이션에 설정된 재연결 옵션**			
	MQCNO_RECONNECT	MQCNO_RECONNECT_Q_MGR	MQCNO_RECONNECT_AS_DEF	MQCNO_RECONNECT_DISABLED
NO	YES	큐 관리자	아니오	아니오
YES	YES	큐 관리자	YES	아니오
QMGR	YES	큐 관리자	큐 관리자	아니오
DISABLED	아니오	아니오	아니오	아니오

MQReconnectTimeout
: 클라이언트에 다시 연결되기까지의 제한시간(초)입니다. 기본 값은 1800초(30분)입니다.

이 속성은 C 및 비관리 .NET 클라이언트 및 관리 .NET 클라이언트에서 읽을 수 있습니다.

IBM MQ classes for JMS 클라이언트에서는 연결 팩토리 특성 CLIENTRECONNECTTIMEOUT을 사용하여 다시 연결 제한시간을 지정할 수 있습니다. 이 특성의 기본값은 1800초(30분)입니다.

IBM MQ classes for XMS .NET 클라이언트는 다음 특성을 사용하여 다시 연결 제한시간을 지정할 수 있습니다.

- 연결 팩토리 특성 CLIENTRECONNECTTIMEOUT. 이 특성의 기본값은 1800초(30분)입니다. 이 특성은 관리 모드에 대해서만 유효합니다.
- 특성 XMSC.WMQ_CLIENT_RECONNECT_TIMEOUT. 이 특성의 기본값은 1800초(30분)입니다. 이 특성은 관리 모드에 대해서만 유효합니다.

ServerConnectionParms
: ServerConnectionParms는 MQSERVER 환경 매개변수와 같고 사용할 IBM MQ 서버의 위치와 통신 메소드를 지정합니다.

이 속성은 C, 비관리 .NET, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

ServerConnectionParms 속성은 단순 채널만을 정의합니다. 이 속성을 사용하여 TLS 채널 또는 채널 엑시트가 있는 채널을 정의할 수 없습니다. 이는 *ChannelName* / *TransportType* / *ConnectionName* 형식의 문자열이고 *ConnectionName*은 완전한 네트워크 이름이어야 합니다. 슬래시 문자는 채널 이름, 전송 유형 및 연결 이름을 구분하는 데 사용되기 때문에 *ChannelName*은 슬래시(/) 문자를 포함할 수 없습니다.

ServerConnectionParms가 클라이언트 채널을 정의하는 데 사용될 때 최대 메시지 길이 100MB가 사용됩니다. 따라서 채널에 적용되는 최대 메시지 크기는 서버에서 SVRCONN 채널에서 지정되는 값입니다.

단일 클라이언트 채널 연결만 수행할 수 있음을 참고하십시오. 예를 들어, 다음 두 가지 입력 항목이 있는 경우:

```
ServerConnectionParms=R1.SVRCONN/TCP/localhost(1963)
ServerConnectionParms=R2.SVRCONN/TCP/localhost(1863)
```

두 번째 입력 항목만 사용됩니다.

`ConnectionName`을 명시된 전송 유형에 대한 쉼표로 구분된 이름 목록으로 지정하십시오. 일반적으로 한 개의 이름만 필요합니다. 동일한 특성과의 다중 연결을 구성하기 위해 다중 *hostnames*를 제공할 수 있습니다. 이러한 연결은 연결이 성공적으로 설정될 때까지 연결 목록에 지정되는 순서대로 시도됩니다. 연결에 실패하면 클라이언트가 다시 처리를 시작합니다. 연결 목록은 큐 관리자 그룹에 대해 다시 연결 가능한 클라이언트에 대한 연결을 구성하기 위한 대안입니다.

Put1DefaultAlwaysSync = NO| YES

`MQPMO_RESPONSE_AS_Q_DEF` 옵션을 사용하여 MQPUT1 함수 호출의 작동을 제어합니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java 및 IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

아니오

MQPUT1을 `MQPMO_SYNCPOINT`를 사용하여 설정하면 `MQPMO_ASYNC_RESPONSE`처럼 작동합니다. 마찬가지로 MQPUT1을 `MQPMO_NO_SYNCPOINT`를 사용하여 설정하면 `MQPMO_SYNC_RESPONSE`처럼 작동합니다. 이 값은 기본값입니다.

YES

MQPUT1은 `MQPMO_SYNCPOINT` 또는 `MQPMO_NO_SYNCPOINT`가 설정되는지 여부에 관계없이 `MQPMO_SYNC_RESPONSE`가 설정되는 것처럼 작동합니다.

PasswordProtection = `Compatible|always|optional`

IBM MQ 8.0에서 TLS를 사용하는 대신 MQCSP 구조에서 보호된 비밀번호를 설정할 수 있도록 합니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java 및 IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

MQCSP 비밀번호 보호 사용은 TLS 암호화 설정에 비해 단순하기 때문에 MQCSP 비밀번호 보호는 테스트 및 개발 용도로 유용하지만 TLS 암호화에 비해 안전하지 않습니다.

자세한 정보는 MQCSP 비밀번호 보호를 참조하십시오.

관련 정보

큐 관리자에 IBM MQ MQI 애플리케이션 연결

클라이언트 구성 파일의 ClientExitPath 스탠자

ClientExitPath 스탠자를 사용하여 클라이언트에서 채널 엑시트의 기본 위치를 지정할 수 있습니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

다음 속성이 ClientExitPath 스탠자에 포함될 수 있습니다.

ExitsDefaultPath = *string*

클라이언트에 대한 32비트 채널 종료의 위치를 지정합니다.

이 속성은 C, 비관리 .NET, 관리 .NET, 관리 XMS .NET, IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트에서 읽을 수 있습니다. IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트는 Java로 작성되지 않는 32비트 채널 엑시트를 찾으려면 이 속성을 사용합니다.

ExitsDefaultPath64 = *string*

클라이언트에 대한 64비트 채널 종료의 위치를 지정합니다.

이 속성은 C, 비관리 .NET, 관리 .NET, 관리 XMS .NET, IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트에서 읽을 수 있습니다. IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트는 Java로 작성되지 않는 64비트 채널 엑시트를 찾으려면 이 속성을 사용합니다.

JavaExitsClassPath = *string*
Java 엑시트가 실행될 때 클래스 경로에 추가될 값입니다. 이 값은 다른 언어로 된 엑시트에서는 무시됩니다.

이 속성은 IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트에서 읽을 수 있습니다.

JMS 구성 파일에서 JavaExitsClassPath 이름에는 표준 com.ibm.mq.cfg. 접두부가 지정되고, 이 전체 이름은 IBM WebSphere MQ 7.0 이상 시스템 특성에도 사용됩니다. IBM WebSphere MQ 6.0에서 이 속성은 IBM WebSphere MQ 6.0 readme 파일에 문서화된 시스템 특성 com.ibm.mq.exitClasspath를 사용하여 지정되었습니다. com.ibm.mq.exitClasspath는 더 이상 사용되지 않습니다. JavaExitsClassPath와 exitClasspath가 모두 있으면 JavaExitsClassPath를 사용합니다. exitClasspath 사용만 있는 경우에는 IBM WebSphere MQ 7.0 이상에서도 계속 사용됩니다.

클라이언트 구성 파일의 JMQI 스탠자

JMQI 스탠자를 사용하여 IBM MQ classes for Java 및 IBM MQ classes for JMS에서 사용하는 Java Message Queuing Interface(JMQI)에 대한 구성 매개변수를 지정합니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

다음 속성이 JMQI 스탠자에 포함될 수 있습니다.

useMQCSPauthentication = NO | YES
큐 관리자로 인증할 때 IBM MQ classes for Java 및 IBM MQ classes for JMS 애플리케이션이 호환 모드여야 하는지 또는 MQCSP 인증 모드여야 하는지 여부를 제어합니다.

이 속성은 IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트에서 읽을 수 있습니다.

이 속성에는 다음 값이 사용될 수 있습니다.

아니오
큐 관리자의 인증을 받을 때 호환 모드를 사용합니다. 이 값은 기본값입니다.

YES
큐 관리자의 인증을 받을 때 MQCSP 인증 모드를 사용합니다.

호환 모드와 MQCSP 인증 모드에 대한 자세한 정보는 Java 클라이언트와의 연결 인증을 참조하십시오.

Windows 클라이언트 구성 파일의 LU62, NETBIOS 및 SPX 스탠자

Windows 시스템에서만 이러한 스탠자를 사용하여 지정된 네트워크 프로토콜에 대해 구성 매개변수를 지정할 수 있습니다.

LU62 스탠자

LU62 스탠자를 사용하여 SNA LU 6.2 프로토콜 구성 매개변수를 지정할 수 있습니다. 다음 속성이 이 스탠자에 포함될 수 있습니다.

Library1 = *DLLName*|WCPIC32
APPC DLL의 이름입니다.

Library2 = *DLLName*|WCPIC32
Library1과 동일하며 코드가 두 개의 별도 라이브러리에 저장된 경우에 사용됩니다.

TPName
원격 사이트에서 시작할 TP 이름입니다.

NETBIOS 스탠자

NETBIOS 스탠자를 사용하여 NetBIOS 프로토콜 구성 매개변수를 지정하십시오. 다음 속성이 이 스탠자에 포함될 수 있습니다.

AdapterNum = *number*|0
LAN 어댑터의 번호입니다.

Library1 = *DLLName*|NETAPI32
NetBIOS DLL의 이름입니다.

LocalName = *name*
LAN에서 이 컴퓨터가 표시되는 이름입니다.

이는 MQNAME 환경 매개변수와 같습니다.

NumCmds = *number*|1
할당할 명령 수입니다.

NumSess = *number*|1
할당할 세션 수입니다.

SPX 스탠자

SPX 스탠자를 사용하여 SPX 프로토콜 구성 매개변수를 지정하십시오. 다음 속성이 이 스탠자에 포함될 수 있습니다.

BoardNum = *number*|0
LAN 어댑터 번호입니다.

KeepAlive = YES|NO
KeepAlive 기능을 켜거나 끕니다.

KeepAlive = YES로 설정하면 SPX가 연결의 다른 측이 여전히 사용 가능한지 주기적으로 검사합니다. 그렇지 않은 경우 채널이 닫힙니다.

Library1 = *DLLName*|WSOCK32.DLL
SPX DLL의 이름입니다.

Library2 = *DLLName*|WSOCK32.DLL
Library1과 동일하며 코드가 두 개의 별도 라이브러리에 저장된 경우에 사용됩니다.

Socket = *number*|5E86
16진 표기법의 SPX 소켓 번호입니다.

클라이언트 구성 파일의 MessageBuffer 스탠자

MessageBuffer 스탠자를 사용하여 메시지 버퍼에 대한 정보를 지정할 수 있습니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

다음 속성이 MessageBuffer 스탠자에 포함될 수 있습니다.

MaximumSize = *integer*|1
미리 읽기 버퍼의 크기로, 값은 1 - 999 999KB 범위입니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

다음 특수 값이 존재합니다.

-1
클라이언트가 적절한 값을 판별합니다.

0
클라이언트에 대해 미리 읽기 가 사용 불가능합니다.

PurgeTime = *integer*|600
미리 읽기 버퍼에 남겨진 메시지가 영구 제거되기 전까지의 간격(초)입니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

클라이언트 애플리케이션이 MsgId 또는 CorrelId에 따라 메시지를 선택하는 경우 미리 읽기 버퍼가 이전에 요청된 MsgId 또는 CorrelId를 사용하여 클라이언트에게 송신된 메시지를 포함할 수 있습니다. 이러한 메시지는 적절한 MsgId 또는 CorrelId로 MQGET이 발행될 때까지 미리 읽기 버퍼에 남아 있습니다. PurgeTime을 설정해서 미리 읽기 버퍼로부터 메시지를 제거할 수 있습니다. 영구 제거 간격보다 긴 시간 동안 미리 읽기 버퍼에 남아 있는 모든 메시지는 자동으로 영구 제거됩니다. 이러한 메시지는 큐 관리자의 큐에서 이미 제거되었으므로 찾아보지 않으면 손실됩니다.

유효한 범위는 1 - 999 999초이거나 영구 제거가 발생하지 않음을 의미하는 특수 값 0입니다.

UpdatePercentage = *integer*|-1

클라이언트 애플리케이션이 서버에 대한 새 요청을 작성하는 시간을 판별하기 위한 임계값 계산에서 사용되는 1 - 100 범위의 업데이트 백분율 값입니다. 특수 값 -1은 클라이언트가 적절한 값을 판별함을 나타냅니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

클라이언트는 클라이언트 애플리케이션이 이용한 데이터 양을 표시하는 요청을 서버에 주기적으로 송신합니다. MQGET 호출을 통해 클라이언트가 검색한 바이트 수 *n*이 임계값 *T*를 초과하면 요청이 전송됩니다. 서버로 새 요청이 전송될 때마다 *n* 이 0으로 재설정됩니다.

임계값 T는 다음과 같이 계산됩니다.

```
T = Upper - Lower
```

상한값은 미리 읽기 버퍼 크기와 같으며 *MaximumSize* 속성에 의해 KB 단위로 지정됩니다. 그 기본값은 100KB입니다.

하한값은 상한값보다 낮으며 *UpdatePercentage* 속성에 의해 지정됩니다. 이 속성은 1 - 100 범위의 숫자이고 기본값은 20입니다. 하한값은 다음과 같이 계산됩니다.

```
Lower = Upper x UpdatePercentage / 100
```

예 1:

MaximumSize 및 UpdatePercentage 속성은 해당 속성의 기본값 100KB 및 20KB를 사용합니다.

클라이언트는 메시지를 검색하기 위해 MQGET을 호출하고 이를 반복적으로 수행합니다. 이는 MQGET이 n바이트를 이용할 때까지 계속됩니다.

다음 계산 사용

```
T = Upper - Lower
```

T는 (100 - 20) = 80KB입니다.

따라서 MQGET 호출이 큐에서 80KB를 제거하면 클라이언트가 자동으로 새 요청을 작성합니다.

예제 2:

MaximumSize 속성은 그 기본값 100KB를 사용하고 UpdatePercentage에 대해서는 값 40이 선택됩니다.

클라이언트는 메시지를 검색하기 위해 MQGET을 호출하고 이를 반복적으로 수행합니다. 이는 MQGET이 n바이트를 이용할 때까지 계속됩니다.

다음 계산 사용

```
T = Upper - Lower
```

T는 (100 - 40) = 60KB입니다.

따라서 MQGET 호출이 큐에서 60KB를 제거하면 클라이언트가 자동으로 새 요청을 작성합니다. 이는 기본값 사용 시 예 1보다 빠릅니다.

따라서 임계값 *T*로 큰 값을 선택할수록 클라이언트에서 서버로 요청이 송신되는 빈도가 감소하게 됩니다. 반대로 임계값 *T*로 작은 값을 선택할수록 클라이언트에서 서버로 송신되는 요청의 빈도가 증가하게 됩니다.

그러나 임계값 *T*를 선택하는 것은 미리 읽기 버퍼가 비어 있는 경우가 증가할 수 있어서 미리 읽기의 성능 이점이 줄어드는 것을 의미할 수 있습니다. 이러한 상황이 발생하면 데이터가 서버에서 도착할 때까지 대기하면서 MQGET 호출을 일시정지해야 할 수 있습니다.

클라이언트 구성 파일의 PreConnect 스탠자

PreConnect 스탠자를 사용하여 `mqclient.ini` 파일에 PreConnect 엑시트를 구성합니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

PreConnect 스탠자에는 다음 속성이 포함될 수 있습니다.

Data = *user_data*

이 속성은 Preconnect 엑시트로 전달되는 사용자 데이터를 지정합니다. Preconnect 엑시트로 전달되는 데이터는 사용 중인 Preconnect 엑시트의 구현 및 전달할 데이터에 따라 다릅니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

예를 들어 이 속성은 LDAP 서버를 사용할 때와 같이 연결 정의를 저장하는 저장소의 URL을 지정하는 데 사용될 수 있습니다.

```
Data = ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
```

Function = *myFunc*

PreConnect 엑시트 코드가 포함된 라이브러리로 들어가는 기능 시작점의 이름입니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

기능 정의는 PreConnect 엑시트 프로토타입 MQ_PRECONNECT_EXIT를 준수합니다.

이 필드의 최대 길이는 MQ_EXIT_NAME_LENGTH입니다.

Module = *myMod*

API 엑시트 코드가 포함된 모듈의 이름입니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

이 필드에 모듈의 전체 경로 이름이 포함되어 있는 경우에는 있는 그대로 사용됩니다.

Sequence = *sequence_number*

이 엑시트가 다른 엑시트에 상대적으로 호출되는 순서입니다. 낮은 순서 번호를 가진 엑시트가 더 높은 순서 번호를 가진 엑시트보다 먼저 호출됩니다. 엑시트의 순서 번호를 연속으로 지정할 필요는 없습니다. 1, 2, 3의 순서는 7, 42, 1096의 순서와 동일한 결과를 제공합니다. 이 속성은 사인되지 않은 숫자 값입니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

`mqclient.ini` 파일 내에 PreConnect 스탠자를 여러 개 정의할 수 있습니다. 각 엑시트의 처리 순서는 스탠자의 순서 속성에 따라 결정됩니다.

관련 정보

저장소의 사전 연결 엑시트를 사용하여 연결 정의 참조

클라이언트 구성 파일의 보안 스탠자

보안 스탠자를 사용하여 큐 관리자에 대한 클라이언트 연결을 위해 AMS를 사용 안함 또는 사용으로 설정하십시오.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

보안 스탠자에 다음 속성을 포함할 수 있습니다.

DisableClientAMS = NO|YES

DisableClientAMS 속성을 사용하면 IBM WebSphere MQ 7.5 이상의 클라이언트를 사용하여 제품의 이전 버전에서 큐 관리자에 연결하고 2085(MQRC_UNKNOWN_OBJECT_NAME) 오류가 보고되면 IBM MQ Advanced Message Security(AMS)를 사용 안함으로 설정해야 합니다.

IBM WebSphere MQ 7.5부터, IBM MQ Advanced Message Security(AMS)는 IBM MQ 클라이언트에서 자동으로 사용 가능하므로 기본적으로 클라이언트는 보안 정책에서 큐 관리자의 오브젝트를 검사하려고 시도합니다. 그러나 IBM WebSphere MQ 7.1과 같이 제품의 이전 버전의 서버는 AMS를 사용으로 설정하지 않았으며 이로 인해 2085(MQRC_UNKNOWN_OBJECT_NAME) 오류가 보고됩니다.

다음 예는 DisableClientAMS 속성을 사용하는 방법을 보여줍니다.

- AMS을 사용하지 않으려면 다음을 수행하십시오.

```
Security:
DisableClientAMS=Yes
```

- AMS을 사용 가능하게 하려면 다음을 수행하십시오.

```
Security:
DisableClientAMS=No
```

이 속성은 C, IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트에서 읽을 수 있습니다.

관련 정보

클라이언트에서 Advanced Message Security 사용 안함

클라이언트 구성 파일의 SSL 스탠자

SSL 스탠자를 사용하여 TLS의 사용에 대한 정보를 지정할 수 있습니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

다음 속성이 SSL 스탠자에 포함될 수 있습니다.

CDPCheckExtensions=YES|NO

CDPCheckExtensions는 이 큐 관리자의 TLS 채널이 CrlDistributionPoint 인증서 확장자에서 이름 지정되는 CDP 서버를 검사하려고 시도할지 여부를 지정합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

이 속성에는 다음 값이 사용될 수 있습니다.

- YES: TLS 채널이 CDP 서버를 검사하여 디지털 인증서가 폐기되었는지 여부를 판별하려고 시도합니다.
- NO: TLS 채널이 CDP 서버를 검사하려고 시도하지 않습니다. 이 값은 기본값입니다.

CertificateLabel = *string*

채널 정의의 인증서 레이블입니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

자세한 정보는 인증서 레이블(CERTLABL)을 참조하십시오.

CertificateValPolicy = *string*

사용되는 인증서 유효성 검증의 유형을 판별합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

이 속성에는 다음 값이 사용될 수 있습니다.

ANY

기본 보안 소켓 라이브러리가 지원하는 모든 인증서 유효성 검증 정책을 사용합니다. 이 설정이 기본 설정입니다.

RFC5280
RFC 5280 표준을 준수하는 인증서 유효성 검증만 사용합니다.

ClientRevocationChecks = REQUIRED|OPTIONAL|DISABLED
클라이언트 연결 호출에 TLS 채널을 사용하는 경우 인증서 폐기 검사가 구성되는 방식을 판별합니다. **OCSPAuthentication**도 참조하십시오.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

이 속성에는 다음 값이 사용될 수 있습니다.

REQUIRED(기본값)
CCDT에서 인증서 폐기 구성을 로드하고 구성된 대로 폐기 검사를 수행하려고 시도합니다. CCDT 파일을 열 수 없거나 인증서를 유효성 검증할 수 없는 경우(예를 들어, OCSP 또는 CRL 서버가 사용 불가능함으로 인해) MQCONN 호출이 실패합니다. CCDT에 폐기 구성이 없는 경우 폐기 검사가 수행되지 않지만 이로 인해 채널이 실패하지는 않습니다.

Windows Windows 시스템에서 CRL 폐기 검사를 위해 Active Directory를 사용할 수도 있습니다. OCSP 폐기 검사에는 Active Directory를 사용할 수 없습니다.

선택사항
REQUIRED의 경우 인증서 폐기 구성을 로드할 수 없는 경우에도 채널이 실패하지 않습니다.

DISABLED
CCDT에서 인증서 폐기 구성을 로드하기 위한 시도를 하지 않고 인증서 폐기 검사도 수행되지 않습니다.

참고: MQCONN 호출이 아닌 MQCONNX를 사용하는 경우 MQSCO를 통해 인증 정보 레코드(MQAIR)를 제공하도록 선택할 수 있습니다. 따라서 CCDT 파일을 열 수 없는 경우에 MQCONNX의 기본 작동은 실패하지 않지만 사용자가 선택하지 않아도 사용자가 MQAIR을 제공하는 것으로 가정합니다.

EncryptionPolicySuiteB = *string*
채널이 Suite-B 준수 암호화를 사용하는지 및 사용할 강도 레벨을 판별합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

이 속성에는 다음 값이 사용될 수 있습니다.

NONE
Suite-B 준수 암호화는 사용되지 않습니다. 이 설정이 기본 설정입니다.

128_BIT,192_BIT
보안 강도를 128비트 및 192비트 레벨로 설정합니다.

128_BIT
보안 강도를 128비트 레벨로 설정합니다.

192_BIT
보안 강도를 192비트 레벨로 설정합니다.

OCSPAuthentication = OPTIONAL|REQUIRED|WARN
OCSP가 사용 가능하고 OCSP 폐기 검사가 인증서 폐기 상태를 판별할 수 없는 경우에 IBM MQ의 작동을 정의합니다. **ClientRevocationChecks**도 참조하십시오.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

이 속성에는 다음 값이 사용될 수 있습니다.

선택사항
OCSP 검사로 판별할 수 없는 폐기 상태인 모든 인증서가 승인되고 경고 또는 오류 메시지가 생성되지 않습니다. SSL 또는 TLS 연결은 폐기 검사가 수행되지 않은 것처럼 계속됩니다.

REQUIRED
OCSP 검사에서는 검사한 모든 SSL 또는 TLS 인증서에 대해 명확한 폐기 결과가 나와야 합니다. 확인할 수 없는 폐기 상태인 SSL 또는 TLS 인증은 오류 메시지와 함께 거부됩니다. 큐 관리자 SSL 이벤트 메시지가 사용 가능한 경우 ReasonQualifier가 MQRQ_SSL_HANDSHAKE_ERROR인 MQRC_CHANNEL_SSL_ERROR 메시지가 생성됩니다. 연결이 닫혔습니다.

이 값이 기본값입니다.

WARN

OCSP 폐기 검사에서 SSL 또는 TLS 인증서의 폐기 상태를 판별할 수 없는 경우 큐 관리자 오류 로그에 경고가 보고됩니다. 큐 관리자 SSL 이벤트 메시지가 사용 가능한 경우 ReasonQualifier가 MQRQ_SSL_UNKNOWN_REVOCATION인 MQRC_CHANNEL_SSL_WARNING 메시지가 생성됩니다. 연결을 계속할 수 있습니다.

OCSPCheckExtensions=YES|NO

IBM MQ가 AuthorityInfoAccess 인증서 확장자에 대해 작동하는지 여부를 제어합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

값이 `NO`로 설정되면 IBM MQ가 AuthorityInfoAccess 인증서 확장자를 무시하고 OCSP 보안 검사를 시도하지 않습니다. 기본값은 YES입니다.

SSLCryptoHardware = *string*

PKCS #11 암호화 하드웨어가 시스템에 표시되도록 구성하기 위해 필요한 매개변수 문자열을 설정합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

`GSK_PKCS11 =` *`driver path and filename`*`;`*`token label`*`;`*`token password`*`;`*`symmetric cipher setting`*`;` 형식의 문자열을 지정하십시오.

예를 들면, 다음과 같습니다. `GSK_PKCS11=/usr/lib/pkcs11/PKCS11_API.so;tokenlabel;passw0rd;SYMMETRIC_CIPHER_ON`

드라이버 경로는 PKCS #11 카드에 대한 지원을 제공하는 공유 라이브러리의 절대 경로입니다. 드라이버 파일 이름은 공유 라이브러리의 이름입니다. PKCS #11 드라이버 경로 및 파일 이름에 필요한 값의 예는 `/usr/lib/pkcs11/PKCS11_API.so`입니다. GSKit을 통해 대칭 암호 조작에 액세스하려면 대칭 암호 설정 매개변수를 지정하십시오. 이 매개변수의 값은 다음 중 하나입니다.

SYMMETRIC_CIPHER_OFF

대칭 암호 조작에 액세스하지 않습니다. 이 설정이 기본 설정입니다.

SYMMETRIC_CIPHER_ON

대칭 암호 조작에 액세스합니다.

문자열의 최대 길이는 256자입니다. 기본값은 공백입니다. 올바른 형식이 아닌 문자열을 지정하는 경우 오류가 생성됩니다.

Linux UNIX 세미콜론 문자가 주석으로 처리되므로 문자열의 여러 컴포넌트를 제공할 때 백슬래시 문자를 사용하여 세미콜론 문자를 이스케이프 처리해야 합니다. 예: '`\;`'

SSLFipsRequired = YES|NO

IBM MQ에서 암호화가 수행되는 경우 FIPS 인증 알고리즘만 사용할 것인지 여부를 지정합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

암호화 하드웨어가 구성된 경우에 사용되는 암호화 모듈은 하드웨어 제품에 제공된 모듈입니다. 이러한 모듈은 사용 중인 하드웨어 제품에 따라 특정 레벨의 FIPS 인증이거나 이 인증이 아닐 수 있습니다.

SSLHTTPProxyName = *string*

문자열은 OCSP 검사를 위해 GSKit에서 사용될 HTTP 프록시 서버의 호스트 이름 또는 네트워크 주소입니다. 이 주소 뒤에 선택적 포트 번호를 괄호로 묶어 표시할 수 있습니다. 포트 번호가 지정되지 않으면 기본 HTTP 포트 80이 사용됩니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

HP-UX PA-RISC 및 Sun Solaris SPARC 플랫폼에서 및 AIX의 32비트 클라이언트에서 네트워크 주소는 IPv4 주소일 수만 있지만 다른 플랫폼에서는 이 주소가 IPv4 또는 IPv6 주소일 수 있습니다.

예를 들어 방화벽이 OCSP 응답자의 URL에 대한 액세스를 막는 경우 이 속성이 필요할 수 있습니다.

SSLKeyRepository = *pathname*

사용자의 디지털 인증서를 보유하는 키 저장소의 위치이며 스템 형식입니다. 즉, 전체 경로와 확장자가 없는 파일 이름을 포함합니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

SSLKeyResetCount = *integer*|`0`

보안 키가 재협상되기 전에 TLS 채널에서 송신 및 수신된 암호화되지 않은 바이트의 수입니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

값의 범위는 0 - 999999999여야 합니다.

기본값은 0이고 보안 키가 재협상되지 않음을 의미합니다.

값 1 - 32768을 지정하는 경우 TLS 채널은 보안 키 재설정 수 32768(32Kb)을 사용합니다. 이는 보안 키 재설정 값이 작으면 발생할 수 있는 과도한 키 재설정 수를 피하기 위한 것입니다.

클라이언트 구성 파일의 TCP 스탠자

TCP 스탠자를 사용하여 TCP 네트워크 프로토콜 구성 매개변수를 지정할 수 있습니다.

참고: 이 스탠자의 각 속성에 대한 설명은 해당 속성을 읽을 수 있는 IBM MQ 클라이언트를 나타냅니다. 모든 IBM MQ MQI client 구성 파일 스탠자에 대한 요약표는 각 클라이언트가 읽을 수 있는 IBM MQ 속성을 참조하십시오.

다음 속성이 TCP 스탠자에 포함될 수 있습니다.

ClntRcvBuffSize = *number*|`0`

클라이언트 연결 서버 연결 채널의 클라이언트 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

값이 0이면 IBM MQ에 의해 버퍼 크기가 고정되는 것과는 반대로 운영 체제가 버퍼 크기를 관리함을 의미합니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

ClntSndBuffSize = *number*|`0`

클라이언트 연결 서버 연결 채널의 클라이언트 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

값이 0이면 IBM MQ에 의해 버퍼 크기가 고정되는 것과는 반대로 운영 체제가 버퍼 크기를 관리함을 의미합니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Connect_Timeout = *number*

소켓 연결 시도가 제한시간 초과하기 전의 시간(초)입니다. 기본값 0은 연결 제한시간이 없음을 지정합니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java 및 IBM MQ classes for JMS 클라이언트에서 읽을 수 있습니다.

IBM MQ 채널 프로세스는 비차단 소켓을 통해 연결합니다. 그러므로 소켓의 다른 끝이 준비되지 않은 경우, connect()는 *EINPROGRESS* 또는 *EWOULDBLOCK*과 함께 즉시 리턴합니다. 그 후에 통신 오류가 보고되면 연결이 다시 최대 총 20회까지 다시 시도됩니다.

Connect_Timeout이 0이 아닌 값으로 설정된 경우, IBM MQ는 소켓이 준비될 때까지 select() 호출에 대해 규정된 기간 동안 대기합니다. 이 경우 후속 connect() 호출 성공의 가능성이 증가합니다. 이 옵션은 네트워크의 높은 로드로 인해서 연결에 일부 대기 시간이 필요한 상황에서 유익할 수 있습니다.

Connect_Timeout, ClntSndBuffSize, ClntRcvBuffSize 매개변수는 서로 관계가 없습니다.

IPAddressVersion = `MQIPADDR_IPV4|MQIPADDR_IPV6`

채널 연결에 사용할 IP 프로토콜을 지정합니다.

이 속성은 C, 비관리 .NET, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

가능한 문자열 값은 `MQIPADDR_IPV4` 또는 `MQIPADDR_IPV6`입니다. 이러한 값은 **`ALTER QMGR IPADDRV`**의 `IPV4` 및 `IPV6` 과 동일한 의미를 갖습니다.

KeepAlive = YES|NO

KeepAlive 기능을 켜거나 끕니다. KeepAlive=YES로 설정하면 TCP/IP가 연결의 다른 측이 여전히 사용 가능한지 주기적으로 검사합니다. 그렇지 않은 경우 채널이 닫힙니다.

이 속성은 C, 비관리 .NET, IBM MQ classes for Java, IBM MQ classes for JMS, 관리 .NET 및 관리 XMS .NET 클라이언트에서 읽을 수 있습니다.

Windows **Library1 = *DLLName*|WSOCK32**

(Windows 전용) TCP/IP 소켓 DLL의 이름입니다.

이 속성은 C 및 비관리 .NET 클라이언트에서 읽을 수 있습니다.

IBM MQ 환경 변수 사용

명령을 사용하여 현재 설정을 표시하거나 IBM MQ 환경 변수의 값을 재설정할 수 있습니다.

이 태스크 정보

다음과 같은 방식으로 환경 변수를 사용할 수 있습니다.

- 영구 변경사항을 작성하기 위해 시스템 프로파일에서 변수 설정
- 이 세션에 대한 변경사항만 작성하기 위해 명령행에서 명령 실행
- 하나 이상의 변수에 실행 중인 애플리케이션에 따른 특정 값을 지정하도록 애플리케이션이 사용하는 명령 스크립트 파일에 명령 추가

각 환경 변수에 대해 명령을 사용하여 현재 설정을 표시하거나 변수의 값을 재설정할 수 있습니다. 이 명령은 달리 언급된 경우가 아니면 모든 IBM MQ MQI client 플랫폼에서 사용할 수 있습니다.

IBM MQ는 사용자가 설정하지 않은 변수에 대해서는 기본값을 사용합니다.

참고: z/OS IBM MQ for z/OS는 어떠한 IBM MQ 환경 변수도 지원하지 않습니다. 이 플랫폼을 서버로 사용하는 경우 z/OS에서 클라이언트 채널 정의 테이블이 생성되는 방식에 대한 정보는 클라이언트 채널 정의 테이블을 참조하십시오. 클라이언트 플랫폼에서 계속해서 IBM MQ 환경 변수를 사용할 수 있습니다.

프로시저

- Windows

  Windows의 경우 각 환경 변수에 대해 다음과 같은 명령을 사용하여 현재 설정을 표시하거나 변수의 값을 재설정하십시오.

  - 환경 변수의 값을 제거하려면 `SET MQSERVER=` 명령을 사용하십시오.
  - 환경 변수의 현재 설정을 표시하려면 `SET MQSERVER` 명령을 사용하십시오.
  - 세션의 모든 환경 변수를 표시하려면 `set` 명령을 사용하십시오.
- Linux UNIX

  UNIX and Linux의 경우 각 환경 변수에 대해 다음과 같은 명령을 사용하여 현재 설정을 표시하거나 변수의 값을 재설정하십시오.

  - 환경 변수의 값을 제거하려면 `unset MQSERVER` 명령을 사용하십시오.
  - 환경 변수의 현재 설정을 표시하려면 `echo $MQSERVER`명령을 사용하십시오.
  - 세션에 대한 모든 환경 변수를 표시하려면 `set` 명령을 사용하십시오.

관련 태스크

48 페이지의 『구성 파일을 사용하여 클라이언트 구성』

텍스트 파일의 속성을 사용하여 클라이언트를 구성할 수 있습니다. 이러한 속성은 환경 변수 또는 다른 플랫폼별 방식으로 대체될 수 있습니다.

관련 정보

환경 변수

V 9.0.0 MQCCDTURL

MQCCDTURL은 MQCHLLIB 및 MQCHLTAB 환경변수의 조합을 설정하는 것과 동등한 기능을 제공합니다.

주의: 클라이언트로 연결되는 고유 프로그램(즉, C, COBOL 또는 C++ 애플리케이션)에 대해서만 환경 변수 옵션을 사용할 수 있습니다. 환경 변수는 Java, JMS 또는 관리 .NET 애플리케이션에 영향을 주지 않습니다.

IBM MQ는 파일, ftp 또는 http URL에서 CCDT 검색을 지원합니다.

그러나 MQCCDTURL은 URL 값만 승인하며, MQCCDTURL은 기존 로컬 파일 시스템 디렉토리 형식을 승인하지 않습니다.

MQCHLLIB 및 MQCHLTAB 대신 MQCCDTURL을 사용하고, 로컬 파일을 사용하려는 경우 `'file://'` 프로토콜을 사용할 수 있습니다. 따라서 다음과 같습니다.

```
export MQCCDTURL=file:///var/mqm/qmgrs/QMGR/@ipcc/MYCHL.TAB
```

위의 예는 다음과 동일합니다.

```
export MQCHLLIB=/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLTAB=MYCHL.TAB
```

JSON 파일을 사용하는 Windows 예제

```
set MQCCDTURL=file:/c:/mq-channels/CCDT-QMGR1.json
```

위의 예는 다음과 동일합니다.

```
set MQCHLLIB=C:\mq-channels
set MQCHLTAB=CCDT-QMGR1.json
```

관련 개념

39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』
IBM MQ 9.0에서 제품은 MQCNO를 사용하여 프로그래밍하거나 환경 변수를 사용하거나 `mqclient.ini` 파일 스탠자를 사용하여 URL을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾을 수 있는 기능을 제공합니다.

36 페이지의 『클라이언트 채널 정의 테이블』
클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

관련 참조

70 페이지의 『MQCHLLIB』
MQCHLLIB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일에 대한 디렉토리 경로를 지정합니다. 파일은 서버에서 작성되지만, IBM MQ MQI client 워크스테이션에서 복사될 수 있습니다.

72 페이지의 『MQCHLTAB』
MQCHLTAB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일의 이름을 지정합니다. 기본 파일 이름은 AMQCLCHL.TAB입니다.

관련 정보

CCDTURL
XMSC_WMQ_CCDTURL
큐 관리자에 IBM MQ MQI 애플리케이션 연결

MQCCSID

MQCCSID는 사용할 코드화된 문자 세트 번호를 지정하고 서버 구성에 사용된 CCSID 값을 대체합니다.

자세한 정보는 클라이언트 또는 서버 코드화 문자 세트 ID(CCSID)를 참조하십시오.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows에서:

```
SET MQCCSID=number
```

- Linux UNIX UNIX and Linux에서:

```
export MQCCSID=number
```

- IBM i IBM i에서:

```
ADDENVVAR ENVVAR(MQCCSID) VALUE(number)
```

MQCERTLABL

MQCERTLABL은 채널 정의의 인증서 레이블을 정의합니다.

자세한 정보는 인증서 레이블(CERTLABL)을 참조하십시오.

MQCERTVPOL

MQCERTVPOL은 사용되는 인증서 유효성 검증 정책을 지정합니다.

IBM MQ의 인증서 유효성 검증 정책에 대한 자세한 정보는 IBM MQ의 인증서 유효성 검증 정책을 참조하십시오.

이 환경 변수는 클라이언트 ini 파일의 SSL 스탠자에서 *CertificateValPolicy* 설정을 대체합니다. 이 변수는 다음 두 값 중 하나로 설정할 수 있습니다.

ANY
: 기본 보안 소켓 라이브러리가 지원하는 모든 인증서 유효성 검증 정책을 사용합니다.

RFC5280
: RFC 5280 표준을 준수하는 인증서 유효성 검증만 사용합니다.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows에서:

```
SET MQCERTVPOL= value
```

- Linux UNIX UNIX and Linux 시스템의 경우

```
export MQCERTVPOL= value
```

- IBM i IBM i의 경우:

```
ADDENVVAR ENVVAR(MQCERTVPOL) VALUE(value)
```

MQCHLLIB

MQCHLLIB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일에 대한 디렉토리 경로를 지정합니다. 파일은 서버에서 작성되지만, IBM MQ MQI client 워크스테이션에서 복사될 수 있습니다.

MQCHLLIB를 설정하지 않으면 클라이언트에 대한 경로는 기본값 설정됩니다.

- Windows Windows: *MQ_INSTALLATION_PATH*
- Linux UNIX UNIX and Linux: `/var/mqm/`
- IBM i IBM i: `/QIBM/UserData/mqm/`

crtmqm 및 **strmqm** 명령의 경우 이 경로는 두 경로 세트 중 하나로 기본값 설정됩니다. *datapath*를 설정하면 기본 경로는 첫 번째 세트 중 하나입니다. *datapath*를 설정하지 않으면 기본 경로는 두 번째 세트 하나입니다.

- Windows Windows: *datapath*\@ipcc
- Linux UNIX UNIX and Linux: *datapath*/@ipcc
- IBM i IBM i: *datapath*/&ipcc

또는

- Windows Windows: *MQ_INSTALLATION_PATH*\data\qmgrs*qmgrname*\@ipcc
- Linux UNIX UNIX and Linux: /prefix/qmgrs/*qmgrname*/@ipcc
- IBM i IBM i: /*prefix*/qmgrs/*qmgrname*/&ipcc

설명:

- *MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.
- 존재하는 경우 *datapath*는 큐 관리자 스탠자에 정의된 DataPath 값입니다.
- *prefix*는 큐 관리자 스탠자에 정의된 접두부 값입니다. 접두부는 일반적으로 다음 값 중 하나입니다.
  - Linux UNIX UNIX and Linux 시스템의 /var/mqm
  - IBM i IBM i의 /QIBM/UserData/mqm/
- *qmgrname*은 큐 관리자 스탠자에 정의된 Directory 속성 값입니다. 이 값은 실제 큐 관리자 이름과 다를 수 있습니다. 이 값은 특수 문자를 대체하기 위해 변경되었을 수 있습니다.
- 큐 관리자 스탠자가 정의되는 위치는 플랫폼에 따라 다릅니다.
  - IBM i Linux UNIX IBM i, UNIX및 Linux의 mqs.ini 파일에서
  - Windows Windows의 경우 레지스트리

참고사항:

1. z/OS 서버로 IBM MQ for z/OS를 사용하는 경우 이 파일은 IBM MQ 클라이언트 워크스테이션에 보관되어야 합니다.
2. 설정된 경우 MQCHLLIB는 CCDT를 찾은 데 사용되는 경로를 대체합니다.
3. V 9.0.0 MQCHLLIB는 MQCHLTAB 환경 변수와 함께 작동하는 URL을 포함할 수 있습니다(39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』 참조).
4. MQCHLLIB와 같은 환경 변수는 프로세스 또는 작업으로 범위가 설정되거나 플랫폼별 방식에서는 시스템 전체로 범위가 설정될 수 있습니다.
5. 서버에서 MQCHLLIB를 시스템 전체에 설정하는 경우 서버의 모든 큐 관리자에 대해 동일한 CCDT 파일 경로를 설정합니다. MQCHLLIB 환경 변수를 설정하지 않는 경우 이 경로는 각 큐 관리자마다 다릅니다. 이 환경 변수가 **crtmqm** 또는 **strmqm** 명령에서 설정되면 큐 관리자가 MQCHLLIB의 값을 읽습니다.
6. 서버에서 다중 큐 관리자를 작성하는 경우 다음과 같은 이유로 해당 큐 관리자를 구분해야 합니다. MQCHLLIB를 시스템 전체에 설정하면 각 큐 관리자는 동일한 CCDT 파일을 업데이트합니다. 이 파일은 서버에 있는 모든 큐 관리자의 클라이언트 연결 정의를 포함합니다. 다중 큐 관리자에서 동일한 정의가 있을 경우(예: SYSTEM.DEF.CLNTCONN) 이 파일은 최근 정의를 포함합니다. 큐 관리자를 작성할 때 MQCHLLIB가 설정되면 SYSTEM.DEF.CLNTCONN이 CCDT에서 업데이트됩니다. 이 업데이트는 다른 큐 관리자가 작성한 SYSTEM.DEF.CLNTCONN을 덮어씁니다. 이전 정의를 수정한 경우 수정사항이 유실됩니다. 이러한 이유로 서버에서 MQCHLLIB를 시스템 전체 환경 변수로 설정하는 것에 대한 대안을 찾는 것을 고려해야 합니다.
7. 클라이언트 정의에서 MQSC 및 PCF NOREPLACE 옵션은 CCDT 파일의 컨텐츠를 검사하지 않습니다. 이전에 작성한 것과 같은 이름이지만 이 큐 관리자가 아닌 다른 큐 관리자에 의해 작성된 클라이언트 연결 채널 정의

는 NOREPLACE 옵션에 관계없이 대체됩니다. 같은 큐 관리자가 이전에 작성한 정의인 경우 이 정의는 대체되지 않습니다.

8. **rcrmqobj** -t clchltab 명령은 CCDT 파일을 삭제하고 다시 작성합니다. 이 파일은 명령 실행 대상 큐 관리자에서 작성된 클라이언트 연결 정의만 사용하여 다시 작성됩니다.
9. CCDT를 업데이트하는 다른 명령은 같은 채널 이름을 가지는 클라이언트 연결 채널만 수정합니다. 파일의 다른 클라이언트 연결 채널은 대체되지 않습니다.
10. MQCHLLIB에 대한 경로는 인용 부호가 필요하지 않습니다.

예:

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows:

```
SET MQCHLLIB=pathname
```

예를 들면, 다음과 같습니다.

```
SET MQCHLLIB=C:\wmqtest
```

- Linux UNIX UNIX and Linux 시스템의 경우

```
export MQCHLLIB=pathname
```

- IBM i IBM i의 경우:

```
ADDENVVAR ENVVAR(MQCHLLIB) VALUE(pathname)
```

관련 참조

72 페이지의 『MQCHLTAB』

MQCHLTAB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일의 이름을 지정합니다. 기본 파일 이름은 AMQCLCHL.TAB입니다.

관련 정보

큐 관리자에 IBM MQ MQI 애플리케이션 연결

MQCHLTAB

MQCHLTAB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일의 이름을 지정합니다. 기본 파일 이름은 AMQCLCHL.TAB입니다.

서버에서 클라이언트 채널 정의 테이블의 위치에 대한 정보는 36 페이지의 『클라이언트 채널 정의 테이블』의 내용을 참조하십시오.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows에서:

```
SET MQCHLTAB=filename
```

- Linux UNIX UNIX and Linux에서:

```
export MQCHLTAB=filename
```

- IBM i IBM i에서:

```
ADDENVVAR ENVVAR(MQCHLTAB) VALUE(filename)
```

예를 들면, 다음과 같습니다.

```
SET MQCHLTAB=ccdf1.tab
```

클라이언트에서와 같은 방식으로 서버에서 MQCHLTAB 환경 변수는 클라이언트 채널 정의 테이블의 이름을 지정합니다.

관련 개념

39 페이지의 『클라이언트 채널 정의 테이블에 대한 웹 주소 지정 가능 액세스』
IBM MQ 9.0에서 제품은 MQCNO를 사용하여 프로그래밍하거나 환경 변수를 사용하거나 `mqclient.ini` 파일 스탠자를 사용하여 URL을 통해 클라이언트 채널 정의 테이블 (CCDT) 을 찾을 수 있는 기능을 제공합니다.

관련 참조

70 페이지의 『MQCHLLIB』
MQCHLLIB는 클라이언트 채널 정의 테이블(CCDT)이 들어 있는 파일에 대한 디렉토리 경로를 지정합니다. 파일은 서버에서 작성되지만, IBM MQ MQI client 워크스테이션에서 복사될 수 있습니다.

69 페이지의 『MQCCDTURL』
MQCCDTURL은 MQCHLLIB 및 MQCHLTAB 환경변수의 조합을 설정하는 것과 동등한 기능을 제공합니다.

관련 정보

큐 관리자에 IBM MQ MQI 애플리케이션 연결

MQIPADDRV

MQIPADDRV는 채널 연결에 사용할 IP를 지정합니다. "MQIPADDR_IPv4" 또는 "MQIPADDR_IPv6" 의 가능한 문자열 값이 있습니다. 이러한 값은 ALTER QMGR IPADDRV에서 IPv4 및 IPv6 와 동일한 의미를 가집니다. 이 값이 설정되지 않으면 "MQIPADDR_IPv4"로 가정됩니다.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows:

```
SET MQIPADDRV=MQIPADDR_IPv4|MQIPADDR_IPv6
```

- Linux UNIX UNIX and Linux:

```
export MQIPADDRV=MQIPADDR_IPv4|MQIPADDR_IPv6
```

- IBM i IBM i:

```
ADDENVVAR ENVVAR(MQIPADDRV) VALUE(MQIPADDR_IPv4|MQIPADDR_IPv6)
```

MQNAME

MQNAME은 IBM MQ 프로세스가 사용할 수 있는 로컬 NetBIOS 이름을 지정합니다.

클라이언트 및 서버에서 우선순위 규칙 및 전체 설명은 188 페이지의 『Windows에서 NetBIOS 연결 정의』의 내용을 참조하십시오.

이 변수를 설정하려면 이 명령을 사용하십시오.

```
SET MQNAME=Your_env_Name
```

예를 들면, 다음과 같습니다.

```
SET MQNAME=CLIENT1
```

IBM MQ MQI client에서 여러 IBM MQ 애플리케이션을 동시에 실행하고 있는 경우 일부 플랫폼에서 NetBIOS는 애플리케이션별로 다른 이름(MQNAME에 의해 설정됨)을 요구합니다.

MQSERVER

MQSERVER 환경 변수는 최소 채널을 정의하는 데 사용됩니다. MQSERVER는 사용할 통신 메소드 및 IBM MQ 서버의 위치를 지정합니다.

MQSERVER를 사용하여 TLS 채널 또는 채널 엑시트의 채널을 정의할 수는 없습니다. TLS 채널을 정의하는 방법에 대한 자세한 내용은 TLS로 채널 보호를 참조하십시오.

*ConnectionName*은 완전한 네트워크 이름이어야 합니다. 슬래시 문자는 채널 이름, 전송 유형 및 연결 이름을 구분하는 데 사용되기 때문에 *ChannelName*은 슬래시(/) 문자를 포함할 수 없습니다. MQSERVER 환경 변수가 클라이언트 채널을 정의하는 데 사용될 때 최대 메시지 길이(MAXMSGL) 100MB가 사용됩니다. 따라서 채널에 적용되는 최대 메시지 크기는 서버에서 SVRCONN 채널에 지정되는 값입니다.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows:

```
SET  MQSERVER='SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)'
```

- Linux UNIX UNIX and Linux:

```
export  MQSERVER='SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)'
```

- IBM i IBM i:

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)')
```

- z/OS z/OS

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414) '
```

*TransportType*은 IBM MQ 클라이언트 플랫폼에 따라 다음 값 중 하나일 수 있습니다.

- LU62
- TCP
- NETBIOS
- SPX

*ConnectionName*은 쉼표로 구분된 연결 이름 목록일 수 있습니다. 목록에 있는 연결 이름은 클라이언트 연결 테이블의 다중 연결과 비슷한 방식으로 사용됩니다. *ConnectionName* 목록은 큐 관리자 그룹에 대한 대안으로 사용되어 시도할 클라이언트에 대해 다중 연결을 지정할 수 있습니다. 다중 인스턴스 큐 관리자를 구성하는 경우 서로 다른 큐 관리자 인스턴스를 지정하기 위해 *ConnectionName* 목록을 사용할 수 있습니다.

TCP/IP 기본 포트

기본적으로 TCP/IP에 대해 IBM MQ는 채널이 포트 1414에 연결되는 것으로 가정합니다.

이 값을 다음과 같이 변경할 수 있습니다.

- 대괄호 안의 포트 번호를 ConnectionName의 마지막 부분에 추가:

- Windows Windows에서:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(PortNumber)
```

- Linux UNIX UNIX and Linux에서:

```
export MQSERVER='ChannelName/TransportType/ConnectionName(PortNumber)'
```

- 포트 번호를 프로토콜 이름에 추가하여 mqclient.ini 파일 변경. 예를 들면, 다음과 같습니다.

```
TCP:
port=2001
```

- 193 페이지의 『UNIX and Linux에서 TCP/IP 리스너 사용』에 설명된 대로 IBM MQ를 서비스 파일에 추가

SPX 기본 소켓

기본적으로 SPX의 경우, IBM MQ은(는) 채널이 5E86 소켓에 연결되었다고 가정합니다.

이 값을 다음과 같이 변경할 수 있습니다.

- 대괄호 안의 소켓 번호를 ConnectionName의 마지막 부분에 추가:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(SocketNumber)
```

  SPX 연결의 경우, ConnectionName 및 소켓을 `network.node(socket)` 양식으로 지정하십시오. IBM MQ 클라이언트와 서버가 같은 네트워크에 있을 경우 네트워크는 지정하지 않아도 됩니다. 기본 소켓을 사용하는 경우 소켓을 지정하지 않아도 됩니다.
- 포트 번호를 프로토콜 이름에 추가하여 qm.ini 파일 변경. 예를 들면, 다음과 같습니다.

```
SPX:
socket=5E87
```

MQSERVER 사용

MQSERVER 환경 변수를 사용하여 IBM MQ MQI client 시스템과 서버 시스템 간 채널을 정의하는 경우 이 채널이 애플리케이션이 사용할 수 있는 유일한 채널이며 클라이언트 채널 정의 테이블(CCDT)에 대한 참조가 작성되지 않습니다.

이 경우 서버 시스템에서 실행 중인 리스너 프로그램이 애플리케이션을 연결할 큐 관리자를 판별합니다. 이 큐 관리자는 리스너 프로그램이 연결되는 것과 같은 큐 관리자입니다.

MQCONN 또는 MQCONNX 요청이 리스너가 연결된 큐 관리자 외의 큐 관리자를 지정하거나 MQSERVER 매개변수 *TransportType*이 인식되지 않는 경우 MQCONN 또는 MQCONNX 요청이 리턴 코드 MQRC_Q_MGR_NAME_ERROR로 실패합니다.

Linux UNIX UNIX and Linux 시스템에서 MQSERVER를 다음 예 중 하나에서와 같이 정의할 수 있습니다.

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56(2002)'
export MQSERVER=CHANNEL1/LU62/BOX99
```

그러면 MQCONNX에 제공된 MQCNO 구조에서 MQCD 구조가 참조된 경우가 아니면 모든 MQCONN 또는 MQCONNX 요청은 사용자가 정의한 채널을 사용하려고 시도하며 이 경우 MQCD 구조에서 지정하는 채널이 MQSERVER 환경 변수가 지정하는 채널보다 우선합니다.

MQSERVER 환경 변수는 MQCHLLIB 및 MQCHLTAB에 의해 지정되는 클라이언트 채널 정의보다 우선합니다.

MQSERVER 취소

MQSERVER를 취소하고 MQCHLLIB 및 MQCHLTAB가 지정하는 클라이언트 채널 정의 테이블로 돌아가려면 다음을 입력하십시오.

- Windows Windows:

```
SET MQSERVER=
```

- Linux UNIX UNIX and Linux:

```
unset MQSERVER
```

MQSSLCRYP

MQSSLCRYP는 시스템에 표시되는 암호화 하드웨어 구성을 가능하게 하는 매개변수 문자열을 보유합니다. 허용되는 값은 **ALTER QMGR** 명령의 **SSLCRYP** 매개변수에 대한 값과 동일합니다.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows 시스템에서:

```
SET MQSSLCRYP=string
```

- Linux UNIX UNIX and Linux 시스템에서:

```
export MQSSLCRYP=string
```

관련 정보

ALTER QMGR 명령 **SSLCRYP** 매개변수

MQSSLFIPS

MQSSLFIPS는 IBM MQ에서 암호화가 수행되는 경우에 FIPS 인증 알고리즘만 사용할지 여부를 지정합니다. 해당 값은 ALTER QMGR 명령의 SSLFIPS 매개변수에 대한 값과 동일합니다.

FIPS 인증 알고리즘의 사용은 암호화 하드웨어의 영향을 받습니다. MQI 클라이언트에서 런타임 시 FIPS 인증 CipherSpec만 사용하도록 지정을 참조하십시오.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows 시스템에서:

```
SET MQSSLFIPS=YES|NO
```

- Linux UNIX UNIX and Linux 시스템에서:

```
export MQSSLFIPS=YES|NO
```

- IBM i IBM i에서:

```
ADDENVVAR ENVVAR(MQSSLFIPS) VALUE(YES|NO)
```

기본값은 NO입니다.

MQSSLKEYR

MQSSLKEYR은 사용자에 속하는 디지털 인증서가 보관된 키 저장소의 위치를 스템 형식으로 지정합니다. 스템 형식은 전체 경로와 확장자 없는 파일 이름을 포함함을 의미합니다.

자세한 내용은 ALTER QMGR 명령의 SSLKEYR 매개변수를 참조하십시오.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows 시스템에서:

```
SET MQSSLKEYR=pathname
```

- Linux UNIX UNIX and Linux 시스템에서:

```
export MQSSLKEYR=pathname
```

- IBM i IBM i에서:

```
ADDENVVAR ENVVAR(MQSSLKEYR) VALUE(pathname)
```

기본값은 없습니다.

MQSSLPROXY

MQSSLPROXY는 OCSP 검사를 위해 GSKit에서 사용하는 HTTP 프록시 서버의 호스트 이름 또는 포트 번호를 지정합니다.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows 시스템에서:

```
SET MQSSLPROXY= string
```

- Linux UNIX UNIX and Linux 시스템에서:

```
export MQSSLPROXY="string"
```

문자열은 OCSP 검사를 위해 GSKit에서 사용될 HTTP 프록시 서버의 호스트 이름 또는 네트워크 주소입니다. 이 주소 뒤에 선택적 포트 번호를 괄호로 묶어 표시할 수 있습니다. 포트 번호가 지정되지 않으면 기본 HTTP 포트 80이 사용됩니다.

Linux UNIX 예를 들어, UNIX and Linux 시스템에서는 다음 명령 중 하나를 사용할 수 있습니다.

- ```
  export MQSSLPROXY="proxy.example.com(80)"
  ```

- ```
  export MQSSLPROXY="127.0.0.1"
  ```

MQSSLRESET

MQSSLRESET는 보안 키가 재협상되기 전에 TLS 채널에서 송신 및 수신된 암호화되지 않은 바이트 수를 나타냅니다.

보안 키 재협상에 대한 자세한 정보는 TLS 보안 키 재설정을 참조하십시오.

이 변수는 0 - 999 999 999 범위의 정수로 설정될 수 있습니다. 기본값은 0이며 보안 키가 재협상되지 않음을 의미합니다. 1바이트 - 32KB 범위의 TLS 보안 키 재설정 수를 지정하면 TLS 채널은 32KB의 보안 키 재설정 수를

사용합니다. 이 보안 키 재설정 수는 TLS 보안 키 재설정 값이 작으면 발생하게 되는 과도한 키 재설정 수를 피하기 위한 것입니다.

이 변수를 설정하려면 다음 명령 중 하나를 사용하십시오.

- Windows Windows 시스템에서:

```
SET MQSSLRESET=integer
```

- Linux UNIX UNIX and Linux 시스템에서:

```
export MQSSLRESET=integer
```

- IBM i IBM i에서:

```
ADDENVVAR ENVVAR(MQSSLRESET) VALUE(integer)
```

ULW MQSUITEB

UNIX, Linux, and Windows 플랫폼의 NSA Suite B 표준을 준수하도록 IBM MQ 를 구성할 수 있습니다.

스위트 B는 보증된 보안 레벨을 제공하기 위해서 사용 가능한 암호화 알고리즘 세트를 제한합니다.

자세한 정보는 스위트 B에 대한 IBM MQ 구성을 참조하십시오.

MQTCPTIMEOUT

IBM MQ가 TCP 연결 호출을 기다리는 기간입니다.

IBM MQ 및 큐 관리자 구성 정보 변경

설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

이 태스크 정보

IBM MQ를 제어하는 구성 속성(또는 매개변수)의 세트에 지정된 값을 변경하여 IBM MQ 구성 정보를 변경할 수 있습니다.

IBM MQ 구성 파일을 편집하여 속성 정보를 변경하십시오. 노드에서 큐 관리자 구성을 변경하는 명령을 사용하여 자동으로 또는 일반 텍스트 편집기를 사용하여 수동으로 구성 파일을 편집할 수 있습니다. 자세한 정보는 80 페이지의 『구성 파일 편집』 의 내용을 참조하십시오.

Windows Linux Windows 및 Linux(x86 및 x86-64 플랫폼)에서는 IBM MQ Explorer를 사용하여 IBM MQ 구성 파일을 편집할 수 있습니다.

Windows amqmdain에 설명된 대로 Windows 시스템에서는 **amqmdain**을 사용하여 구성 정보를 변경할 수도 있습니다.

프로시저

- 사용자 플랫폼에 대한 IBM MQ 및 큐 관리자 구성에 대한 추가 정보는 다음 하위 주제를 참조하십시오.

관련 개념

5 페이지의 『IBM MQ 구성』
하나 이상의 컴퓨터에서 하나 이상의 큐 관리자를 작성하고 개발, 테스트, 프로덕션 시스템에서 해당 큐 관리자가 비즈니스 데이터를 포함하는 메시지를 처리하도록 구성합니다.

관련 태스크

592 페이지의 『z/OS에서 큐 관리자 구성』

IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

관련 정보

계획 중

IBM MQ 관리

ULW UNIX, Linux, and Windows에서 구성 정보 변경

UNIX, Linux, and Windows에서 구성 파일에 포함되어 있는 IBM MQ 구성 속성을 노드 및 큐 관리자 레벨에서 변경할 수 있습니다.

이 태스크 정보

UNIX, Linux, and Windows 플랫폼에서 다음 파일에 있는 IBM MQ 구성 속성을 변경할 수 있습니다.

- 노드 전체에서 IBM MQ에 대한 변경사항을 적용하기 위한 IBM MQ 구성 파일(`mqs.ini`)입니다. 각 노드에 대해 하나의 `mqs.ini` 파일이 있습니다. `mqs.ini`에 포함된 스탠자 속성에 대한 자세한 정보는 98 페이지의 『IBM MQ 구성 정보를 변경하기 위한 속성』의 내용을 참조하십시오.
- 특정 큐 관리자에 변경사항이 적용되도록 하는 큐 관리자 구성 파일(`qm.ini`)입니다. 노드에 있는 각 큐 관리자에 대해 한 개의 `qm.ini` 파일이 있습니다. `qm.ini`에 포함된 스탠자에 대한 자세한 정보는 104 페이지의 『큐 관리자 구성 정보 변경』의 내용을 참조하십시오.

클라이언트 구성 옵션은 일반적으로 `mqclient.ini` 이름의 클라이언트 구성 파일에서 개별적으로 유지됩니다. 추가 정보는 48 페이지의 『구성 파일을 사용하여 클라이언트 구성』의 내용을 참조하십시오.

구성 파일(또는 스탠자 파일)에는 하나 이상의 스탠자가 포함되며, 이들 스탠자는 `.ini` 파일에서 모두 공통 기능을 갖거나 시스템의 부분(예: 로그 기능, 채널 기능 및 설치 가능한 서비스)을 정의하는 행 그룹입니다.

IBM MQ 구성 파일은 큐 관리자와 연관된 데이터를 찾는 데 사용되므로 존재하지 않거나 잘못된 구성 파일이 일부 또는 모든 MQSC 명령을 실패하게 할 수 있습니다. 또한 애플리케이션은 IBM MQ 구성 파일에 정의되지 않은 큐 관리자에 연결할 수 없습니다.

중요사항: 구성 파일에 작성하는 모든 변경사항은 일반적으로 큐 관리자가 다음에 다시 시작될 때까지 적용되지 않습니다.

프로시저

- 구성 파일을 편집하기 전에 이 파일을 백업하여 필요에 따라 되돌릴 수 있는 사본을 만들어 두십시오.
- 명령 또는 표준 텍스트 편집기를 사용하여 구성 파일을 편집하십시오.

  자세한 정보는 80 페이지의 『구성 파일 편집』을 참조하십시오.
- Windows Linux

  Windows 및 Linux(x86 및 x86-64) 시스템의 경우 구성 파일을 변경하려면 IBM MQ Explorer를 사용하십시오.

  자세한 정보는 MQ 탐색기를 사용하여 IBM MQ 구성을 참조하십시오.
- Windows

  Windows 시스템의 경우 IBM MQ Explorer를 사용하는 대신 **amqmdain** 명령을 사용하여 구성 파일을 변경하십시오.

  자세한 정보는 amqmdain을 참조하십시오.

관련 개념

5 페이지의 『IBM MQ 구성』

하나 이상의 컴퓨터에서 하나 이상의 큐 관리자를 작성하고 개발, 테스트, 프로덕션 시스템에서 해당 큐 관리자가 비즈니스 데이터를 포함하는 메시지를 처리하도록 구성합니다.

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』

설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

87 페이지의 『IBM i에서 구성 정보 변경』
IBM MQ를 제어하는 구성 속성(또는 매개변수) 세트에 지정된 값을 수정하여 설치 필요에 맞게 큐 관리자의 작동을 변경할 수 있습니다.

104 페이지의 『큐 관리자 구성 정보 변경』
개별 큐 관리자의 구성을 수정하는 데 사용할 수 있는 속성은 IBM MQ의 모든 속성을 대체합니다.

관련 참조

98 페이지의 『IBM MQ 구성 정보를 변경하기 위한 속성』
IBM MQ for Windows 시스템 및 IBM MQ for Linux (x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 구성 정보를 수정하십시오. 기타 시스템에서는 mqs.ini 구성 파일을 편집하여 이 정보를 수정합니다.

관련 정보

계획 중

IBM MQ 관리

ULW 구성 파일 편집

명령 또는 표준 문서 편집기를 사용하여 구성 파일을 편집합니다.

구성 파일을 편집하기 전에, 필요에 따라 되돌릴 수 있도록 백업 사본을 만들어 두십시오.

다음과 같은 방법으로 구성 파일을 편집할 수 있습니다.

- 노드의 큐 관리자 구성을 변경하는 명령을 사용하여 자동으로 편집
- 표준 텍스트 편집기를 사용하여 수동으로 편집

설치 후 IBM MQ 구성 파일의 기본값을 편집할 수 있습니다.

구성 파일 속성에 올바르지 않은 값을 설정한 경우, 이 값을 무시하고 문제점을 표시하는 운영자 메시지가 발행됩니다. (결과는 속성 전체를 누락한 것과 동일합니다.)

새 큐 관리자를 작성할 때 다음을 수행하십시오.

- IBM MQ 구성 파일 백업
- 새 큐 관리자 구성 파일 백업

주석 텍스트 앞에 ";" 또는 "#" 문자를 추가하여 주석을 구성 파일에 포함시킬 수 있습니다. 주석을 나타내는 문자 없이 ";" 또는 "#" 문자를 사용하려는 경우 이 문자에 "\" 문자를 접두부로 지정할 수 있습니다. 이는 구성 데이터의 일부로 사용됩니다.

구성 파일을 언제 편집해야 합니까?

구성 파일을 편집하여 백업에서 복구하거나, 큐 관리자를 이동하거나, 기본 큐 관리자를 변경하거나 IBM 지원을 지원합니다.

다음과 같은 경우에 구성 파일을 편집해야 할 수도 있습니다.

- 구성 파일이 손실되었습니다. (가능한 경우 백업에서 복구하십시오.)
- 하나 이상의 큐 관리자를 새 디렉토리로 이동해야 합니다.
- 기본 큐 관리자를 변경해야 합니다. 이는 기존 큐 관리자를 실수로 삭제한 경우에 발생할 수 있습니다.
- IBM 지원 센터는 이를 수행하도록 권고합니다.

구성 파일 우선순위

속성의 값이 여러 위치에서 정의됩니다. 명령에서 설정된 속성이 구성 파일의 속성에 우선합니다.

구성 파일의 속성 값은 다음 우선순위에 따라 설정됩니다.

- 명령행에 입력된 매개변수가 구성 파일에 정의된 값보다 우선합니다.
- qm.ini 파일에서 정의된 값이 mqs.ini 파일에서 정의된 값에 우선합니다.

구성 파일의 주석

Linux UNIX

세미콜론 ';' 및 해시 '#' 문자를 사용하여 구성 파일 내에 주석의 시작을 표시할 수 있습니다. 이는 전체 행을 주석으로 표시하거나 설정 값에 포함되지 않는 행의 끝에 주석을 표시할 수 있습니다.

값에 이러한 문자 중 하나가 필요한 경우, 백슬래시 문자 '\'를 사용하여 이 문자를 이스케이프 처리해야 합니다.

ULW IBM MQ 구성 파일 mqs.ini

IBM MQ 구성 파일 `mqs.ini`에는 노드의 모든 큐 관리자와 관련된 정보가 있습니다. 이 파일은 설치 중 자동으로 작성됩니다.

디렉토리 위치

Linux UNIX UNIX 및 Linux에서 데이터 디렉토리 및 로그 디렉토리는 항상 각각 `/var/mqm` 및 `/var/mqm/log`입니다.

Windows Windows 시스템에서 데이터 디렉토리 `mqs.ini`의 위치 및 로그 디렉토리의 위치는 다를 수 있으므로 레지스트리에 저장됩니다. The installation configuration information, which is contained in `mqinst.ini` on UNIX and Linux systems, is also in the registry, as there is no `mqinst.ini` file on Windows (see 86 페이지의 『설치 구성 파일, mqinst.ini』).

Windows Windows 시스템의 `mqs.ini` 파일은 `HKLM\SOFTWARE\IBM\WebSphere MQ` 키에 지정된 WorkPath 에 의해 제공됩니다. 이 파일은 다음을 포함합니다.

- 큐 관리자의 이름
- 기본 큐 관리자의 이름
- 파일 각각에 연관된 파일의 위치

새 설치용 `LogDefaults` 스탠자

새 IBM MQ 설치의 제공된 `LogDefaults` 스탠자에 속성에 대한 명시 값이 포함되어 있지 않습니다. 속성 부족은 새 큐 관리자 작성 시 이 값에 대한 기본값이 사용됨을 의미합니다. 82 페이지의 그림 5의 `LogDefaults` 스탠자에 대해 기본값이 표시됩니다. `LogBufferPages` 속성의 값이 0이면 512를 의미합니다.

비기본값이 필요한 경우 `LogDefaults` 스탠자에 그 값을 명시적으로 지정해야 합니다.

`mqs.ini` 파일 예

UNIX

```
#*******************************************************************#
#* Module Name: mqs.ini                                            *#
#* Type       : IBM MQ Machine-wide Configuration File             *#
#* Function   : Define IBM MQ resources for an entire machine      *#
#*******************************************************************#
#* Notes      :                                                    *#
#* 1) This is the installation time default configuration          *#
#*                                                                 *#
#*******************************************************************#
AllQueueManagers:
#*******************************************************************#
#* The path to the qmgrs directory, below which queue manager data *#
#* is stored                                                       *#
#*******************************************************************#
DefaultPrefix=/var/mqm

LogDefaults:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=4096
   LogType=CIRCULAR
   LogBufferPages=0
   LogDefaultPath=/var/mqm/log

QueueManager:
   Name=saturn.queue.manager
   Prefix=/var/mqm
   Directory=saturn!queue!manager
   InstallationName=Installation1

QueueManager:
   Name=pluto.queue.manager
   Prefix=/var/mqm
   Directory=pluto!queue!manager
   InstallationName=Installation2

DefaultQueueManager:
   Name=saturn.queue.manager

ApiExitTemplate:
   Name=OurPayrollQueueAuditor
   Sequence=2
   Function=EntryPoint
   Module=/usr/ABC/auditor
   Data=123

ApiExitCommon:
   Name=MQPoliceman
   Sequence=1
   Function=EntryPoint
   Module=/usr/MQPolice/tmqp
   Data=CheckEverything
```

그림 *5. UNIX* 에 대한 *IBM MQ* 구성 파일의 예제

ULW 큐 관리자 구성 파일, qm.ini

큐 관리자 구성 파일인 `qm.ini`에는 특정 큐 관리자에 관한 정보가 있습니다.

큐 관리자마다 한 개의 큐 관리자 구성 파일이 있습니다. `qm.ini` 파일은 연관된 큐 관리자가 작성될 때 자동으로 작성됩니다.

참고: `qm.ini` 파일의 변경사항이 적용되는 시기에 대한 정보는 79 페이지의 『UNIX, Linux, and Windows에서 구성 정보 변경』의 내용을 참조하십시오.

V 9.0.4 V 9.0.0.2 IBM MQ 9.0.4 및 IBM MQ 9.0.0 Fix Pack 2에서는 큐 관리자를 완전히 시작하기 전에 **strmqm** 명령이 `qm.ini` 파일에서 CHANNELS 및 SSL 스탠자의 구문을 검사합니다. 따라서 **strmqm**이 `qm.ini` 파일에 오류가 포함되어 있음을 발견한 경우, 잘못된 사항을 더 쉽게 확인하고 빠르게 수정할 수 있습니다. 자세한 정보는 strmqm을 참조하십시오.

qm.ini 파일의 위치

Linux UNIX UNIX and Linux 시스템의 경우 qm.ini 파일은 큐 관리자가 점유한 디렉토리 트리의 루트에 보관됩니다. 예를 들어, QMNAME이라는 큐 관리자의 구성 파일에 대한 경로 및 이름은 다음과 같습니다.

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

Windows Windows 시스템의 경우 qm.ini 파일의 위치는 HKLM\SOFTWARE\IBM\WebSphere MQ 키에 지정된 WorkPath에 의해 제공됩니다. 예를 들어, QMNAME이라는 큐 관리자의 구성 파일에 대한 경로 및 이름은 다음과 같습니다.

```
C:\ProgramData\IBM\MQ\qmgrs\QMNAME\qm.ini
```

큐 관리자 이름의 최대 길이는 48자입니다. 그러나 이것은 이름이 올바르거나 고유하다는 것을 보장하지 않습니다. 따라서 디렉토리 이름은 큐 관리자 이름을 기반으로 생성됩니다. 이 프로세스를 이름 변환이라고 합니다. 설명은 IBM MQ 파일 이름 이해를 참조하십시오.

qm.ini 파일 예

Linux UNIX

다음 예제는 속성 그룹이 IBM MQ for UNIX 및 Linux 시스템의 큐 관리자 구성 파일에 배열되는 방법을 보여줍니다.

V 9.0.5 IBM MQ 9.0.5부터 큐 관리자 구성 파일의 예는 다음과 같습니다.

```
#* Module Name: qm.ini                                                  *#
#* Type       : IBM MQ queue manager configuration file                 *#
#  Function   : Define the configuration of a single queue manager      *#
#*                                                                      *#
#************************************************************************#
#* Notes      :                                                         *#
#* 1) This file defines the configuration of the queue manager         *#
#*                                                                      *#
#************************************************************************#

ExitPath:
   ExitsDefaultPath=/var/mqm/exits
   ExitsDefaultPath64=/var/mqm/exits64

Service:
   Name=AuthorizationService
   EntryPoints=14

ServiceComponent:
   Service=AuthorizationService
   Name=MQSeries.UNIX.auth.service
   Module=amqzfu
   ComponentDataSize=0

Log:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=4096
   LogType=CIRCULAR
   LogBufferPages=0 1
   LogPath=/var/mqm/log/saturn!queue!manager/

XAResourceManager:
   Name=DB2 Resource Manager Bank
   SwitchFile=/usr/bin/db2swit
   XAOpenString=MQBankDB
   XACloseString=
   ThreadOfControl=THREAD

Channels: 2
   MaxChannels=200
   MaxActiveChannels=100
```

```
   MQIBindType=STANDARD

TCP:
   SndBuffSize=0
   RcvBuffSize=0
   RcvSndBuffSize=0
   RcvRcvBuffSize=0
   ClntSndBuffSize=0
   ClntRcvBuffSize=0
   SvrSndBuffSize=0
   SvrRcvBuffSize=0

QMErrorLog:
   ErrorLogSize=262144
   ExcludeMessage=7234
   SuppressMessage=9001,9002,9202
   SuppressInterval=30

ApiExitLocal:
   Name=ClientApplicationAPIchecker
   Sequence=3
   Function=EntryPoint
   Module=/usr/Dev/ClientAppChecker
   Data=9.20.176.20

TuningParameters:
   ImplSyncOpenOutput=2
```

IBM MQ 9.0.4 이전 및 LTS의 경우 구성 파일의 예는 다음과 같습니다.

```
#* Module Name: qm.ini                                               *#
#* Type       : IBM MQ queue manager configuration file              *#
#  Function   : Define the configuration of a single queue manager *#
#*                                                                   *#
#*******************************************************************#
#* Notes      :                                                      *#
#* 1) This file defines the configuration of the queue manager      *#
#*                                                                   *#
#*******************************************************************#

ExitPath:
   ExitsDefaultPath=/var/mqm/exits
   ExitsDefaultPath64=/var/mqm/exits64

Service:
   Name=AuthorizationService
   EntryPoints=14

ServiceComponent:
   Service=AuthorizationService
   Name=MQSeries.UNIX.auth.service
   Module=amqzfu
   ComponentDataSize=0

Log:
   LogPrimaryFiles=3
   LogSecondaryFiles=2
   LogFilePages=4096
   LogType=CIRCULAR
   LogBufferPages=0 1
   LogPath=/var/mqm/log/saturn!queue!manager/

XAResourceManager:
   Name=DB2 Resource Manager Bank
   SwitchFile=/usr/bin/db2swit
   XAOpenString=MQBankDB
   XACloseString=
   ThreadOfControl=THREAD

Channels: 2
   MaxChannels=200
   MaxActiveChannels=100
   MQIBindType=STANDARD

TCP:
   SndBuffSize=0
   RcvBuffSize=0
```

```
   RcvSndBuffSize=0
   RcvRcvBuffSize=0
   ClntSndBuffSize=0
   ClntRcvBuffSize=0
   SvrSndBuffSize=0
   SvrRcvBuffSize=0

QMErrorLog:
   ErrorLogSize=262144
   ExcludeMessage=7234
   SuppressMessage=9001,9002,9202
   SuppressInterval=30

ApiExitLocal:
   Name=ClientApplicationAPIchecker
   Sequence=3
   Function=EntryPoint
   Module=/usr/Dev/ClientAppChecker
   Data=9.20.176.20
```

참고사항:

1. LogBufferPages의 값 0은 값 512를 제공합니다.
2. Channel 스탠자에 대한 자세한 정보는 172 페이지의 『초기화 및 구성 파일』의 내용을 참조하십시오.
3. XAResourceManager 스탠자의 최대 길이는 255로 제한됩니다. 그러나 트랜잭션 성능 저하를 피하려면 적은 수의 스탠자만 사용해야 합니다.
4. Multi IBM MQ 8.0부터 멀티플랫폼의 새 큐 관리자에 대해 qm.ini file의 TCP 스탠자에 있는 기본 TCP 송신 및 수신 버퍼 크기는 운영 체제에서 관리하도록 설정됩니다. 송신 및 수신 버퍼에 대한 다음 기본 설정을 사용하여 새 큐 관리자가 자동으로 작성됩니다.

   ```
   TCP:
   SndBuffSize=0
   RcvBuffSize=0
   RcvSndBuffSize=0
   RcvRcvBuffSize=0
   ClntSndBuffSize=0
   ClntRcvBuffSize=0
   SvrSndBuffSize=0
   SvrRcvBuffSize=0
   ```

   IBM MQ 8.0 이전에는 수동 성능 조정 없이 이러한 값이 기본적으로 고정된 크기의 32Kb 버퍼로 설정되었습니다.

   이 변경사항은 새 큐 관리자에만 적용됩니다. 이전 릴리스로부터 마이그레이션된 큐 관리자에 대한 TCP 송수신 버퍼 설정은 유지됩니다.

   마이그레이션된 큐 관리자에 대한 운영 체제 기본 작동을 사용하도록 수동으로 TCP 버퍼 크기를 설정하는 방법에 대한 정보는 TCP, LU62, NETBIOS 및 SPX를 참조하십시오.

AccessMode 스탠자

Windows

Windows의 qm.ini 파일에는 추가 스탠자 *AccessMode*가 포함되어 있습니다.

```
AccessMode:
SecurityGroup=wmq\wmq
```

APIExitLocal 스탠자

ApiExitLocal 스탠자는 단일 Module만 지정하도록 허용하지만 다음과 같이 4개의 모듈을 제공해야 합니다.

- 32비트 스레드되지 않음
- 32비트 스레드됨

- 64비트 스레드되지 않음
- 64비트 스레드됨

IBM MQ 는 엑시트의 스레드 버전을 식별하기 위해 제공된 모듈 이름에 _r 를 추가하지만, IBM MQ 는 32비트및 64비트변형에 대해 직접적으로 동등한 메커니즘을 제공하지 않습니다.

규정되지 않은 모듈 이름이 제공되는 경우, IBM MQ은(는) 32비트 변형의 경우 /var/mqm/exits, 64비트 변형의 경우 /var/mqm/exits64에 표시됩니다.

예를 들어, module=amqsaxe는 다음을 의미합니다.

```
/var/mqm/exits/amqsaxe   - 32 bit unthreaded variant
/var/mqm/exits/amqsaxe_r - 32 bit threaded variant
/var/mqm/exits64/amqsaxe - 64 bit unthreaded variant
/var/mqm/exits64/amqsaxe_r - 64 bit threaded variant
```

prefix/mqm/samp/bin에 제공되는 amqsaxe0 및 amqsaxe0_r의 버전은 빌드되는 플랫폼에서 큐 관리자의 원시 크기에 대해 빌드되며(지금은 모두 64비트) 동일한 원시 크기에서 실행 중인 애플리케이션에서만 사용할 수 있습니다.

관련 참조

114 페이지의 『TCP, LU62 및 NETBIOS』

네트워크 프로토콜 구성 매개변수를 지정하려면 이들 큐 관리자 특성 페이지 또는 qm.ini 파일의 스탠자를 사용하십시오. 채널의 기본 속성을 대체합니다.

ULW 설치 구성 파일, mqinst.ini

UNIX 또는 Linux의 경우 설치 구성 파일 mqinst.ini에는 모든 IBM MQ 설치에 대한 정보가 포함되어 있습니다. Windows의 경우 설치 구성 정보가 레지스트리에 있습니다.

mqinst.ini 파일의 위치

Linux UNIX

mqinst.ini 파일은 UNIX and Linux 시스템의 /etc/opt/mqm 디렉토리에 있습니다. 여기에는 어떤 설치(있는 경우)가 기본 설치인지에 대한 정보는 물론, 각 설치에 대한 다음과 같은 정보가 포함되어 있습니다.

- 설치 이름
- 설치 설명
- 설치 ID
- 설치 경로

중요사항: mqinst.ini 파일의 형식은 고정되어 있지 않고 변경이 가능하므로 해당 파일을 직접 편집하거나 참조해서는 안 됩니다.

내부 전용의 설치 ID는 자동으로 설정되므로 변경해서는 안됩니다.

mqinst.ini 파일을 직접 편집하는 대신, 다음 명령을 사용하여 파일에서 값을 작성, 삭제, 조회 및 수정해야 합니다.

입력 항목을 작성하려는 경우 crtmqinst
입력 항목을 삭제하려는 경우 dltmqinst
입력 항목을 표시하려는 경우 dspmqinst
입력 항목을 설정하려는 경우 setmqinst

Windows의 설치 구성 정보

Windows

Windows에 `mqinst.ini` 파일이 없습니다. 설치 구성 정보는 레지스트리의 다음 키에 포함되어 있습니다.

```
HKLM\SOFTWARE\IBM\WebSphere MQ\Installation\InstallationName
```

중요사항: 이 키의 형식은 고정되지 않고 변경이 가능하므로 해당 키를 직접 편집하거나 참조해서는 안됩니다. 대신 레지스트리의 값을 조회 및 수정하려면 다음 명령을 사용해야 합니다.

입력 항목을 표시하려는 경우 dspmqinst
입력 항목을 설정하려는 경우 setmqinst

Windows에서는 **crtmqinst** 및 **dltmqinst** 명령을 사용할 수 없습니다. 설치 및 설치 제거 프로세스는 필요한 레지스트리 입력 항목의 작성과 삭제를 핸들링합니다.

Windows

IBM i IBM i에서 구성 정보 변경

IBM MQ를 제어하는 구성 속성(또는 매개변수) 세트에 지정된 값을 수정하여 설치 필요에 맞게 큐 관리자의 작동을 변경할 수 있습니다.

이 태스크 정보

IBM MQ 구성 파일을 편집하여 구성 속성을 변경할 수 있습니다.

프로시저

- IBM i에서 구성 값 수정에 대한 정보는 다음 주제를 참조하십시오.
  - 88 페이지의 『IBM i용 IBM MQ 구성 파일』
  - 89 페이지의 『IBM i에서 구성 정보 변경을 위한 속성』
  - 91 페이지의 『IBM i에서 큐 관리자 구성 정보 변경』
  - 96 페이지의 『IBM i용 mqs.ini 및 qm.ini 파일 예』

관련 개념

5 페이지의 『IBM MQ 구성』
하나 이상의 컴퓨터에서 하나 이상의 큐 관리자를 작성하고 개발, 테스트, 프로덕션 시스템에서 해당 큐 관리자가 비즈니스 데이터를 포함하는 메시지를 처리하도록 구성합니다.

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

79 페이지의 『UNIX, Linux, and Windows에서 구성 정보 변경』
UNIX, Linux, and Windows에서 구성 파일에 포함되어 있는 IBM MQ 구성 속성을 노드 및 큐 관리자 레벨에서 변경할 수 있습니다.

104 페이지의 『큐 관리자 구성 정보 변경』
개별 큐 관리자의 구성을 수정하는 데 사용할 수 있는 속성은 IBM MQ의 모든 속성을 대체합니다.

관련 참조

98 페이지의 『IBM MQ 구성 정보를 변경하기 위한 속성』
IBM MQ for Windows 시스템 및 IBM MQ for Linux (x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 구성 정보를 수정하십시오. 기타 시스템에서는 mqs.ini 구성 파일을 편집하여 이 정보를 수정합니다.

관련 정보

계획 중
IBM MQ 관리

IBM i IBM i용 IBM MQ 구성 파일

이 정보를 사용하여 IBM MQ for IBM i를 구성하기 위한 메소드에 대해 알아봅니다.

IBM i에서 다음 위치에 있는 IBM MQ 구성 속성을 수정하십시오.

- IBM MQ 구성 파일(`mqs.ini`)은 노드 전체에 대한 변경사항을 적용합니다. 각 IBM MQ 설치마다 하나의 `mqs.ini` 파일이 있습니다.
- 큐 관리자 구성 파일 `qm.ini`는 특정 큐 관리자에 변경사항을 적용합니다. 노드에 있는 각 큐 관리자에 대해 한 개의 `qm.ini` 파일이 있습니다.

.ini 파일은 IFS에 상주하는 스트림 파일임을 참고하십시오.

구성 파일(**스탠자** 파일이라고 부를 수 있음)은 하나 이상의 스탠자를 포함하며 이러한 스탠자는 .ini 파일 내 행의 그룹으로, 함께 공통 기능을 가지거나 시스템의 부분(예: 로그 함수 및 채널 기능)을 정의합니다. 구성 파일에 작성하는 모든 변경사항은 큐 관리자가 다음에 다시 시작될 때까지 적용되지 않습니다.

구성 파일 편집

구성 파일을 편집하기 전에, 필요에 따라 되돌릴 수 있도록 백업 사본을 만들어 두십시오.

다음과 같은 방법으로 구성 파일을 편집할 수 있습니다.

- 노드에서 큐 관리자 구성을 변경하는 명령을 사용하여 자동으로 편집
- EDTF CL 편집기를 사용하여 수동으로 편집

설치 후 IBM MQ 구성 파일의 기본값을 편집할 수 있습니다. 구성 파일 속성에 올바르지 않은 값을 설정한 경우, 이 값을 무시하고 문제점을 표시하는 운영자 메시지가 발행됩니다. (결과는 속성 전체를 누락한 것과 동일합니다.)

새 큐 관리자를 작성할 때 다음을 수행하십시오.

- IBM MQ 구성 파일 백업
- 새 큐 관리자 구성 파일 백업

구성 파일을 언제 편집해야 합니까?

다음과 같은 경우에 구성 파일을 편집해야 할 수도 있습니다.

- 구성 파일이 손실되었으며 가능한 경우 백업에서 복구해야 합니다.
- 하나 이상의 큐 관리자를 새 디렉토리로 이동해야 합니다.
- 기본 큐 관리자를 변경해야 합니다. 이는 기존 큐 관리자를 실수로 삭제한 경우에 발생할 수 있습니다.
- IBM 지원 센터는 이를 수행하도록 권고합니다.

구성 파일 우선순위

구성 파일의 속성 값은 다음 우선순위에 따라 설정됩니다.

- 명령행에 입력된 매개변수가 구성 파일에 정의된 값보다 우선합니다.
- `qm.ini` 파일에서 정의된 값이 `mqs.ini` 파일에서 정의된 값에 우선합니다.

IBM MQ 구성 파일 mqs.ini

IBM MQ 구성 파일 `mqs.ini`에는 IBM MQ 설치에 대한 모든 큐 관리자와 관련된 정보가 포함되어 있습니다. 이 파일은 설치 중 자동으로 작성됩니다. 특히 `mqs.ini` 파일은 각 큐 관리자와 연관된 데이터를 찾는 데 사용됩니다.

`mqs.ini` 파일은 `/QIBM/UserData/mqm`에 저장됩니다.

`mqs.ini` 파일은 다음을 포함합니다.

- 큐 관리자의 이름

- 기본 큐 관리자의 이름
- 각 큐 관리자와 연관된 파일의 위치
- API 엑시트 식별 정보(자세한 정보는 API 엑시트 구성 참조)

큐 관리자 구성 파일, qm.ini

큐 관리자 구성 파일인 `qm.ini`에는 특정 큐 관리자에 관한 정보가 있습니다. 큐 관리자마다 한 개의 큐 관리자 구성 파일이 있습니다. `qm.ini` 파일은 연관된 큐 관리자가 작성될 때 자동으로 작성됩니다.

`qm.ini` 파일은 *mqmdata directory*/*QMNAME*/`qm.ini`에 있습니다. 여기서, *mqmdata directory*은(는) 기본적으로 `/QIBM/UserData/mqm`이고 *QMNAME*은(는) 초기화 파일이 적용되는 큐 관리자의 이름입니다.

참고:

1. `mqs.ini` 파일에서 *mqmdata directory*를 변경할 수 있습니다.
2. 큐 관리자 이름의 최대 길이는 48자입니다. 그러나 이것은 이름이 올바르거나 고유하다는 것을 보장하지 않습니다. 따라서 디렉토리 이름은 큐 관리자 이름을 기반으로 생성됩니다. 이 프로세스를 **이름 변환**이라고 합니다. 자세한 정보는 IBM MQ for IBM i 큐 관리자 라이브러리 이름 이해를 참조하십시오.

IBM i IBM i에서 구성 정보 변경을 위한 속성

이 정보를 사용하여 구성 정보 스탠자에 대해 알아봅니다.

다음 속성 그룹이 `mqs.ini`에서 발생합니다.

- 89 페이지의 『AllQueueManagers 스탠자』
- 90 페이지의 『DefaultQueueManager 스탠자』
- 90 페이지의 『ExitProperties 스탠자』
- 91 페이지의 『QueueManager 스탠자』

API 엑시트 `ApiExitCommon` 및 `ApiExitTemplate`와 연관된 두 개의 스탠자도 있습니다. 이러한 API 엑시트 사용에 대한 자세한 내용은 API 엑시트 구성을 참조하십시오.

AllQueueManagers 스탠자

`AllQueueManagers` 스탠자는 다음을 지정할 수 있습니다.

- 큐 관리자와 연관된 파일이 저장되는 qmgrs 디렉토리에 대한 경로
- 실행 파일 라이브러리에 대한 경로
- EBCDIC 형식 데이터를 ASCII 형식으로 변환하기 위한 메소드

스탠자에 대한 설명에서 밑줄이 그어진 값이 기본값이고 | 기호는 또는을 의미합니다.

DefaultPrefix=*directory_name*

큐 관리자 데이터가 보관되는 qmgrs 디렉토리에 대한 경로입니다. 큐 관리자의 기본 접두부를 변경하는 경우 설치 시 작성된 디렉토리 구조를 복제해야 합니다. 특히 qmgrs 구조를 작성해야 합니다. 기본 접두부를 변경하기 전에 IBM MQ를 중지하고 구조를 새 위치로 이동하고 기본 접두부를 변경한 후에만 IBM MQ를 재시작하십시오.

기본 접두부 변경에 대한 대안으로 환경 변수 MQSPREFIX를 사용하여 CRTMQM 명령의 `DefaultPrefix`를 대체할 수 있습니다.

ConvEBCDICNewline=NL_TO_LF|TABLE|ISO

ASCII의 일부 ISO 변형에는 대등한 문자가 포함되기는 하지만 ASCII 코드 페이지에서 지원되지 않는 줄 바꾸기(NL) 문자가 EBCDIC 코드 페이지에 있습니다.

ConvEBCDICNewline 속성을 사용하여 EBCDIC NL 문자를 ASCII 형식으로 변환할 때 IBM MQ가 사용할 메소드를 지정하십시오.

NL_TO_LF
EBCDIC를 모두 ASCII로 변환하려면 EBCDIC NL 문자(X'15')를 ASCII 줄 바꾸기 문자 LF(X'0A')로 변환하십시오.

NL_TO_LF가 기본값입니다.

TABLE
EBCDIC를 모두 ASCII로 변환하려면 IBM i에서 사용되는 변환 테이블에 따라 EBCDIC NL 문자를 변환하십시오.

이 유형의 변환 영향은 언어에 따라 다를 수 있음을 참고하십시오.

ISO
다음의 경우 ISO를 지정하십시오.

- ISO CCSID가 TABLE 메소드를 사용하여 변환되도록 하려는 경우
- 기타 모든 CCSID가 NL_TO_CF 메소드를 사용하여 변환되도록 하려는 경우

가능한 ISO CCSID는 90 페이지의 표 10에 표시되어 있습니다.

표 *10.* 가능한 *ISO CCSID* 목록

CCSID	**코드 세트**
819	ISO8859-1
912	ISO8859-2
915	ISO8859-5
1089	ISO8859-6
813	ISO8859-7
916	ISO8859-8
920	ISO8859-9
1051	roman8

ASCII CCSID가 ISO 서브세트가 아닌 경우 ConvEBCDICNewline은 NL_TO_LF로 기본값 설정됩니다.

DefaultQueueManager 스탠자

`DefaultQueueManager` 스탠자는 노드를 위한 기본 큐 관리자를 지정합니다.

Name=*default_queue_manager*
기본 큐 관리자는 큐 관리자 이름을 명시적으로 지정하지 않은 명령을 처리합니다. 새 기본 큐 관리자를 작성하면 `DefaultQueueManager` 속성이 자동으로 업데이트됩니다. 의도치 않게 기본 큐 관리자를 작성한 후 다시 원래의 기본 큐 관리자로 되돌리려는 경우 수동으로 `DefaultQueueManager` 속성을 대체해야 합니다.

ExitProperties 스탠자

`ExitProperties` 스탠자는 큐 관리자 엑시트 프로그램이 사용하는 구성 옵션을 지정합니다.

스탠자에 대한 설명에서 밑줄이 그어진 값이 기본값이고 | 기호는 또는을 의미합니다.

CLWLMode= SAFE |FAST
클러스터 워크로드 엑시트인 CLWL을 사용하면 클러스터에서 MQI 호출(예: MQOPEN 또는 MQPUT)에 대한 응답으로 열리는 클러스터 큐를 지정할 수 있습니다. CLWL 엑시트는 CLWLMode 속성에 지정한 값에 따라 FAST 모드 또는 SAFE 모드로 실행됩니다. CLWLMode 속성을 생략하면 클러스터 워크로드 엑시트가 SAFE 모드로 실행됩니다.

안전
큐 관리자에 대한 별도의 프로세스로 CLWL 엑시트를 실행하십시오. 기본값입니다.

SAFE 모드로 실행할 때 사용자 작성 CLWL 엑시트에서 문제점이 발생하면 다음이 발생합니다.

- CLWL 서버 프로세스(amqzlwa0)가 실패합니다.
- 큐 관리자가 CLWL 서버 프로세스를 재시작합니다.
- 오류 로그에 오류가 보고됩니다. MQI 호출이 진행 중인 경우 잘못된 리턴 코드 양식으로 된 알림이 수신됩니다.

큐 관리자의 무결성은 보존됩니다.

참고: 별도의 프로세스로 CLWL 엑시트를 실행하는 것에는 성능상 악영향이 있을 수 있습니다.

FAST

큐 관리자 프로세스에서 클러스터 엑시트 인라인을 실행합니다.

이 옵션을 지정하면 SAFE 모드 실행과 연관된 오버헤드를 방지하여 성능을 향상시키지만 큐 관리자 무결성은 손상됩니다. CLWL 엑시트에 문제점이 **없음**을 확신하고 성능 오버헤드가 특히 걱정되는 경우에만 CLWL 엑시트를 FAST 모드로 실행하십시오.

CLWL 엑시트를 FAST 모드로 실행하는 동안 문제점이 발생하면 큐 관리자는 실패하고 큐 관리자의 무결성이 손상될 위험이 있습니다.

QueueManager 스탠자

큐 관리자마다 한 개의 `QueueManager` 스탠자가 있습니다. 이러한 속성은 해당 큐 관리자와 연관된 파일이 들어 있는 디렉토리의 이름 및 큐 관리자 이름을 지정합니다. 이 디렉토리의 이름은 큐 관리자 이름을 기반으로 하지만 큐 관리자 이름이 유효한 파일 이름이 아닌 경우 변환됩니다.

이름 변환에 대한 자세한 정보는 IBM MQ for IBM i 큐 관리자 라이브러리 이름 이해를 참조하십시오.

Name=*queue_manager_name*

대기열 관리자의 이름입니다.

Prefix=*prefix*

큐 관리자 파일이 저장되는 위치입니다. 기본적으로 이는 `mqs.ini` 파일에서 `AllQueueManager` 스탠자의 DefaultPrefix 속성에 지정된 값과 같습니다.

Directory=*name*

큐 관리자 파일이 저장되는 *prefix*`\QMGRS` 디렉토리 아래에 있는 서브디렉토리의 이름입니다. 이 이름은 큐 관리자 이름을 기반으로 하지만 중복 이름이 있거나 큐 관리자 이름이 유효한 파일 이름이 아닌 경우 변환될 수 있습니다.

Library=*name*

IBM i 오브젝트가 이 큐 관리자(예: 저널 및 저널 수신자)에 관련된 라이브러리의 이름이 저장됩니다. 이 이름은 큐 관리자 이름을 기반으로 하지만 중복 이름이 있거나 큐 관리자 이름이 유효한 라이브러리 이름이 아닌 경우 변환될 수 있습니다.

IBM i IBM i에서 큐 관리자 구성 정보 변경

이 정보를 사용하여 큐 관리자 구성 스탠자에 대해 알아봅니다.

API 엑시트 `ApiExitCommon` 및 `ApiExitTemplate`와 연관된 두 개의 스탠자가 있습니다. 이러한 스탠자 사용에 대한 자세한 내용은 API 엑시트 구성을 참조하십시오.

다음 속성 그룹은 특정 큐 관리자의 `qm.ini` 파일에서 발생하거나 `mqs.ini`에서 설정되는 값을 대체하는 데 사용될 수 있습니다.

특정 옵션에 대한 구성 정보 변경에 대해서는 다음 주제를 참조하십시오.

- 92 페이지의 『IBM i의 로그 스탠자』
- 92 페이지의 『IBM i의 채널 스탠자』
- 93 페이지의 『IBM i의 QMErrorLog 스탠자』
- 95 페이지의 『IBM i의 TCP 스탠자』
- 63 페이지의 『클라이언트 구성 파일의 PreConnect 스탠자』

IBM i IBM i의 로그 스탠자

로그 파일을 구성하기 위한 매개변수입니다.

`Log` 스탠자는 특정 큐 관리자의 로그 속성을 지정합니다. 기본적으로 이 속성은 큐 관리자가 작성될 때 `mqs.ini` 파일의 `LogDefaults` 스탠자에 지정된 설정에서 상속됩니다.

이 스탠자의 속성은 큐 관리자를 다른 큐 관리자와 다르게 구성하려는 경우에만 변경하십시오.

`qm.ini` 파일의 속성에 지정된 값은 큐 관리자가 시작될 때 읽힙니다. 이 파일은 큐 관리자가 작성될 때 작성됩니다.

LogBufferSize
저널 버퍼 크기(바이트)입니다. 32 000 - 15 761 440 범위의 숫자를 입력하십시오. 기본값은 32 000입니다.

LogPath=*library_name*
이 큐 관리자의 저널 및 저널 수신자를 저장하는 데 사용되는 라이브러리의 이름입니다.

LogReceiverSize
저널 수신자 크기(KB)입니다. 기본값은 100 000입니다.

IBM i IBM i의 채널 스탠자

`Channels` 스탠자는 채널에 대한 정보를 포함합니다.

MaxChannels=100|*number*
허용되는 현재 채널의 최대 수입니다. z/OS에서 이 값은 1 - 9999 범위여야 하고 기본값은 200입니다. 다른 모든 플랫폼에서 기본값은 100입니다.

MaxActiveChannels=*MaxChannels_value*
언제든지 활성일 수 있는 채널의 최대 수입니다. 기본값은 `MaxChannels` 속성에서 지정되는 값입니다.

MaxInitiators=3|*number*
시작기의 최대 수입니다. 기본값 및 최대값은 3입니다.

MQIBINDTYPE=FASTPATH|STANDARD
애플리케이션에 대한 바인딩입니다.

FASTPATH
채널이 MQCONNX FASTPATH를 사용하여 연결합니다. 즉, 에이전트 프로세스가 없습니다.

STANDARD
채널이 STANDARD를 사용하여 연결합니다.

ThreadedListener= NO |YES
리스너로 RUNMQLSR(YES)을 시작할지 또는 AMQCLMAA(NO)를 시작할지 여부입니다.

ThreadedListener=YES를 지정하는 경우 채널이 단일 작업의 스레드로 실행됩니다. 이는 단일 작업에 사용 가능한 자원에 대한 연결을 제한합니다.

ThreadedListener=NO를 지정하는 경우 비스레드 리스너(AMQCLMAA)가 각 인바운드 TCP/IP 채널에 대해 새 응답자 작업(AMQCRSTA)을 시작합니다. 이 기술의 단점은 RUNMQLSR 작업에서 스레드를 시작할 때 새 AMQCRSTA를 시작하는 것처럼 빠르지 않으므로 스레드화되지 않은 리스너에 대한 연결 시간이 스레드화된 리스너에 대한 연결 시간에 비해 느리다는 점입니다.

AdoptNewMCA= NO |SVR|SNDR|RCVR|CLUSRCVR|ALL|FASTPATH
IBM MQ가 채널 시작 요청을 수신했지만 같은 채널에 대해 amqcrsta 프로세스가 존재하는 것을 알게 되는 경우 새 프로세스를 시작하려면 먼저 기존 프로세스를 중지해야 합니다. `AdoptNewMCA` 속성을 사용하면 기존 프로세스의 종료와 지정된 채널 유형에 대한 새 프로세스 시동을 제어할 수 있습니다.

지정된 채널 유형에 대해 `AdoptNewMCA` 속성을 지정하지만 이미 실행 중이어서 새 채널이 시작에 실패하는 경우:

1. 새 채널이 이전 채널을 종료하려고 시도합니다.
2. AdoptNewMCATimeout 대기 간격이 만료될 때까지 이전 채널 서버가 종료되지 않으면 이전 채널 서버에 대한 프로세스(또는 스레드)가 종료됩니다.

3. 이전 채널 서버가 단계 2 후 및 AdoptNewMCATimeout 대기 간격이 두 번째로 만료된 후에도 종료되지 않으면 IBM MQ가 `CHANNEL IN USE` 오류로 채널을 종료합니다.

다음 목록에서 쉼표 또는 공백으로 분리된 하나 이상의 값을 지정합니다.

아니오
: `AdoptNewMCA` 기능은 필요하지 않습니다. 기본값입니다.

SVR
: 서버 채널을 채택합니다.

SNDR
: 송신자 채널을 채택합니다.

RCVR
: 수신자 채널을 채택합니다.

CLUSRCVR
: 클러스터 수신자 채널을 채택합니다.

모두
: FASTPATH 채널을 제외한 모든 채널 유형을 채택합니다.

FASTPATH
: FASTPATH 채널인 경우 이 채널을 채택합니다. 이는 적절한 채널 유형(예: AdoptNewMCA=RCVR,SVR,FASTPATH)도 지정된 경우에만 발생합니다.

주의!: AdoptNewMCA 속성은 큐 관리자의 내부 설계로 인해 FASTPATH 채널을 사용하여 예측할 수 없는 방식으로 작동할 수 있습니다. FASTPATH 채널에 대해 `AdoptNewMCA` 속성을 사용으로 설정할 때에는 매우 주의해야 합니다.

AdoptNewMCATimeout= 60 |1-3600
: 새 프로세스가 이전 프로세스가 종료될 때까지 대기하는 시간의 양(초)입니다. 1 - 3600 범위의 값(초)을 지정하십시오. 기본값은 60입니다.

AdoptNewMCACheck=QM|ADDRESS|NAME|ALL
: `AdoptNewMCACheck` 속성을 사용하면 `AdoptNewMCA` 속성을 사용으로 설정할 때 필요한 유형 검사를 지정할 수 있습니다. 채널이 의도치 않게 또는 악의적으로 종료되지 않도록 보호하기 위해 가능한 경우 다음 세 가지 검사 모두 수행하는 것이 중요합니다. 최소한 채널 이름이 일치하는지는 검사하십시오.

다음에서 쉼표 또는 공백으로 분리된 하나 이상의 값을 지정합니다.

QM
: 리스너 프로세스가 큐 관리자 이름이 일치하는지 검사합니다.

ADDRESS
: 리스너 프로세스가 통신 주소(예: TCP/IP 주소)를 검사합니다.

이름
: 리스너 프로세스가 채널 이름이 일치하는지 검사합니다.

모두
: 리스너 프로세스가 일치하는 큐 관리자 이름, 통신 주소 및 일치하는 채널 이름이 있는지 검사합니다.

기본값은 `AdoptNewMCACheck=NAME,ADDRESS,QM`입니다.

관련 개념

156 페이지의 『채널 상태』
채널은 언제든지 여러 상태 중 하나일 수 있습니다. 일부 상태에는 하위 상태도 있습니다. 채널은 제공된 상태에서 다른 상태로 이동할 수 있습니다.

IBM i *IBM i*의 *QMErrorLog* 스탠자

`qm.ini` 파일의 `QMErrorLog` 스탠자를 사용하여 큐 관리자 오류 로그의 조작 및 컨텐츠를 사용자 정의하십시오.

V 9.0.4 **ErrorLogSize=** ***maxsize***

백업으로 복사할 큐 관리자 오류 로그의 크기를 지정합니다. *maxsize*는 32768 - 2147483648바이트 범위에 있어야 합니다. **ErrorLogSize**를 지정하지 않은 경우 기본값인 33554432바이트(32MB)가 사용됩니다.

필요한 경우 이 속성을 사용하여 최대 크기를 다시 이전 최대 크기인 2MB로 줄일 수 있습니다.

중요사항: IBM MQ 9.0.4에서는 **ErrorLogSize** 속성이 기본 크기가 늘어났습니다. 이는 IBM MQ 9.0.3에서 변경된 것입니다.

MQMAXERRORLOGSIZE 환경 변수를 사용하여 로그의 크기를 설정할 수 있습니다.

ExcludeMessage= ***msgIds***

큐 관리자 오류 로그에 기록하지 않을 메시지를 지정합니다. *msqIds*에는 다음에서 제공하는 메시지 ID가 쉼표로 구분되어 있습니다.

7163 - 작업이 시작됨 메시지(IBM i만 해당됨)
7234 - 로드된 메시지 수
8245
9001 - 채널 프로그램이 정상적으로 종료됨
9002 - 채널 프로그램이 시작됨
9202 - 원격 호스트가 사용 불가능함
9208 - 호스트로부터 수신 시 오류
9209 - 연결 종료됨
9228 - 채널 응답자를 시작할 수 없음
9508 - 큐 관리자에 연결할 수 없음
9524 - 리모트 큐 관리자가 사용 불가능함
9528 - 사용자가 채널을 닫도록 요청함
9558 - 원격 채널 사용 불가능
9776 - 채널이 사용자 ID에 의해 차단됨
9777 - 채널이 NOACCESS 맵에 의해 차단됨
9782 - 연결이 주소에 의해 차단됨
9999 - 채널 프로그램이 비정상적으로 종료됨

SuppressMessage= ***msgIds***

지정된 시간 간격에 한 번만 큐 관리자 오류 로그에 기록되는 메시지를 지정합니다. 시간 간격은 **SuppressInterval**에 의해 지정됩니다. *msqIds*에는 다음에서 제공하는 메시지 ID가 쉼표로 구분되어 있습니다.

7163 - 작업이 시작됨 메시지(IBM i만 해당됨)
7234 - 로드된 메시지 수
8245
9001 - 채널 프로그램이 정상적으로 종료됨
9002 - 채널 프로그램이 시작됨
9202 - 원격 호스트가 사용 불가능함
9208 - 호스트로부터 수신 시 오류
9209 - 연결 종료됨
9228 - 채널 응답자를 시작할 수 없음
9508 - 큐 관리자에 연결할 수 없음
9524 - 리모트 큐 관리자가 사용 불가능함
9528 - 사용자가 채널을 닫도록 요청함
9558 - 원격 채널 사용 불가능
9776 - 채널이 사용자 ID에 의해 차단됨
9777 - 채널이 NOACCESS 맵에 의해 차단됨
9782 - 연결이 주소에 의해 차단됨
9999 - 채널 프로그램이 비정상적으로 종료됨

`SuppressMessage`와 `ExcludeMessage`에 동일한 메시지 ID가 지정될 경우 해당 메시지가 제외됩니다.

SuppressInterval= ***length***
`SuppressMessage`에 지정된 메시지가 큐 관리자 오류 로그에 한 번만 기록되는 시간 간격(초)을 지정합니다. *length*는 1 - 86400초 범위에 있어야 합니다. `SuppressInterval`이 지정되지 않으면 기본값 30초가 사용됩니다.

IBM i *IBM i*의 *TCP* 스탠자

다음 큐 관리자 특성 페이지 또는 `qm.ini` 파일의 스탠자를 사용하여 네트워크 프로토콜 구성 매개변수를 지정합니다. 채널의 기본 속성을 대체합니다.

참고: 기본값에 대한 변경사항을 나타내는 속성만 지정해야 합니다.

TCP
다음 속성을 지정할 수 있습니다.

Port=1414|*port_number*
TCP/IP 세션에 대한 10진수 표기법의 기본 포트 번호입니다. IBM MQ 8.0의 기본 포트 번호는 1414입니다.

KeepAlive= NO |YES
KeepAlive 기능을 켜거나 끕니다. KeepAlive=YES로 설정하면 TCP/IP가 연결의 다른 측이 여전히 사용 가능한지 주기적으로 검사합니다. 그렇지 않은 경우 채널이 닫힙니다.

ListenerBacklog=number
TCP/IP에서 수신 시, 최대 미해결 연결 요청 수가 설정됩니다. 이 수는 리스너가 요청을 승인할 때까지 TCP/IP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다. IBM i에 대한 기본 리스너 백로그 값은 255이고 최대값은 512입니다. 백로그가 값 512에 도달하면 TCP/IP 연결이 거부되고 채널을 시작할 수 없습니다.

MCA 채널의 경우에는, 채널이 RETRY 상태가 되고 나중에 연결을 재시도하게 됩니다.

클라이언트 연결의 경우, 클라이언트가 MQCONN에서 MQRC_Q_MGR_NOT_AVAILABLE 이유 코드를 수신하며 나중에 연결을 재시도해야 합니다.

`ListenerBacklog` 속성을 사용하면 TCP/IP 리스너에 대한 미해결 요청 수의 기본값을 대체할 수 있습니다.

Connect_Timeout=*number*|0
소켓 연결 시도가 제한시간 초과하기 전의 시간(초)입니다. 기본값 0은 연결 제한시간이 없음을 지정합니다.

다음 특성 그룹은 TCP/IP에서 사용되는 버퍼의 크기를 제어하는 데 사용할 수 있습니다. 값은 운영 체제의 TCP/IP 계층으로 직접 전달됩니다. 이러한 특성을 사용할 때에는 매우 주의해야 합니다. 값이 잘못 설정되면 TCP/IP 성능에 부정적인 영향을 미칠 수 있습니다. 이것이 성능에 미치는 영향에 대한 추가 정보는 사용자 환경에 대한 TCP/IP 문서를 참조하십시오. 값이 0이면 IBM MQ에 의해 버퍼 크기가 고정되는 것과는 반대로 운영 체제가 버퍼 크기를 관리함을 의미합니다.

SndBuffSize=*number*|0
채널의 송신 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다. 이 스탠자 값은 채널 유형에 더 특정한 스탠자(예: RcvSndBuffSize)에 의해 대체될 수 있습니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

RcvBuffSize=*number*|0
채널의 수신 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다. 이 스탠자 값은 채널 유형에 더 특정한 스탠자(예: RcvRcvBuffSize)에 의해 대체될 수 있습니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

RcvSndBuffSize=*number*|0
수신자 채널의 송신자 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

RcvRcvBuffSize=*number*|0
수신자 채널의 수신 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

SvrSndBuffSize=*number*|0
: 클라이언트 연결 서버 연결 채널의 서버 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

SvrRcvBuffSize=*number*|0
: 클라이언트 연결 서버 연결 채널의 서버 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

V 9.0.5 IBM i *IBM i*의 성능 조정 매개변수 스탠자

큐 관리자의 성능 조정을 위해 옵션을 지정하려면 `qm.ini` 파일에서 `TuningParameters` 스탠자를 사용하십시오.

ImplSyncOpenOutput=*value*

`ImplSyncOpenOutput`은 동기점 외부에서 지속적 넣기(PUT)에 내재된 동기점을 사용할 수 있기 전에 넣기(PUT)를 위해 큐가 열려 있는 최소 애플리케이션 수입니다. **`ImplSyncOpenOutput`**의 기본값은 2입니다.

이는 넣기(PUT) 조작을 위해 큐가 열려 있는 애플리케이션이 하나일 경우 **`ImplSyncOpenOutput`**이 꺼짐으로 전환되는 효과가 있습니다.

`ImplSyncOpenOutput`=1을 지정하면 내재된 동기화가 항상 고려됩니다.

양의 정수 값을 설정할 수 있습니다. 내재된 동기점을 추가하지 않으려면 ImplSyncOpenOutput=OFF로 설정하십시오.

관련 정보

암시적 동기점

IBM i IBM i용 mqs.ini 및 qm.ini 파일 예

IBM MQ 구성 파일 예

다음 예제는 IBM i에 대한 `mqs.ini` 파일을 표시합니다.

```
#*******************************************************************#
#* Module Name: mqs.ini                                            *#
#* Type       : IBM MQ Configuration File                          *#
#* Function   : Define IBM MQ resources for the node               *#
#*                                                                 *#
#*******************************************************************#
#* Notes      :                                                    *#
#* 1) This is an example IBM MQ configuration file                 *#
#*                                                                 *#
#*******************************************************************#
AllQueueManagers:
#*******************************************************************#
#* The path to the qmgrs directory, within which queue manager data *#
#* is stored                                                       *#
#*******************************************************************#
DefaultPrefix=/QIBM/UserData/mqm

QueueManager:
Name=saturn.queue.manager
Prefix=/QIBM/UserData/mqm
Library=QMSATURN.Q
Directory=saturn!queue!manager

QueueManager:
Name=pluto.queue.manager
Prefix=/QIBM/UserData/mqm
Library=QMPLUTO.QU
Directory=pluto!queue!manager

DefaultQueueManager:
Name=saturn.queue.manager
```

큐 관리자 구성 파일 예

V 9.0.5 IBM MQ 9.0.5에서 다음 예제는 속성 그룹이 IBM i의 큐 관리자 구성 파일에 배열되는 방법을 보여줍니다.

```
#*******************************************************************#
#* Module Name: qm.ini                                             *#
#* Type        : IBM MQ queue manager configuration file           *#
# Function     : Define the configuration of a single queue manager *#
#*                                                                 *#
#*******************************************************************#
#* Notes       :                                                   *#
#* 1) This file defines the configuration of the queue manager     *#
#*                                                                 *#
#*******************************************************************#
Log:
LogPath=QMSATURN.Q
LogReceiverSize=65536

CHANNELS:
MaxChannels = 20       ; Maximum number of channels allowed.
                       ; Default is 100.
MaxActiveChannels = 10 ; Maximum number of channels allowed to be
                       ; active at any time. The default is the
                       ; value of MaxChannels.

TCP:                   ; TCP/IP entries.
KeepAlive = Yes        ; Switch KeepAlive on.
SvrSndBuffSize=20000   ; Size in bytes of the TCP/IP send buffer for each
                       ; channel instance. Default is 32768.
SvrRcvBuffSize=20000   ; Size in bytes of the TCP/IP receive buffer for each
                       ; channel instance. Default is 32768.
Connect_Timeout=10000  ; Number of seconds before an attempt to connect the
                       ; channel instance times out. Default is zero (no timeout).

QMErrorLog:
ErrorLogSize = 262144
ExcludeMessage = 7234
SuppressMessage = 9001,9002,9202
SuppressInterval = 30

TuningParameters:
   ImplSyncOpenOutput=2
```

IBM MQ 9.0.4 이전 및 LTS의 경우 구성 파일의 예는 다음과 같습니다.

```
#*******************************************************************#
#* Module Name: qm.ini                                             *#
#* Type        : IBM MQ queue manager configuration file           *#
# Function     : Define the configuration of a single queue manager *#
#*                                                                 *#
#*******************************************************************#
#* Notes       :                                                   *#
#* 1) This file defines the configuration of the queue manager     *#
#*                                                                 *#
#*******************************************************************#
Log:
LogPath=QMSATURN.Q
LogReceiverSize=65536

CHANNELS:
MaxChannels = 20       ; Maximum number of channels allowed.
                       ; Default is 100.
MaxActiveChannels = 10 ; Maximum number of channels allowed to be
                       ; active at any time. The default is the
                       ; value of MaxChannels.

TCP:                   ; TCP/IP entries.
KeepAlive = Yes        ; Switch KeepAlive on.
SvrSndBuffSize=20000   ; Size in bytes of the TCP/IP send buffer for each
                       ; channel instance. Default is 32768.
SvrRcvBuffSize=20000   ; Size in bytes of the TCP/IP receive buffer for each
                       ; channel instance. Default is 32768.
Connect_Timeout=10000  ; Number of seconds before an attempt to connect the
                       ; channel instance times out. Default is zero (no timeout).
```

```
QMErrorLog:
ErrorLogSize = 262144
ExcludeMessage = 7234
SuppressMessage = 9001,9002,9202
SuppressInterval = 30
```

참고사항:

1. 노드에 있는 IBM MQ는 큐 관리자 및 저널의 기본 위치를 사용합니다.
2. 큐 관리자 saturn.queue.manager가 이 노드의 기본 큐 관리자입니다. 이 큐 관리자와 연관된 파일의 디렉토리가 파일 시스템에 대해 유효한 파일 이름으로 자동 변환되었습니다.
3. IBM MQ 구성 파일이 큐 관리자와 연관된 데이터를 찾는 데 사용되므로 존재하지 않거나 잘못된 구성 파일로 인해 일부 또는 모든 IBM MQ 명령이 실패할 수 있습니다. 또한 애플리케이션은 IBM MQ 구성 파일에 정의되지 않은 큐 관리자에 연결할 수 없습니다.

IBM MQ 구성 정보를 변경하기 위한 속성

IBM MQ for Windows 시스템 및 IBM MQ for Linux (x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 구성 정보를 수정하십시오. 기타 시스템에서는 mqs.ini 구성 파일을 편집하여 이 정보를 수정합니다.

특정 컴포넌트의 속성에 대해서는 다음 하위 주제를 참조하십시오.

관련 개념

5 페이지의 『IBM MQ 구성』
하나 이상의 컴퓨터에서 하나 이상의 큐 관리자를 작성하고 개발, 테스트, 프로덕션 시스템에서 해당 큐 관리자가 비즈니스 데이터를 포함하는 메시지를 처리하도록 구성합니다.

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

104 페이지의 『큐 관리자 구성 정보 변경』
개별 큐 관리자의 구성을 수정하는 데 사용할 수 있는 속성은 IBM MQ의 모든 속성을 대체합니다.

관련 정보

계획 중

IBM MQ 관리

모든 큐 관리자

IBM MQ Explorer의 `General` 및 `Extended` IBM MQ 특성 페이지 또는 mqs.ini 파일의 `AllQueueManagers` 스탠자를 사용하여 모든 큐 관리자에 대한 다음 정보를 지정하십시오.

DefaultPrefix=*directory_name*

이 속성은 큐 관리자 데이터가 보관되는 qmgrs 디렉토리에 대한 경로를 지정합니다.

큐 관리자의 기본 접두부를 변경하는 경우 설치 시 작성된 디렉토리 구조를 복제하십시오.

특히 qmgrs 구조를 작성해야 합니다. 기본 접두부를 변경하기 전에 IBM MQ를 중지하고 구조를 새 위치로 이동하고 기본 접두부를 변경한 후에만 IBM MQ를 재시작하십시오.

참고: UNIX and Linux 시스템에서 `/var/mqm/errors` 디렉토리를 삭제하거나 Windows 시스템에서 `\errors` 디렉토리를 삭제하지 마십시오.

기본 접두부 변경에 대한 대안으로 환경 변수 MQSPREFIX를 사용하여 `crtmqm` 명령의 **`DefaultPrefix`**를 대체할 수 있습니다.

운영 체제 제한이 있으므로 경로 길이 및 큐 관리자 이름의 합계가 최대 70자 길이가 되도록 제공된 경로의 길이를 짧게 유지하십시오.

Multi **ConvEBCDICNewline=NL_TO_LF|TABLE|ISO**

ASCII 코드 페이지에서 지원되지 않는 줄 바꾸기(NL) 문자가 EBCDIC 코드 페이지에 있습니다(ASCII의 일부 ISO 변형에는 대등한 문자가 포함됨). **`ConvEBCDICNewline`** 속성을 사용하여 IBM MQ 가 EBCDIC NL 문자를 ASCII 형식으로 변환하는 방법을 지정하십시오.

IBM i IBM MQ for IBM i에서 CCSID 1253은 ISO CCSID로 간주되고 NL_TO_LF는 ISO 및 ASCII 변환 모두에 영향을 줍니다.

z/OS **ConvEBCDICNewline** 속성은 z/OS에서 사용 불가능합니다. z/OS 의 동작은 `ConvEBCDICNewline=TABLE`와 같습니다. 다른 플랫폼의 기본값은 다를 수 있다는 점을 참고하십시오.

NL_TO_LF
EBCDIC를 모두 ASCII로 변환하려면 EBCDIC NL 문자(X'15')를 ASCII 줄 바꾸기 문자 LF(X'0A')로 변환하십시오.

NL_TO_LF가 기본값입니다.

TABLE
EBCDIC를 모두 ASCII로 변환하려면 사용자의 플랫폼에 사용되는 변환 테이블에 따라 EBCDIC NL 문자를 변환하십시오.

이 변환 유형의 결과는 플랫폼 및 언어에 따라 다르며 서로 다른 CCSID를 사용하는 경우 동일한 플랫폼에서도 다르게 작동합니다.

ISO
다음을 변환합니다.

- TABLE 메소드를 사용하여 ISO CCSID를 변환합니다.
- 기타 모든 CCSID는 NL_TO_CF 메소드를 사용하여 변환합니다.

가능한 ISO CCSID는 99 페이지의 표 11에 표시되어 있습니다.

표 *11.* 가능한 *ISO CCSID* 목록

CCSID	**코드 세트**
819	ISO8859-1
912	ISO8859-2
915	ISO8859-5
1089	ISO8859-6
813	ISO8859-7
916	ISO8859-8
920	ISO8859-9
1051	roman8

ASCII CCSID가 ISO 서브세트가 아닌 경우, **ConvEBCDICNewline** 기본값은 NL_TO_LF입니다.

V 9.0.0.6 IBM MQ 9.0.0 Fix Pack 6에서 **ConvEBCDICNewline** 스탠자 속성 대신 **AMQ_CONVEBCDICNEWLINE** 환경 변수 를 사용할 수 있습니다. 예를 들어, `mqs.ini` 파일을 사용할 수 없는 상황에서 클라이언트 측에 **ConvEBCDICNewline** 기능을 제공할 수 있습니다. 이 환경 변수는 **ConvEBCDICNewline** 속성과 동일한 값(`NL_TO_LF`, `TABLE` 또는 `ISO`)을 사용합니다. 해당 속성과 환경 변수가 모두 설정된 경우 스탠자 속성이 우선적으로 적용됩니다.

기본 큐 관리자

IBM MQ Explorer의 `General` IBM MQ 특성 페이지 또는 mqs.ini 파일의 `DefaultQueueManager` 스탠자를 사용하여 기본 큐 관리자를 지정하십시오.

Name=*default_queue_manager*
기본 큐 관리자는 큐 관리자 이름을 명시적으로 지정하지 않은 명령을 처리합니다. 새 기본 큐 관리자를 작성하면 `DefaultQueueManager` 속성이 자동으로 업데이트됩니다. 의도치 않게 기본 큐 관리자를 작성한 후 다시 원래의 기본 큐 관리자로 되돌리려는 경우 수동으로 `DefaultQueueManager` 속성을 대체하십시오.

엑시트 특성

Use the `Extended` IBM MQ properties page from the IBM MQ Explorer, or the `ExitProperties` stanza in the mqs.ini file to specify configuration options used by queue manager exit programs.

CLWLMode= SAFE |FAST

클러스터 워크로드(CLWL) 엑시트를 사용하면 클러스터에서 MQI 호출(예: `MQOPEN`, `MQPUT`)에 대한 응답으로 열리는 클러스터 큐를 지정할 수 있습니다. CLWL 엑시트는 CLWLMode 속성에 지정한 값에 따라 FAST 모드 또는 SAFE 모드로 실행됩니다. CLWLMode 속성을 생략하면 클러스터 워크로드 엑시트가 SAFE 모드로 실행됩니다.

안전

큐 관리자와는 별도의 프로세스로 CLWL 엑시트를 실행하십시오. 기본값입니다.

SAFE 모드로 실행할 때 사용자 작성 CLWL 엑시트에서 문제점이 발생하면 다음이 발생합니다.

- CLWL 서버 프로세스(amqzlwa0)가 실패합니다.
- 큐 관리자가 CLWL 서버 프로세스를 재시작합니다.
- 오류 로그에 오류가 보고됩니다. MQI 호출이 진행 중인 경우 리턴 코드 양식으로 된 알림이 수신됩니다.

큐 관리자의 무결성은 보존됩니다.

참고: 별도의 프로세스로 CLWL 엑시트를 실행하면 성능에 영향을 줄 수 있습니다.

FAST

큐 관리자 프로세스에서 클러스터 엑시트 인라인을 실행합니다.

이 옵션을 지정하면 SAFE 모드 실행과 연관된 프로세스 전환 비용이 발생하는 것을 방지하여 성능을 향상시키지만 큐 관리자 무결성은 손상됩니다. CLWL 엑시트에 문제점이 **없음**을 확신하고 성능에 대해 특히 걱정되는 경우에만 CLWL 엑시트를 FAST 모드로 실행해야 합니다.

CLWL 엑시트를 FAST 모드로 실행하는 동안 문제점이 발생하면 큐 관리자는 실패하고 큐 관리자의 무결성이 손상될 위험이 있습니다.

IBM MQ의 로그 기본값

Use the `Default log settings` IBM MQ properties page from the IBM MQ Explorer, or the `LogDefaults` stanza in the mqs.ini file to specify information about log defaults for all queue managers.

스탠자가 없는 경우 MQ 기본값이 사용됩니다. 큐 관리자를 작성할 때 기본값으로 로그 기본값이 사용되지만 `crtmqm` 명령에서 로그 속성을 지정하는 경우에는 대체될 수 있습니다. 이 명령의 자세한 내용은 **`crtmqm`** 을 참조하십시오.

큐 관리자가 작성되고 나면 해당 큐 관리자에 대한 로그 속성은 107 페이지의 『큐 관리자 로그』에서 설명된 설정에서 가져옵니다.

기본 접두부(98 페이지의 『모든 큐 관리자』에서 지정됨) 및 특정 큐 관리자에 대해 지정된 로그 경로(107 페이지의 『큐 관리자 로그』에서 지정됨)를 사용하면 큐 관리자 및 해당 로그가 다른 물리적 드라이브에 있을 수 있습니다. 이는 기본적으로 같은 드라이브에 있더라도 권장되는 메소드입니다.

로그 크기 계산에 대한 정보는 483 페이지의 『로그의 크기 계산』의 내용을 참조하십시오.

참고: 다음 매개변수에서 지정되는 한계는 IBM MQ에서 설정되는 한계입니다. 운영 체제 한계는 가능한 최대 로그 크기를 감소시킬 수 있습니다.

LogPrimaryFiles= 3 | *2-254*(Windows)| *2-510*(UNIX and Linux 시스템)

로그 파일은 큐 관리자가 작성될 때 할당됩니다.

보유할 수 있는 최소 기본 로그 파일 수는 2개이며 최대 기본 로그 파일 수는 Windows의 경우 254개이고 UNIX 및 Linux의 경우 510개입니다. 기본은 3입니다.

기본 및 보조 로그 파일의 합계는 Windows에서는 255개를 초과해서는 안되고 UNIX 및 Linux에서는 511개를 초과해서는 안되며 3개 이상이어야 합니다.

큐 관리자가 작성되거나 시작될 때 값을 검토합니다. 큐 관리자가 작성된 후에는 이 값을 변경할 수 있습니다. 그러나 변경된 값은 큐 관리자가 재시작될 때까지 적용되지 않으며 즉시 적용되지 않을 수도 있습니다.

LogSecondaryFiles= 2|*1-253* (Windows)|*1-509* (UNIX 및 Linux)

로그 파일은 1차 파일이 모두 사용될 때 할당됩니다.

최소 보조 로그 파일 수는 1개이며 최대 보조 로그 파일 수는 Windows의 경우 253개이고 UNIX 및 Linux의 경우 509개입니다. 기본 수는 2입니다.

기본 및 보조 로그 파일의 합계는 Windows에서는 255개를 초과해서는 안되고 UNIX 및 Linux에서는 511개를 초과해서는 안되며 3개 이상이어야 합니다.

이 값은 큐 관리자가 시작될 때 조사됩니다. 이 값을 변경할 수 있지만 변경사항은 큐 관리자가 재시작될 때까지 적용되지 않으며 즉시 적용되지 않을 수도 있습니다.

LogFilePages=*number*

로그 데이터는 로그 파일이라는 일련의 파일에 보유됩니다. 로그 파일 크기는 4KB 페이지 단위로 지정됩니다.

로그 파일 기본 페이지 수는 4096이고 로그 파일 크기 16MB를 지정합니다.

UNIX 및 Linux에서 최소 로그 파일 페이지 수는 64개이고 Windows에서 최소 로그 파일 페이지 수는 32개이며 두 경우 모두에서 최대 수는 65,535개입니다.

참고: 큐 관리자 작성 중 지정된 로그 파일 크기는 큐 관리자에 대해 변경할 수 없습니다.

LogType= `CIRCULAR` |`LINEAR`

사용할 로그의 유형입니다. 기본값은 CIRCULAR입니다.

CIRCULAR

로그를 사용한 재시작 복구를 시작하여 시스템이 중지되었을 때 진행 중이었던 트랜잭션을 롤백합니다.

순환 로깅에 대한 자세한 설명은 478 페이지의 『로깅 유형』의 내용을 참조하십시오.

LINEAR

재시작 복구 및 매체나 정방향 복구(로그의 컨텐츠를 재실행하여 손실되거나 손상된 데이터 작성)에 사용됩니다.

선형 로깅에 대한 자세한 설명은 478 페이지의 『로깅 유형』의 내용을 참조하십시오.

기본값을 변경하려는 경우 `LogType` 속성을 편집하거나 `crtmqm` 명령을 사용하여 선형 로깅을 지정할 수 있습니다.

V 9.0.4 IBM MQ 9.0.4부터는 큐 관리자가 작성된 후에 로깅 메소드를 변경할 수 없습니다. 자세한 정보는 migmqlog를 참조하십시오.

LogBufferPages=0|*0-4096*

버퍼의 크기를 4KB 페이지 단위로 작성하고 지정하기 위해 버퍼 레코드에 할당되는 메모리의 양입니다.

최소 버퍼 페이지 수는 18이고 최대는 4096입니다. 버퍼가 클수록 처리량이 늘어나며 특히 메시지가 큰 경우가 이에 해당됩니다.

0(기본값)을 지정하면 큐 관리자가 크기를 선택합니다. IBM WebSphere MQ 7.1에서 이는 512(2048KB)입니다.

1 - 17 범위의 숫자를 지정하는 경우 큐 관리자는 18(72KB)로 기본값 설정됩니다. 18 - 4096 범위의 숫자를 지정하는 경우 큐 관리자는 지정된 숫자를 사용하여 할당되는 메모리를 설정합니다.

LogDefaultPath=*directory_name*

큐 관리자의 로그 파일이 상주하는 디렉토리입니다. 이 디렉토리는 가급적이면 메시지 큐와는 다른 드라이브에서 큐 관리자가 기록할 수 있는 로컬 디바이스에 상주합니다. 다른 드라이브를 지정하면 시스템이 실패할 경우에 추가 보호를 제공합니다.

기본값은 다음과 같습니다.

- *`DefaultPrefix`*`\log` for IBM MQ for Windows where *`DefaultPrefix`* is the value specified on the `DefaultPrefix` attribute on the `All Queue Managers` IBM MQ properties page. 이 값은 설치 시 설정됩니다.

- IBM MQ for UNIX and Linux 시스템의 경우 `/var/mqm/log`

또는 -ld 플래그를 사용하여 `crtmqm` 명령에 디렉토리의 이름을 지정할 수 있습니다. 큐 관리자가 작성될 때 디렉토리도 큐 관리자 디렉토리 아래에서 작성되며 이 디렉토리는 로그 파일을 보유하는 데 사용됩니다. 이 디렉토리의 이름은 큐 관리자 이름을 기반으로 합니다. 이는 로그 파일 경로가 고유하도록 보장하고 디렉토리 이름 길이에 대한 제한을 준수하도록 합니다.

`crtmqm` 명령에 -ld를 지정하지 않으면 mqs.ini 파일의 `LogDefaultPath` 속성 값이 사용됩니다.

다중 큐 관리자가 서로 다른 로그 디렉토리를 사용하도록 큐 관리자 이름이 디렉토리 이름에 추가됩니다.

큐 관리자가 작성될 때 구성 정보의 로그 속성에서 `LogPath` 값이 작성되어 큐 관리자의 로그에 대한 전체 디렉토리 이름을 제공합니다. 이 값은 큐 관리자가 시작되거나 삭제될 때 로그를 찾는 데 사용됩니다.

LogWriteIntegrity=SingleWrite|DoubleWrite|`TripleWrite`
로거가 신뢰할 수 있는 로그 레코드를 쓰기 위해 사용하는 메소드입니다.

`TripleWrite`
이는 기본 메소드입니다.

DoubleWrite를 선택할 수 있지만, 선택하면 시스템이 이를 **TripleWrite**로 해석한다는 점에 유의하십시오.

SingleWrite
IBM MQ 복구 로그를 호스팅하는 파일 시스템 및 디바이스가 명시적으로 4KB 쓰기의 원자성을 보장하는 경우에만 **SingleWrite**를 사용해야 합니다.

즉, 어떤 이유로 4KB 페이지를 쓰는 데 실패하는 경우에는 두 가지 상태(사전 이미지 또는 사후 이미지)만 가능합니다. 중간 상태는 발생하지 않아야 합니다.

참고: 지속 워크로드에 충분한 동시성이 있을 경우 기본값 **TripleWrite** 외의 값을 설정하는 데 있어 최소한의 이점이 있을 수 있습니다.

고급 구성 및 전원 인터페이스(ACPI)

IBM MQ Explorer의 ACPI IBM MQ 특성 페이지를 사용하여 시스템이 일시중단 요청을 수신할 때 IBM MQ 가 작동하는 방식을 지정하십시오.

Windows에서는 ACPI(Advanced Configuration and Power Interface) 표준을 지원합니다. 이를 통해 ACPI가 사용으로 설정된 하드웨어가 있는 Windows 사용자가 시스템이 일시중단 모드에 진입하고 다시 재개될 때 채널을 중지하고 재시작할 수 있습니다.

ACPI IBM MQ 특성 페이지에서 지정된 설정은 경보 모니터가 실행 중일 때에만 적용됨에 유의하십시오. 경보 모니터 아이콘은 경보 모니터가 실행 중인 경우 작업 표시줄에 표시됩니다.

DoDialog= Y | N
일시중단 요청 시 대화 상자에 표시됩니다.

DenySuspend=Y | N
일시중단 요청을 거부합니다. DoDialog=N이거나 DoDialog=Y인 경우에 사용되며 노트북 덮개가 닫힘 등으로 인해 대화 상자를 표시할 수 없습니다.

CheckChannelsRunning=Y | N
채널이 실행 중인지 여부를 검사합니다. 이 결과는 다른 설정의 결과를 판별할 수 있습니다.

다음 표에는 이러한 매개변수의 각 결합 효과가 요약되어 있습니다.

DoDialog	DenySuspend	CheckChannels 실행	Action
N	N	N	일시중단 요청을 승인합니다.
N	N	Y	일시중단 요청을 승인합니다.
N	Y	N	일시중단 요청을 거부합니다.

N	Y	Y	실행 중인 채널이 있는 경우 일시중단 요청을 거부하고 그렇지 않은 경우 요청을 승인합니다.
Y	N	N	대화 상자를 표시합니다(참고 ; 일시중단 요청 승인 참조). 기본값입니다.
Y	N	Y	실행 중인 채널이 없는 경우 일시중단 요청을 승인하고 있을 경우 대화 상자를 표시합니다(참고; 요청 승인 참조).
Y	Y	N	대화 상자를 표시합니다(참고 ; 일시중단 요청 거부).
Y	Y	Y	실행 중인 채널이 없는 경우 일시중단 요청을 승인하고 있을 경우에는 대화 상자를 표시합니다(참고 ; 요청 거부).

참고: 조치가 대화 상자를 표시하는 것인 경우에, 대화 상자를 표시할 수 없으면(예를 들어, 노트북 덮개가 닫힘으로 인해) DenySuspend 옵션이 일시중단 요청이 승인되는지 또는 거부되는지 판별하는 데 사용됩니다.

API 엑시트

IBM MQ Explorer 또는 `amqmdain` 명령을 사용하여 API 엑시트에 대한 입력 항목을 변경합니다.

Use the `Exits` IBM MQ properties page from the IBM MQ Explorer, or the `ApiExitTemplate` and `ApiExitCommon` stanza in the mqs.ini file to identify API exit routines for all queue managers. Windows 시스템에서 `amqmdain` 명령을 사용하여 API 엑시트에 대한 입력 항목을 변경할 수도 있습니다. (개별 큐 관리자에 대한 API 엑시트를 식별하려면 82 페이지의 『큐 관리자 구성 파일, qm.ini』에 설명된 대로 `ApiExitLocal` 스탠자를 사용합니다.)

이러한 스탠자의 속성에 대한 전체 설명은 API 엑시트 구성을 참조하십시오.

큐 관리자

큐 관리자마다 한 개의 `QueueManager` 스탠자가 있습니다. 스탠자를 사용하여 큐 관리자 디렉토리의 위치를 지정합니다.

Windows, UNIX and Linux 시스템에는 큐 관리자마다 한 개의 `QueueManager` 스탠자가 있습니다. 이러한 속성은 해당 큐 관리자와 연관된 파일이 들어 있는 디렉토리의 이름 및 큐 관리자 이름을 지정합니다. 이 디렉토리의 이름은 큐 관리자 이름을 기반으로 하지만 큐 관리자 이름이 유효한 파일 이름이 아닌 경우 변환됩니다. 이름 변환에 대한 자세한 정보는 IBM MQ 파일 이름에 대한 이해를 참조하십시오.

Name=*queue_manager_name*
: 대기열 관리자의 이름입니다.

Prefix=*prefix*
: 큐 관리자 파일이 저장되는 위치입니다. 기본적으로 이 값은 모든 큐 관리자 정보의 DefaultPrefix 속성에서 지정된 값과 같습니다.

Directory=*name*
: 큐 관리자 파일이 저장되는 `*prefix*\QMGRS` 디렉토리 아래에 있는 서브디렉토리의 이름입니다. 이 이름은 큐 관리자 이름을 기반으로 하지만 중복 이름이 있거나 큐 관리자 이름이 유효한 파일 이름이 아닌 경우 변환될 수 있습니다.

DataPath=*path*
: 큐 관리자가 작성될 때 제공된 명시적 데이터 경로이며 큐 관리자 데이터에 대한 경로로서 접두부 및 디렉토리를 대체합니다.

InstallationName=*name*
: 이 큐 관리자와 연관된 IBM MQ 설치의 이름입니다. 이 설치의 명령은 이 큐 관리자와 상호작용할 때 사용되어야 합니다. InstallationName 값이 표시되지 않으면 큐 관리자는 IBM WebSphere MQ 7.1 이전의 설치 제품과 연관됩니다.

관련 태스크

359 페이지의 『큐 관리자를 설치와 연관』
큐 관리자를 작성하면 **crtmqm** 명령을 실행하는 설치와 자동으로 연관됩니다. UNIX, Linux, and Windows에서 **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 변경할 수 있습니다.

큐 관리자 구성 정보 변경

개별 큐 관리자의 구성을 수정하는 데 사용할 수 있는 속성은 IBM MQ의 모든 속성을 대체합니다.

이 태스크 정보

UNIX and Linux 시스템에서 `qm.ini` 구성 파일을 편집하여 큐 관리자 구성 정보를 수정하십시오. `qm.ini`에서 스탠자를 정의할 때 각 항목을 새 행에서 시작하지 않아도 됩니다 세미콜론(;)이나 해시 문자(#)를 사용해서 주석을 나타낼 수 있습니다.

Windows 및 Linux x86-64 시스템에서는 IBM MQ Explorer를 사용하여 일부 구성 정보를 수정할 수 있습니다. 그러나 설치 가능 서비스와 해당 컴포넌트를 변경하는 것은 중대한 의미를 가지므로 설치 가능 서비스는 IBM MQ Explorer에서 읽기 전용입니다. 따라서 설치 가능 서비스에 대한 변경은 Windows의 경우 **regedit**를 사용해야 하고 UNIX and Linux에서는 `qm.ini` 파일을 편집하여 수행해야 합니다.

프로시저

- 큐 관리자 구성 정보 변경에 대한 자세한 정보는 다음 하위 주제를 참조하십시오.

관련 개념

5 페이지의 『IBM MQ 구성』
하나 이상의 컴퓨터에서 하나 이상의 큐 관리자를 작성하고 개발, 테스트, 프로덕션 시스템에서 해당 큐 관리자가 비즈니스 데이터를 포함하는 메시지를 처리하도록 구성합니다.

관련 태스크

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

관련 참조

98 페이지의 『IBM MQ 구성 정보를 변경하기 위한 속성』
IBM MQ for Windows 시스템 및 IBM MQ for Linux (x86 및 x86-64 플랫폼) 시스템에서 IBM MQ Explorer를 사용하여 구성 정보를 수정하십시오. 기타 시스템에서는 mqs.ini 구성 파일을 편집하여 이 정보를 수정합니다.

관련 정보

계획 중

IBM MQ 관리

Windows 액세스 모드

Access Mode는 Windows Server에만 적용됩니다. AccessMode 스탠자는 **crtmqm** 명령의 `-a [r]` 옵션에 의해 설정됩니다. 큐 관리자가 작성된 후에는 AccessMode 스탠자를 변경하지 마십시오.

crtmqm 명령의 액세스 그룹(`-a [r]`) 옵션을 사용하여 멤버에 모든 큐 관리자 데이터 파일에 대한 전체 액세스가 부여되는 Windows 보안 그룹을 지정하십시오. 그룹은 사용하는 구문에 따라서 로컬 또는 글로벌 그룹일 수 있습니다. 그룹 이름의 올바른 구문은 다음과 같습니다.

LocalGroup
Domain name\GlobalGroup name
GlobalGroup name @ Domain name

`-a [r]` 옵션과 함께 crtmqm 명령을 실행하기 전에 추가 액세스 그룹을 정의해야 합니다.

`-a` 대신 `-ar`을 사용하여 그룹을 지정하면 로컬 mqm 그룹에 큐 관리자 데이터 파일에 대한 액세스가 부여되지 않습니다. 큐 관리자 데이터 파일을 호스팅하는 파일 시스템이 로컬로 정의된 그룹에 대한 액세스 제어 입력 항목을 지원하지 않는 경우 이 옵션을 사용하십시오.

이 그룹은 공유 큐 관리자 데이터 및 로그 폴더에 대한 액세스 권한과 함께 다중 인스턴스 큐 관리자를 제공하는 데 사용되는 일반적인 글로벌 보안 그룹입니다. 폴더에 대해 읽기 및 쓰기 권한을 설정하거나 큐 관리자 데이터 및 로그 파일이 포함된 폴더를 공유하려면 추가 보안 액세스 그룹을 사용하십시오.

추가 보안 액세스 그룹은 mqm이라는 로컬 그룹을 사용하여 큐 관리자 데이터 및 로그가 포함된 폴더에 대한 권한을 설정하는 또 다른 방법입니다. 로컬 그룹 mqm과 달리, 추가 보안 액세스 그룹은 로컬 또는 글로벌 그룹으로 설정할 수 있습니다. 다중 인스턴스 큐 관리자에 사용되는 데이터 및 로그 파일이 포함된 공유 폴더에 대한 권한을 설정하려면 글로벌 그룹이어야 합니다.

Windows 운영 체제는 큐 관리자 데이터 및 로그 파일을 읽고 쓰기 위한 액세스 권한을 확인합니다. 큐 관리자 프로세스를 실행하는 사용자 ID의 권한을 검사합니다. 확인되는 사용자 ID는 큐 관리자를 서비스로 시작했는지, 아니면 대화식으로 시작했는지에 따라 달라집니다. 큐 관리자를 서비스로 시작한 경우, Windows 시스템에서 확인한 사용자 ID는 IBM MQ **준비** 마법사를 사용하여 구성한 사용자 ID입니다. 큐 관리자를 대화식으로 시작한 경우, **strmqm** 명령을 실행한 사용자 ID를 Windows 시스템에서 검사합니다.

큐 관리자를 시작하려면 사용자 ID가 로컬 mqm 그룹의 멤버여야 합니다. 사용자 ID가 추가 보안 액세스 그룹의 멤버일 경우 큐 관리자가 해당 그룹을 사용하여 권한이 부여된 파일을 읽고 쓸 수 있습니다.

제한사항: Windows 운영 체제에서만 추가적인 보안 액세스 그룹을 지정할 수 있습니다. 다른 운영 체제에서 추가적인 보안 액세스 그룹을 지정하는 경우 **crtmqm** 명령이 오류를 리턴합니다.

관련 개념

427 페이지의 『Windows에서 비공유 큐 관리자 데이터와 로그 디렉토리 및 파일 보안 설정』
424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』

관련 태스크

401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』

관련 정보

crtmqm

ULW 설치 가능 서비스 구성

Windows에서는 **regedit**를 사용하고 UNIX and Linux에서는 `qm.ini` 파일의 `Service` 스탠자를 사용하여 설치 가능 서비스를 변경합니다.

참고: 설치 가능 서비스 및 해당 컴포넌트를 변경하는 것은 중요한 의미가 있습니다. 이러한 이유로 설치 가능 서비스는 IBM MQ Explorer에서 읽기 전용입니다.

Windows 시스템에서 설치 가능 서비스를 변경하려면 **regedit**를 사용하고, UNIX and Linux 시스템에서는 `qm.ini` 파일의 `Service` 스탠자를 사용하십시오. 서비스 내 각 컴포넌트에 대해 해당 컴포넌트의 코드를 포함하는 모듈의 이름 및 경로도 지정해야 합니다. UNIX and Linux 시스템에서 이에 대한 `ServiceComponent` 스탠자를 사용하십시오.

Name= AuthorizationService |NameService
필요한 서비스의 이름입니다.

AuthorizationService
IBM MQ에서 권한 서비스 컴포넌트를 오브젝트 권한 관리자 또는 OAM이라고 합니다. 큐 관리자가 작성될 때 `AuthorizationService` 스탠자 및 그에 연관된 `ServiceComponent` 스탠자가 자동으로 추가됩니다. 다른 `ServiceComponent` 스탠자는 수동으로 추가하십시오.

NameService
기본적으로 이름 서비스는 제공되지 않습니다. 이름 서비스가 필요한 경우 `NameService` 스탠자를 수동으로 추가해야 합니다.

EntryPoints=*number-of-entries*
서비스에 대해 정의된 시작점의 수입니다.

여기에는 초기화 및 종료 시작점이 포함됩니다.

Windows **SecurityPolicy= Default|NTSIDsRequired**
Windows 시스템에서 SecurityPolicy 속성은 지정된 서비스가 기본 권한 서비스 즉, OAM인 경우에만 적용됩니다. SecurityPolicy 속성을 사용하면 각 큐 관리자에 대한 보안 정책을 지정할 수 있습니다.

가능한 값은 다음과 같습니다.

기본값
: 적용할 기본 보안 정책을 사용합니다. Windows 보안 ID(NT SID)가 특정 사용자 ID의 OAM에 전달되지 않으면, 관련 보안 데이터베이스를 검색하여 적절한 SID를 확보하려고 시도합니다.

NTSIDsRequired
: 보안 검사를 수행할 때 NT SID를 OAM에 전달합니다.

  자세한 정보는 Windows 보안 ID(SID)를 참조하십시오.

권한 서비스 스탠자 구성: Windows 시스템도 참조하십시오.

Linux UNIX **SecurityPolicy=user|group|default**
UNIX and Linux 시스템에서 이 값은 큐 관리자가 사용자 기반 권한 부여를 사용하는지 또는 그룹 기반 권한 부여를 사용하는지 지정합니다. 값은 대소문자를 구분하지 않습니다.

이 속성을 포함시키지 않으면 그룹 기반 권한 부여를 사용하는 `default`가 사용됩니다. 변경사항을 적용하려면 큐 관리자를 재시작하십시오. 권한 서비스 스탠자 구성: UNIX and Linux 시스템도 참조하십시오.

SharedBindingsUserId=*user-type*
SharedBindingsUserId 속성은 지정된 서비스가 기본 권한 서비스 즉, OAM인 경우에만 적용됩니다. SharedBindingsUserId 속성은 공유 바인딩과 관련해서만 사용됩니다. 이 값을 사용하면 MQZ_AUTHENTICATE_USER 함수에서 *IdentityContext* 구조의 *UserIdentifier* 필드가 유효한 사용자 ID이거나 실제 사용자 ID입니다.

MQZ_AUTHENTICATE_USER 함수에 대한 정보는 MQZ_AUTHENTICATE_USER - 사용자 인증을 참조하십시오.

가능한 값은 다음과 같습니다.

기본값
: *UserIdentifier* 필드의 값이 실제 사용자 ID로 설정됩니다.

실제
: *UserIdentifier* 필드의 값이 실제 사용자 ID로 설정됩니다.

Effective
: *UserIdentifier* 필드의 값이 유효 사용자 ID로 설정됩니다.

FastpathBindingsUserId=*user-type*
FastpathBindingsUserId 속성은 지정된 서비스가 기본 권한 서비스 즉, OAM인 경우에만 적용됩니다. FastpathBindingsUserId 속성은 빠른 경로 바인딩과 관련해서만 사용됩니다. 이 값을 사용하면 MQZ_AUTHENTICATE_USER 함수에서 *IdentityContext* 구조의 *UserIdentifier* 필드가 유효한 사용자 ID이거나 실제 사용자 ID입니다.

MQZ_AUTHENTICATE_USER 함수에 대한 정보는 MQZ_AUTHENTICATE_USER - 사용자 인증을 참조하십시오.

가능한 값은 다음과 같습니다.

기본값
: *UserIdentifier* 필드의 값이 실제 사용자 ID로 설정됩니다.

실제
: *UserIdentifier* 필드의 값이 실제 사용자 ID로 설정됩니다.

Effective
: *UserIdentifier* 필드의 값이 유효 사용자 ID로 설정됩니다.

IsolatedBindingsUserId= *user-type*
`IsolatedBindingsUserId` 속성은 지정된 서비스가 기본 권한 서비스 즉, OAM인 경우에만 적용됩니다. **`IsolatedBindingsUserId`** 속성은 격리 바인딩과 관련해서만 사용됩니다. 이 값을 사용하면 MQZ_AUTHENTICATE_USER 함수에서 *IdentityContext* 구조의 *UserIdentifier* 필드가 유효한 사용자 ID이거나 실제 사용자 ID입니다.

MQZ_AUTHENTICATE_USER 함수에 대한 정보는 MQZ_AUTHENTICATE_USER - 사용자 인증을 참조하십시오.

가능한 값은 다음과 같습니다.

기본값
UserIdentifier 필드의 값이 유효 사용자 ID로 설정됩니다.

실제
UserIdentifier 필드의 값이 실제 사용자 ID로 설정됩니다.

Effective
UserIdentifier 필드의 값이 유효 사용자 ID로 설정됩니다.

설치 가능 서비스 및 컴포넌트에 대한 자세한 정보는 UNIX, Linux, and Windows의 설치 가능 서비스 및 컴포넌트를 참조하십시오.

일반적으로 보안 서비스에 대한 자세한 정보는 UNIX and Linux 시스템에서 보안 설정을 참조하십시오.

관련 정보
설치 가능 서비스 참조 정보

ULW 서비스 컴포넌트

새 설치 가능 서비스를 설치할 때 서비스 컴포넌트 정보를 지정해야 합니다. Windows 시스템에서는 **`regedit`**를 사용하고 UNIX and Linux 시스템에서는 qm.ini 파일의 **`ServiceComponent`** 스탠자를 사용하십시오. 권한 서비스 스탠자는 기본적으로 표시되고 연관된 컴포넌트 OAM은 활성입니다.

다음과 같이 서비스 컴포넌트를 지정하십시오.

Service=*service_name*
필요한 서비스의 이름입니다. 이는 서비스 구성 정보의 `Name` 속성에서 지정된 값과 일치해야 합니다.

Name=*component_name*
서비스 컴포넌트의 설명 이름입니다. 이는 고유해야 하고 IBM MQ 오브젝트의 이름에 유효한 문자만 포함해야 합니다(예: 큐 이름). 이 이름은 서비스에서 생성되는 운영자 메시지에 나타납니다. 이 이름은 회사 상표 또는 이와 유사하게 구별되는 문자열로 시작하는 것이 좋습니다.

Module=*module_name*
이 컴포넌트에 대한 코드가 포함된 모듈의 이름입니다. 이 이름은 전체 경로 이름이어야 합니다.

ComponentDataSize=*size*
각 호출 시 컴포넌트에 전달된 컴포넌트 데이터 영역의 크기(바이트)입니다. 컴포넌트 데이터가 필요하지 않으면 0을 지정하십시오.

설치 가능 서비스 및 컴포넌트에 대한 자세한 정보는 UNIX, Linux 및 Windows용 설치 가능 서비스 및 컴포넌트를 참조하십시오.

ULW 큐 관리자 로그

Log 의 IBM MQ Explorer큐 관리자 특성 페이지 또는 qm.ini 파일의 Log 스탠자를 사용하여 큐 관리자에 대한 로깅에 대한 정보를 지정하십시오.

기본적으로 이러한 설정은 큐 관리자에 대한 기본 로그 설정에 대해 지정된 설정에서 상속됩니다(100 페이지의 『IBM MQ의 로그 기본값』에 설명됨). 이 큐 관리자를 다른 방식으로 구성하려는 경우에만 이러한 설정을 변경하십시오.

로그 크기 계산에 대한 정보는 483 페이지의 『로그의 크기 계산』의 내용을 참조하십시오.

참고: 다음 매개변수에서 지정되는 한계는 IBM MQ에서 설정됩니다. 운영 체제 한계는 가능한 최대 로그 크기를 감소시킬 수 있습니다.

LogPrimaryFiles= 3 | *2-254*(Windows)| *2-510*(UNIX and Linux 시스템)
로그 파일은 큐 관리자가 작성될 때 할당됩니다.

사용할 수 있는 최소 기본 로그 파일 수는 2이며 최대 기본 로그 파일 수는 Windows의 경우 254이고 UNIX and Linux 시스템의 경우 510입니다. 기본은 3입니다.

기본 및 보조 로그 파일의 합계는 Windows에서는 255를 초과해서는 안되고 UNIX and Linux 시스템에서는 511을 초과해서는 안되며 3 이상이어야 합니다.

큐 관리자가 작성되거나 시작될 때 값을 검토합니다. 큐 관리자가 작성된 후에는 이 값을 변경할 수 있습니다. 그러나 변경된 값은 큐 관리자가 재시작될 때까지 적용되지 않으며 즉시 적용되지 않을 수도 있습니다.

LogSecondaryFiles= 2 | *1-253*(Windows)| *1-509*(UNIX and Linux 시스템)

로그 파일은 1차 파일이 모두 사용될 때 할당됩니다.

최소 보조 로그 파일 수는 1이며 최대 보조 로그 파일 수는 Windows의 경우 253이고 UNIX and Linux 시스템의 경우 509입니다. 기본 수는 2입니다.

기본 및 보조 로그 파일의 합계는 Windows에서는 255를 초과해서는 안되고 UNIX and Linux 시스템에서는 511을 초과해서는 안되며 3 이상이어야 합니다.

이 값은 큐 관리자가 시작될 때 조사됩니다. 이 값을 변경할 수 있지만 변경사항은 큐 관리자가 재시작될 때까지 적용되지 않으며 즉시 적용되지 않을 수도 있습니다.

LogFilePages=*number*

로그 데이터는 로그 파일이라는 일련의 파일에 보유됩니다. 로그 파일 크기는 4KB 페이지 단위로 지정됩니다.

로그 파일 기본 페이지 수는 4096이고 로그 파일 크기 16MB를 지정합니다.

UNIX and Linux 시스템에서 최소 로그 파일 페이지 수는 64이고 Windows에서 최소 로그 파일 페이지 수는 32이며 두 경우 모두에서 최대 수는 65 535입니다.

참고: 큐 관리자 작성 중 지정된 로그 파일 크기는 큐 관리자에 대해 변경할 수 없습니다.

LogType= CIRCULAR |LINEAR

큐 관리자가 사용할 로깅 유형입니다. 큐 관리자가 작성되고 나면 사용할 로깅 유형을 변경할 수 없습니다. 필요한 로깅 유형으로 큐 관리자를 작성하는 것에 대한 정보는 100 페이지의 『IBM MQ의 로그 기본값』의 LogType 속성에 대한 설명을 참조하십시오.

CIRCULAR

로그를 사용한 재시작 복구를 시작하여 시스템이 중지되었을 때 진행 중이었던 트랜잭션을 롤백합니다.

순환 로깅에 대한 자세한 설명은 478 페이지의 『로깅 유형』의 내용을 참조하십시오.

LINEAR

재시작 복구 및 매체나 정방향 복구(로그의 컨텐츠를 재실행하여 손실되거나 손상된 데이터 작성)에 사용됩니다.

선형 로깅에 대한 자세한 설명은 478 페이지의 『로깅 유형』의 내용을 참조하십시오.

LogBufferPages=0|*0-4096*

버퍼의 크기를 4KB 페이지 단위로 작성하고 지정하기 위해 버퍼 레코드에 할당되는 메모리의 양입니다.

최소 버퍼 페이지 수는 18이고 최대는 4096입니다. 버퍼가 클수록 처리량이 늘어나며 특히 메시지가 큰 경우가 이에 해당됩니다.

0(기본값)을 지정하면 큐 관리자가 크기를 선택합니다. IBM WebSphere MQ 7.1에서 이는 512(2048KB)입니다.

1 - 17 범위의 숫자를 지정하는 경우 큐 관리자는 18(72KB)로 기본값 설정됩니다. 18 - 4096 범위의 숫자를 지정하는 경우 큐 관리자는 지정된 숫자를 사용하여 할당되는 메모리를 설정합니다.

이 값은 큐 관리자가 시작될 때 조사됩니다. 이 값은 명시된 한계 내에서 증가되거나 감소될 수 있습니다. 그러나 값의 변경은 다음에 큐 관리자가 시작될 때까지 적용되지 않습니다.

LogPath=*directory_name*

큐 관리자의 로그 파일이 상주하는 디렉토리입니다. 이 디렉토리는 가급적이면 메시지 큐와는 다른 드라이브에서 큐 관리자가 기록할 수 있는 로컬 디바이스에 존재해야 합니다. 다른 드라이브를 지정하면 시스템이 실패할 경우에 추가 보호를 제공합니다.

기본값은 다음과 같습니다.

- IBM MQ for Windows의 `C:\ProgramData\IBM\MQ\log` .
- `/var/mqm/log` in IBM MQ for UNIX and Linux systems.

-ld 플래그를 사용하여 crtmqm 명령에 디렉토리의 이름을 지정할 수 있습니다. 큐 관리자가 작성될 때 디렉토리도 큐 관리자 디렉토리 아래에서 작성되며 이 디렉토리는 로그 파일을 보유하는 데 사용됩니다. 이 디렉토리의 이름은 큐 관리자 이름을 기반으로 합니다. 이는 로그 파일 경로가 고유하도록 보장하고 디렉토리 이름 길이에 대한 제한을 준수하도록 합니다.

crtmqm 명령에 -ld를 지정하지 않으면 LogDefaultPath 속성의 값이 사용됩니다.

IBM MQ for UNIX 및 Linux 시스템에서 사용자 ID mqm 및 그룹 mqm에는 로그 파일에 대한 전체 권한이 있어야 합니다. 이들 파일의 위치를 변경할 경우, 해당 권한을 직접 부여해야 합니다. 이는 로그 파일이 제품에서 제공된 기본 위치에 있는 경우 필요하지 않습니다.

LogWriteIntegrity=SingleWrite|DoubleWrite|TripleWrite
로거가 신뢰할 수 있는 로그 레코드를 쓰기 위해 사용하는 메소드입니다.

TripleWrite
이는 기본 메소드입니다.

DoubleWrite를 선택할 수 있지만, 선택하면 시스템이 이를 **TripleWrite**로 해석한다는 점에 유의하십시오.

SingleWrite
IBM MQ 복구 로그를 호스팅하는 파일 시스템 및 디바이스가 명시적으로 4KB 쓰기의 원자성을 보장하는 경우에만 **SingleWrite**를 사용해야 합니다.

즉, 어떤 이유로 4KB 페이지를 쓰는 데 실패하는 경우에는 두 가지 상태(사전 이미지 또는 사후 이미지)만 가능합니다. 중간 상태는 발생하지 않아야 합니다.

참고: 지속 워크로드에 충분한 동시성이 있을 경우 기본값 **TripleWrite** 외의 값을 설정하는 데 있어 최소한의 이점이 있을 수 있습니다.

V 9.0.2 **LogManagement= Manual |Automatic | Archive**
큐 관리자에 의해 또는 수동으로 로그 익스텐트를 관리하는 데 사용되는 메소드입니다.

이 속성은 **LogType**이 LINEAR인 경우에만 적용됩니다.

LogManagement 값을 변경하는 경우 변경사항은 큐 관리자가 재시작될 때까지 적용되지 않습니다.

속성에 대해 인식되지 않는 값이 있는 경우 큐 관리자는 값이 정정될 때까지 시작되지 않습니다.

수동
로그 익스텐트를 수동으로 관리합니다. 이 옵션을 지정하는 것은 로그 익스텐트가 복구를 위해 더 이상 필요하지 않은 경우에도 큐 관리자가 로그 익스텐트를 재사용하거나 삭제하지 않는다는 것을 의미합니다.

자동
큐 관리자가 자동으로 로그 익스텐트를 관리합니다. 이 옵션을 지정하는 것은 복구를 위해 더 이상 로그 익스텐트가 필요하지 않게 되는 즉시 큐 관리자가 로그 익스텐트를 재사용하거나 삭제할 수 있음을 의미합니다. 아카이브를 위한 허용량은 작성되지 않습니다.

아카이브
큐 관리자가 로그 익스텐트를 관리하지만 각 로그 익스텐트의 아카이브가 완료될 때 사용자가 큐 관리자에 알려야 합니다.

이 옵션을 지정하는 것은 복구를 위해 더 이상 필요하지 않은 로그 익스텐트가 아카이브될 때 큐 관리자에게 알리는 즉시 큐 관리자는 자유롭게 로그 익스텐트를 재사용하거나 삭제할 수 있음을 의미합니다.

RESET QMGR MQSC 명령 또는 큐 관리자 재설정 PCF 명령을 사용하여 이 알림을 수행합니다.

Linux UNIX 제한 모드

이 옵션은 UNIX and Linux 시스템에만 적용됩니다. RestrictedMode 스탠자는 **crtmqm** 명령에 **-g** 옵션을 사용하여 설정됩니다. 큐 관리자가 작성된 후에는 이 스탠자를 변경하지 마십시오. **-g** 옵션을 사용하지 않으면 qm.ini 파일에 스탠자가 작성되지 않습니다.

IBM MQ 애플리케이션이 큐 관리자 데이터 디렉토리 내에서 큐 관리자에 연결되는 파일을 작성하는 몇 개의 디렉토리가 있습니다. 애플리케이션이 이들 디렉토리에 파일을 작성하기 위해서 세계 쓰기 액세스 권한이 부여됩니다.

- `/var/mqm/sockets/`*`QMgrName`*`/@ipcc/ssem/`*`hostname`*`/`
- `/var/mqm/sockets/`*`QMgrName`*`/@app/ssem/`*`hostname`*`/`
- `/var/mqm/sockets/`*`QMgrName`*`/zsocketapp/`*`hostname`*`/`

여기서 *QMGRNAME*은 큐 관리자의 이름이고 *hostname*은 호스트 이름입니다.

일부 시스템에서는 모든 사용자에게 이들 디렉토리에 대한 쓰기 액세스 권한을 부여하는 것을 허용할 수 없습니다. 예를 들어, 큐 관리자에 액세스할 필요가 없는 사용자에게는 부여하지 않습니다. 제한 모드는 큐 관리자 데이터를 저장하는 디렉토리의 권한을 수정합니다. 해당 디렉토리는 지정된 애플리케이션 그룹의 구성원만 액세스할 수 있습니다. 큐 관리자와 통신하는 데 사용되는 System V IPC 공유 메모리에 대한 권한도 같은 방법으로 수정됩니다.

애플리케이션 그룹은 다음을 수행할 권한을 갖고 있는 그룹의 이름입니다.

- MQI 애플리케이션 실행
- 모든 IPCC 자원 업데이트
- 일부 큐 관리자 디렉토리의 컨텐츠 변경

큐 관리자에 대해 제한 모드를 사용하려면,

- 큐 관리자의 작성자가 mqm 그룹 및 애플리케이션 그룹에 있어야 합니다.
- mqm 사용자 ID가 애플리케이션 그룹에 있어야 합니다.
- 큐 관리자를 관리하려는 모든 사용자는 mqm 그룹 및 애플리케이션 그룹에 있어야 합니다.
- IBM MQ 애플리케이션을 실행하려는 모든 사용자는 애플리케이션 그룹에 있어야 합니다.

애플리케이션 그룹에 있지 않은 사용자가 실행하는 모든 MQCONN 또는 MQCONNX 호출은 이유 코드 MQRC_Q_MGR_NOT_AVAILABLE과 함께 실패합니다.

중요사항: 많은 운영 체제에서 그룹에 사용자를 추가한 후 인식되게 하려면 해당 사용자를 로그오프하고 다시 로그온해야 합니다.

제한 모드는 IBM MQ 권한 서비스와 함께 작동합니다. 따라서 사용자에게 IBM MQ에 연결하고 IBM MQ 권한 서비스를 사용할 때 필요한 자원에 액세스할 권한도 부여해야 합니다.

ULW IBM MQ 권한 서비스 구성에 대한 추가 정보는 Windows, UNIX and Linux 시스템에서 보안 설정에서 찾을 수 있습니다.

권한 서비스가 제공하는 제어가 큐 관리자 자원의 충분한 격리를 제공하지 않을 때만 IBM MQ 제한 모드를 사용하십시오.

XA 자원 관리자

Use the `XA resource manager` queue manager properties page from the IBM MQ Explorer, or the `XAResourceManager` stanza in the qm.ini file, to specify the following information about the resource managers involved in global units of work coordinated by the queue manager.

글로벌 작업 단위에 참여하는 자원 관리자의 각 인스턴스에 대해 XA 자원 관리자 구성 정보를 수동으로 추가하십시오. 기본값은 제공되지 않습니다.

자원 관리자 속성에 관한 자세한 정보는 데이터베이스 조정을 참조하십시오.

Name= *name*(필수)

이 속성은 자원 관리자 인스턴스를 식별합니다.

`Name` 값은 길이가 최대 31 문자까지 가능합니다. XA-switch 속성에서 정의되는 자원 관리자의 이름을 사용할 수 있습니다. 그러나, 동일한 자원 관리자의 둘 이상의 인스턴스를 사용 중인 경우, 각 인스턴스에 대해 고유한 이름을 구성해야 합니다. 예를 들어 `Name` 문자열에 데이터베이스의 이름을 포함하여 고유성을 보장할 수 있습니다.

IBM MQ는 메시지 및 dspmqtrn 명령의 출력에서 Name 값을 사용합니다.

연관된 큐 관리자가 시작했고 자원 관리자 이름이 적용된 후에는 자원 관리자 인스턴스의 이름을 변경하거나 구성 정보에서 해당 항목을 삭제하지 마십시오.

SwitchFile= *name*(필수)
자원 관리자의 XA 스위치 구조를 포함하는 로드 파일의 완전한 이름입니다.

32비트 애플리케이션에서 64비트 큐 관리자를 사용 중인 경우, name 값은 자원 관리자의 XA 스위치 구조를 포함하는 로드 파일의 기본 이름만 포함해야 합니다.

32비트 파일은 ExitsDefaultPath로 지정된 경로에서 애플리케이션으로 로드됩니다.

64비트 파일은 ExitsDefaultPath64로 지정된 경로에서 큐 관리자로 로드됩니다.

XAOpenString= *string*(선택사항)
자원 관리자의 xa_open 시작점으로 전달될 데이터의 문자열입니다. 문자열의 컨텐츠는 자원 관리자 자체에 따라 다릅니다. 예를 들어, 문자열은 자원 관리자의 이 인스턴스가 액세스할 데이터베이스를 식별할 수 있습니다. 이 속성 정의에 대한 자세한 정보는 다음을 참조하십시오.

- Db2에 대한 자원 관리자 구성 정보 추가
- Oracle에 대한 자원 관리자 구성 정보 추가
- Sybase의 자원 관리자 구성 정보 추가
- Informix®에 대한 자원 관리자 구성 정보 추가

그리고 적절한 문자열에 대해서는 자원 관리자 문서를 참조하십시오.

XACloseString= *string*(선택사항)
자원 관리자의 xa_close 시작점으로 전달될 데이터의 문자열입니다. 문자열의 컨텐츠는 자원 관리자 자체에 따라 다릅니다. 이 속성 정의에 대한 자세한 정보는 다음을 참조하십시오.

- Db2에 대한 자원 관리자 구성 정보 추가
- Oracle에 대한 자원 관리자 구성 정보 추가
- Sybase의 자원 관리자 구성 정보 추가
- Informix에 대한 자원 관리자 구성 정보 추가

그리고 적절한 문자열에 대해서는 데이터베이스 문서를 참조하십시오.

ThreadOfControl=THREAD|PROCESS
Windows 이 속성은 IBM MQ for Windows에 대해 필수입니다. 큐 관리자는 자체 멀티스레드 프로세스 중 하나에서 자원 관리자를 호출해야 할 때 직렬화를 위해 이 값을 사용합니다.

THREAD
자원 관리자가 완전히 스레드 인지입니다. 멀티스레드 IBM MQ 프로세스에서, 복수 스레드에서 외부 자원 관리자로 동시에 XA 함수 호출을 작성할 수 있습니다.

PROCESS
자원 관리자가 스레드 안전하지 않습니다. 멀티스레드 IBM MQ 프로세스에서, 자원 관리자에 대해 한 번에 하나의 XA 함수 호출만 작성할 수 있습니다.

ThreadOfControl 항목은 멀티스레드 애플리케이션 프로세스에서 큐 관리자에 의해 발행되는 XA 함수 호출에 적용되지 않습니다. 일반적으로 서로 다른 스레드에 동시 작업 단위를 갖는 애플리케이션은 이 조작 모드가 각 자원 관리자에 의해 지정되어야 합니다.

Channels 스탠자의 속성

이 속성은 채널의 구성을 판별합니다.

z/OS 이 정보는 IBM MQ for z/OS에 적용할 수 없습니다.

채널에 대한 정보를 지정하려면 Channels 의 IBM MQ Explorer큐 관리자 특성 페이지 또는 qm.ini 파일의 CHANNELS 스탠자를 사용하십시오.

MaxChannels=100|*number*
: 허용되는 현재 채널의 최대 수입니다.

기본값은 100입니다.

필요한 경우 최대 현재 채널 수를 제한하기 위해 **MaxChannels**를 다른 값으로 설정할 수 있습니다. IBM MQ Appliance의 경우 기본값은 999 999 999이며 이는 변경하지 않아야 합니다.

MaxActiveChannels=*MaxChannels_value*
: 언제든지 활성일 수 있는 채널의 최대 수입니다. 기본값은 MaxChannels 속성에 지정된 값입니다.

MaxInitiators=3|*number*
: 시작기의 최대 수입니다. 기본값 및 최대값은 3입니다.

MQIBindType=FASTPATH|STANDARD
: 애플리케이션에 대한 바인딩:

FASTPATH
: 채널이 MQCONNX FASTPATH를 사용하여 연결합니다. 에이전트 프로세스가 없습니다.

STANDARD
: 채널이 STANDARD를 사용하여 연결합니다.

PipeLineLength=1|*number*
: 채널이 사용할 동시 스레드의 최대 수입니다. 기본값은 1입니다. 1보다 큰 모든 값은 2로 처리됩니다.

파이프라이닝을 사용할 때는 채널의 양 끝에 있는 큐 관리자가 1보다 큰 *PipeLineLength*를 갖도록 구성하십시오.

참고: 파이프라이닝은 TCP/IP 채널에만 유효합니다.

AdoptNewMCA= NO |SVR|SDR|RCVR|CLUSRCVR|ALL|FASTPATH
: IBM MQ가 채널을 시작하는 요청을 수신하지만 채널의 인스턴스가 이미 실행 중임을 발견하는 경우, 일부 경우에는 기존 채널 인스턴스가 중지되어야 새 채널 인스턴스가 시작할 수 있습니다. AdoptNewMCA 속성을 사용하면 이 방식으로 종료할 수 있는 채널의 유형을 제어할 수 있습니다.

특정 채널 유형에 대해 AdoptNewMCA 속성을 지정하지만 일치하는 채널 인스턴스가 이미 실행 중이기 때문에 새 채널이 시작하지 못하는 경우,

1. 새 채널은 이전 채널을 종료하도록 요청하여 이전 채널을 중지하려고 합니다.
2. 이전 채널 서버가 AdoptNewMCATimeout 대기 간격이 만기할 때까지 이 요청에 응답하지 않는 경우 이전 채널 서버에 대한 스레드나 프로세스가 종료됩니다.
3. 이전 채널 서버가 단계 2 후 및 AdoptNewMCATimeout 대기 간격이 두 번째로 만료된 후에도 종료되지 않으면 IBM MQ가 CHANNEL IN USE 오류로 채널을 종료합니다.

AdoptNewMCA 기능은 서버, 송신자, 수신자 및 클러스터-서버 채널에 적용됩니다. 송신자 또는 서버 채널의 경우에는 특정 이름을 갖는 채널의 한 인스턴스만이 수신 큐 관리자에서 실행 중일 수 있습니다. 장애 복구 또는 클러스터-수신자 채널의 경우, 특정 이름을 갖는 채널의 복수 인스턴스가 수신 큐 관리자에서 실행 중일 수 있지만 임의 시간에 특정 리모트 큐 관리자의 한 인스턴스만이 실행할 수 있습니다.

참고: AdoptNewMCA는 요청자 또는 서버 연결 채널에서 지원되지 않습니다.

다음 목록에서 하나 이상의 값을 쉼표나 공백으로 구분하여 지정하십시오.

아니오
: AdoptNewMCA 기능은 필요하지 않습니다. 기본값입니다.

SVR
: 서버 채널을 채택합니다.

SDR
: 송신자 채널을 채택합니다.

RCVR
: 수신자 채널을 채택합니다.

CLUSRCVR
: 클러스터 수신자 채널을 채택합니다.

모두
FASTPATH 채널을 제외한 모든 채널 유형을 채택합니다.

FASTPATH
FASTPATH 채널인 경우 이 채널을 채택합니다. 이것은 적절한 채널 유형도 지정되는 경우에만(예: `AdoptNewMCA=RCVR,SVR,FASTPATH`) 발생합니다.

주의!: AdoptNewMCA 속성은 FASTPATH 채널에서 예측할 수 없는 형식으로 작동할 수 있습니다. FASTPATH 채널에서 AdoptNewMCA 속성을 사용하는 경우 신중하게 실행하십시오.

AdoptNewMCATimeout= 60 |1 - 3600
새 채널 인스턴스가 이전 채널 인스턴스의 종료를 대기하는 시간(초)입니다. 1 - 3600 범위의 값을 지정하십시오. 기본값은 60입니다.

AdoptNewMCACheck=QM|ADDRESS|NAME|ALL
`AdoptNewMCA` 속성을 사용으로 설정할 때 필요한 검사의 유형입니다. 가능한 경우, 채널이 부주의하게 또는 악의적으로 시스템 종료되는 것을 보호하기 위한 전체 검사를 수행하십시오. 적어도 채널 이름이 일치하는지 확인하십시오.

QM, *NAME* 또는 *ALL*의 경우에 하나 이상의 값을 쉼표 또는 공백으로 구분하여 지정하십시오.

QM
큐 관리자 이름이 일치하는지 확인합니다.

큐 관리자 이름 자체는 일치하지만, QMID는 아닙니다.

ADDRESS
통신 소스 IP 주소를 검사하십시오. (예: TCP/IP 주소).

참고: 쉼표로 구분된 CONNAME 값이 대상 주소에 적용되므로 이 옵션과는 관련이 없습니다.

다중 인스턴스 큐 관리자가 `hosta`에서 `hostb`로 페일오버되는 경우, 해당 큐 관리자의 아웃바운드 채널은 `hostb`의 소스 IP 주소를 사용합니다. 이것이 `hosta`와 다를 경우 AdoptNewMCACheck=*ADDRESS*가 일치하지 않습니다.

상호 인증과 함께 SSL 또는 TLS를 사용하여 공격자가 기존에 실행 중인 채널을 방해하지 못하도록 할 수 있습니다. 또는 다중 인스턴스 큐 관리자 대신 IP 인수가 포함된 HACMP 유형 솔루션을 사용하거나 네트워크 로드 밸런서를 사용하여 소스 IP 주소를 마스킹하십시오.

이름
채널 이름이 일치하는지 확인하십시오.

모두
큐 관리자 이름 일치, 통신 주소 및 채널 이름 일치를 확인하십시오.

기본값은 `AdoptNewMCACheck=NAME,ADDRESS,QM`입니다.

V 9.0.4 **ChlauthEarlyAdopt = Y| N**
연결 인증 및 채널 인증 규칙이 처리되는 순서는 IBM MQ 클라이언트 애플리케이션 연결의 보안 컨텍스트를 결정하는 데 중요한 요소입니다

주의: **`ChlauthEarlyAdopt`**가 qm.ini 파일에 없는 경우 기본값은 N이지만 IBM MQ 9.0.4에서 모든 큐 관리자는 자동으로 qm.ini 파일에 추가된 **`ChlauthEarlyAdopt`**=Y로 작성됩니다.

`ChlauthEarlyAdopt`는 ADOPTCTX(YES)가 큐 관리자의 연결 인증 AUTHINFO 오브젝트에 설정된 경우 연결 인증을 위해 큐 관리자에 제공된 사용자 ID만 채택합니다.

`ChlauthEarlyAdopt`에 대해 올바른 값은 다음과 같은 값입니다.

Y
채널 인증 규칙을 적용하기 전에 채널이 큐 관리자 연결 인증을 사용하여 애플리케이션에서 제공하는 사용자 ID 및 비밀번호 신임 정보의 유효성을 검증하고 이를 채택합니다. 이 작동 모드에서는 채널 인증 규칙이 연결 인증 검사를 통해 생성되는 사용자 ID와 일치합니다.

N

채널 인증 규칙이 적용될 때까지 채널이 애플리케이션에서 제공하는 사용자 ID 및 비밀번호 신임 정보의 연결 인증 유효성 검증을 지연시킵니다. 이 작동 모드에서는 채널 인증 블로킹 및 맵핑 규칙이 사용자 ID 및 비밀번호 유효성 검증의 결과를 고려할 수 없습니다.

예를 들어, 기본 인증 정보 오브젝트는 **ADOPTCTX(YES)**로 설정되어 있고 사용자 fred가 로그인되어 있습니다. 다음 두 가지 CHLAUTH 규칙이 구성되어 있습니다.

```
SET CHLAUTH('MY.CHLAUTH') TYPE(ADDRESSMAP) DESCR('Block all access by
default') ADDRESS('*') USERSRC(NOACCESS) ACTION(REPLACE)
SET CHLAUTH('MY.CHLAUTH') TYPE(USERMAP) DESCR('Allow user bob and force
CONNAUTH') CLNTUSER('bob') CHCKCLNT(REQUIRED) USERSRC(CHANNEL)
```

다음 명령을 사용자 bob의 채택된 보안 컨텍스트로 인증하려는 의도를 가지고 다음 명령이 실행됩니다.

```
runmqsc -c -u bob QMGR
```

실제로 큐 관리자는 bob이 아니라 fred의 보안 컨텍스트를 사용하고 연결이 실패합니다.

bob의 보안 컨텍스트를 사용하려면 **ChlauthEarlyAdopt**를 Y로 설정해야 합니다.

PasswordProtection = Compatible|always|optional

IBM MQ 8.0에서 TLS를 사용하는 대신 MQCSP 구조에서 보호된 비밀번호를 설정합니다.

MQCSP 비밀번호 보호 사용은 TLS 암호화 설정에 비해 단순하기 때문에 MQCSP 비밀번호 보호는 테스트 및 개발 용도로 유용하지만 TLS 암호화에 비해 안전하지 않습니다.

자세한 정보는 MQCSP 비밀번호 보호를 참조하십시오.

ChlauthIssueWarn = y

SET CHLAUTH 명령에서 WARN = YES 속성을 설정한 경우 메시지 AMQ9787을 생성하려면 이 속성을 설정하십시오.

관련 개념

156 페이지의 『채널 상태』

채널은 언제든지 여러 상태 중 하나일 수 있습니다. 일부 상태에는 하위 상태도 있습니다. 채널은 제공된 상태에서 다른 상태로 이동할 수 있습니다.

TCP, LU62 및 NETBIOS

네트워크 프로토콜 구성 매개변수를 지정하려면 이들 큐 관리자 특성 페이지 또는 qm.ini 파일의 스탠자를 사용하십시오. 채널의 기본 속성을 대체합니다.

TCP

Use the TCP queue manager properties page from the IBM MQ Explorer, or the TCP stanza in the qm.ini file, to specify Transmission Control Protocol/Internet Protocol (TCP/IP) configuration parameters.

Port= 1414| *port_number*

TCP/IP 세션에 대한 10진수 표기법의 기본 포트 번호입니다. IBM MQ 의 잘 알려진 포트 번호는 1414입니다.

Library1= *DLLName1* (IBM MQ for Windows only)

TCP/IP 소켓 DLL의 이름입니다.

기본값은 WSOCK32입니다.

KeepAlive= NO |YES

KeepAlive 기능을 켜거나 끕니다. KeepAlive=YES로 설정하면 TCP/IP가 연결의 다른 측이 여전히 사용 가능한지 주기적으로 검사합니다. 그렇지 않은 경우 채널이 닫힙니다.

ListenerBacklog=number

TCP/IP 리스너에 대한 미해결 요청의 기본 수를 대체합니다.

TCP/IP에서 수신 시, 최대 미해결 연결 요청 수가 설정됩니다. 이것은 리스너가 요청을 승인할 때까지 TCP/IP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다. 기본 리스너 백로그 값은 115 페이지의 표 12에 나와 있습니다.

표 12. 기본 미해결 연결 요청(TCP)	
플랫폼	**기본 ListenerBacklog 값**
Windows서버	100
Windows Workstation	5
Linux	100
Solaris	100
HP-UX	20
AIX 5.3 이상	100

참고: 일부 운영 체제는 표시된 기본값보다 더 큰 값을 지원합니다. 연결 한계 도달을 피하려면 이것을 사용하십시오.

반대로, 일부 운영 체제는 TCP 백로그의 크기를 제한할 수 있으므로 유효 TCP 백로그가 여기에서 요청된 것보다 더 작을 수 있습니다.

백로그가 115 페이지의 표 12에 표시된 값에 도달하는 경우 TCP/IP 연결은 거부되고 채널은 시작할 수 없습니다. 메시지 채널의 경우에는, 채널이 RETRY 상태로 가서 나중에 연결을 재시도하게 됩니다. 클라이언트 연결의 경우 클라이언트가 `MQCONN`으로부터 MQRC_Q_MGR_NOT_AVAILABLE 이유 코드를 수신하고 나중에 연결을 재시도합니다.

다음 특성 그룹은 TCP/IP에서 사용되는 버퍼의 크기를 제어하는 데 사용할 수 있습니다. 값은 운영 체제의 TCP/IP 계층으로 직접 전달됩니다. 이러한 특성을 사용할 때에는 매우 주의해야 합니다. 값이 잘못 설정되면 TCP/IP 성능에 부정적인 영향을 미칠 수 있습니다. 이것이 성능에 미치는 영향에 대한 추가 정보는 사용자 환경에 대한 TCP/IP 문서를 참조하십시오. 값이 0이면 IBM MQ에 의해 버퍼 크기가 고정되는 것과는 반대로 운영 체제가 버퍼 크기를 관리함을 의미합니다.

Connect_Timeout= 0 |number

소켓 연결 시도가 제한시간 초과하기 전의 시간(초)입니다. 기본값 0은 연결 제한시간이 없음을 지정합니다.

IBM MQ 채널 프로세스는 비차단 소켓을 통해 연결합니다. 그러므로 소켓의 다른 끝이 준비되지 않은 경우, connect()는 *EINPROGRESS* 또는 *EWOULDBLOCK*과 함께 즉시 리턴합니다. 그 후에 통신 오류가 보고되면 연결이 다시 최대 총 20회까지 다시 시도됩니다.

`Connect_Timeout`이 0이 아닌 값으로 설정되는 경우, IBM MQ는 소켓이 준비되도록 하기 위한 select() 호출에 대해 규정된 기간 동안 대기합니다. 이 경우 후속 connect() 호출 성공의 가능성이 증가합니다. 이 옵션은 네트워크의 높은 로드로 인해서 연결에 일부 대기 시간이 필요한 상황에서 유익할 수 있습니다.

SndBuffSize=number|0

채널의 송신 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다. 이 스탠자 값은 채널 유형에 더 특정한 스탠자(예: RcvSndBuffSize)에 의해 대체될 수 있습니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Multi IBM MQ 8.0부터 새 큐 관리자가 기본 설정 0으로 자동으로 작성됩니다(82 페이지의 『큐 관리자 구성 파일, qm.ini』 참조).

RcvBuffSize=number|0

채널의 수신 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다. 이 스탠자 값은 채널 유형에 더 특정한 스탠자(예: RcvRcvBuffSize)에 의해 대체될 수 있습니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Multi IBM MQ 8.0부터 새 큐 관리자가 기본 설정 0으로 자동으로 작성됩니다(82 페이지의 『큐 관리자 구성 파일, qm.ini』 참조).

RcvSndBuffSize=number|0

수신자 채널의 송신자 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Multi IBM MQ 8.0부터 새 큐 관리자가 기본 설정 0으로 자동으로 작성됩니다(82 페이지의 『큐 관리자 구성 파일, qm.ini』 참조).

RcvRcvBuffSize=number|0
수신자 채널의 수신 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Multi IBM MQ 8.0부터 새 큐 관리자가 기본 설정 0으로 자동으로 작성됩니다(82 페이지의 『큐 관리자 구성 파일, qm.ini』 참조).

SvrSndBuffSize=number|0
클라이언트 연결 서버 연결 채널의 서버 측에서 사용되는 TCP/IP 송신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Multi IBM MQ 8.0부터 새 큐 관리자가 기본 설정 0으로 자동으로 작성됩니다(82 페이지의 『큐 관리자 구성 파일, qm.ini』 참조).

SvrRcvBuffSize=number|0
클라이언트 연결 서버 연결 채널의 서버 측에서 사용되는 TCP/IP 수신 버퍼의 크기(바이트)입니다. 값이 0으로 설정된 경우 운영 체제 기본값을 사용합니다. 값을 설정하지 않은 경우 IBM MQ 기본값 32768이 사용됩니다.

Multi IBM MQ 8.0부터 새 큐 관리자가 기본 설정 0으로 자동으로 작성됩니다(82 페이지의 『큐 관리자 구성 파일, qm.ini』 참조).

Windows LU62(IBM MQ for Windows 전용)
Use the `LU6.2` queue manager properties page from the IBM MQ Explorer, or the `LU62` stanza in the qm.ini file, to specify SNA LU 6.2 protocol configuration parameters.

TPName
원격 사이트에서 시작할 TP 이름입니다.

Library1= ***DLLName 1***
APPC DLL의 이름입니다.

기본값은 WCPIC32입니다.

Library2= ***DLLName2***
Library1과 동일하며 코드가 두 개의 별도 라이브러리에 저장된 경우에 사용됩니다.

기본값은 WCPIC32입니다.

Windows NetBIOS(IBM MQ for Windows만 해당)
Use the `Netbios` queue manager properties page from the IBM MQ Explorer, or the `NETBIOS` stanza in the qm.ini file, to specify NetBIOS protocol configuration parameters.

LocalName= ***name***
이 머신이 LAN에서 알려지는 이름입니다.

AdapterNum= 0| ***adapter_number***
LAN 어댑터의 번호입니다. 기본값은 어댑터 0입니다.

NumSess= 1| ***number_of_sessions***
할당할 세션의 수입니다. 기본값은 1입니다.

NumCmds= 1| ***number_of_commands***
할당할 명령의 수입니다. 기본값은 1입니다.

NumNames= 1| ***number_of_names***
할당할 이름의 수입니다. 기본값은 1입니다.

Library1= ***DLLName1***
NetBIOS DLL의 이름입니다.

기본값은 NETAPI32입니다.

Windows SPX(IBM MQ for Windows만 해당)

SPX 프로토콜 구성 매개변수를 지정하려면 IBM MQ Explorer의 SPX 큐 관리자 특성 페이지 또는 qm.ini 파일의 SPX 스탠자를 사용하십시오.

Socket= 5E86| *socket_number*

16진 표기법의 SPX 소켓 번호입니다. 기본값은 X'5E86'입니다.

BoardNum= 0| *adapter_number*

LAN 어댑터 번호입니다. 기본값은 어댑터 0입니다.

KeepAlive=NO|YES

KeepAlive 기능을 켜거나 끕니다.

KeepAlive=YES로 설정하면 연결의 다른 편 끝이 여전히 사용 가능한지 SPX가 주기적으로 검사합니다. 그렇지 않은 경우 채널이 닫힙니다.

Library1= *DLLName1*

SPX DLL의 이름입니다.

기본값은 WSOCK32.DLL입니다.

Library2= *DLLName2*

LibraryName1과 동일하며 코드가 두 개의 별도 라이브러리에 저장된 경우에 사용됩니다.

기본값은 WSOCK32.DLL입니다.

ListenerBacklog=number

SPX 리스너에 대한 미해결 요청의 기본 수를 대체합니다.

SPX에서 수신 시, 최대 미해결 연결 요청 수가 설정됩니다. 이것은 리스너가 요청을 승인하기 위해 SPX 소켓에서 대기 중인 요청의 백로그인 것으로 간주됩니다. 기본 리스너 백로그 값은 117 페이지의 표 13에 나와 있습니다.

표 13. 기본 미해결 연결 요청(SPX)

플랫폼	기본 ListenerBacklog 값
Windows서버	100
Windows Workstation	5

참고: 일부 운영 체제는 표시된 기본값보다 더 큰 값을 지원합니다. 연결 한계 도달을 피하려면 이것을 사용하십시오.

반대로, 일부 운영 체제는 SPX 백로그의 크기를 제한할 수 있으므로 유효 SPX 백로그가 여기에서 요청된 것보다 더 작을 수 있습니다.

백로그가 117 페이지의 표 13에 표시된 값에 도달하는 경우 SPX 연결은 거부되고 채널은 시작할 수 없습니다. 메시지 채널의 경우에는, 채널이 RETRY 상태로 가서 나중에 연결을 재시도하게 됩니다. 클라이언트 연결의 경우 클라이언트가 MQCONN으로부터 MQRC_Q_MGR_NOT_AVAILABLE 이유 코드를 수신하고 나중에 연결을 재시도해야 합니다.

엑시트 경로

Exits 의 IBM MQ Explorer큐 관리자 특성 페이지 또는 qm.ini 파일의 ExitPath 스탠자를 사용하여 큐 관리자 시스템의 사용자 엑시트 프로그램에 대한 경로를 지정하십시오.

ExitsDefaultPath= *string*

ExitsDefaultPath 속성은 다음의 위치를 지정합니다.

- 클라이언트의 32비트 채널 엑시트
- 서버의 32비트 채널 엑시트 및 데이터 변환 엑시트
- 규정되지 않은 XA 스위치 로드 파일

ExitsDefaultPath64= *string*

ExitsDefaultPath64 속성은 다음의 위치를 지정합니다.

- 클라이언트의 64비트 채널 엑시트
- 서버의 64비트 채널 엑시트 및 데이터 변환 엑시트
- 규정되지 않은 XA 스위치 로드 파일

API 엑시트

서버의 경우, IBM MQ Explorer의 Exits 큐 관리자 특성 페이지 또는 qm.ini 파일의 ApiExitLocal 스탠자를 사용하여 큐 관리자에 대한 API 엑시트 루틴을 식별하십시오. 클라이언트의 경우 큐 관리자에 대한 API 엑시트 루틴을 식별하려면 mqclient.ini 파일의 ApiExitLocal 스탠자를 수정하십시오.

Windows 시스템에서는 amqmdain 명령을 사용하여 API 엑시트에 대한 항목을 변경할 수 있습니다. (모든 큐 관리자에 대한 API 엑시트 루틴을 식별하려면 103 페이지의 『API 엑시트』에서 설명하는 대로 ApiExitCommon 및 ApiExitTemplate 스탠자를 사용합니다.)

API 엑시트가 올바르게 작업하기 위해서는 서버의 메시지가 변환되지 않고 클라이언트로 전송되어야 함을 참고하십시오. API 엑시트가 메시지를 처리한 후, 메시지가 클라이언트에서 변환되어야 합니다. 그러므로 이것은 클라이언트에 모든 변환 엑시트를 설치했어야 합니다.

이러한 스탠자의 속성에 대한 전체 설명은 API 엑시트 구성을 참조하십시오.

V 9.0.5 진단 메시지 로깅

IBM MQ의 진단 메시지 로그는 IBM MQ 시스템의 다양한 컴포넌트가 IBM MQ 구성 및 런타임 상태 변경 및 문제점과 관련된 진단 메시지를 보고 할 수 있게 하는 메커니즘입니다.

이러한 로그는 때때로 IBM MQ 오류 로그라고도 하지만 항상 IBM MQ 정보 및 경고 메시지와 오류 메시지가 포함되어 있습니다. 이러한 로그를 보고하는 IBM MQ의 세 개의 기본 컴포넌트는 다음과 같습니다.

- 큐 관리자
- IBM MQ 클라이언트
- IBM MQ 시스템의 나머지

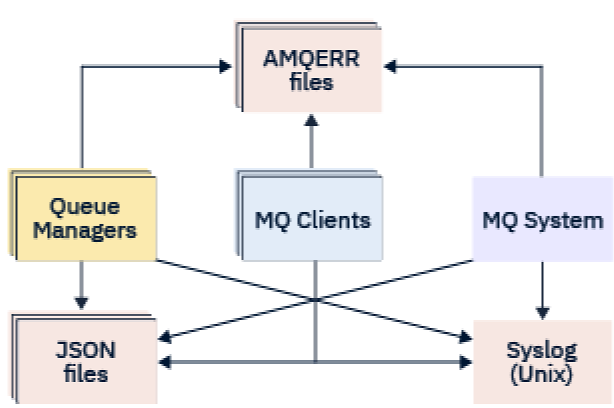
IBM MQ는 진단 메시지 서비스라고 하는 다양한 방법을 통해 진단 메시지를 보고하는 기능을 지원하므로 이 정보를 기록하고 사용하는 데 적합한 방법을 사용할 수 있습니다.

- AMQERRnn 로그 파일
- JSON 형식화된 로그 파일
- UNIX JSON 형식의 Syslog

IBM MQ에 의한 JSON 출력은 JSON 로그 또는 syslog 레코드의 각 개별 행이 유효한 JSON 오브젝트를 나타내도록 단일 행 JSON 오브젝트로 형식화됩니다. 전체적으로 로그는 단일 JSON 오브젝트로 캡슐화되지 않습니다.

다음 그림은 큐 관리자, IBM MQ 클라이언트 및 IBM MQ 시스템이 설명된 방법을 사용하여 모두 진단 메시지를 보고할 수 있다는 것을 표시합니다.

그림 6. *IBM MQ*의 다른 부분이 진단 메시지를 보고하는 방법



IBM MQ 진단 로그 구성 방법:

진단 로그는 `qm.ini` 파일 내의 스탠자를 필요로 하는 IBM MQ 컴포넌트에 대한 스탠자를 사용하여 정의되고 사용자 정의됩니다. 각 고유 로깅 엔드 포인트는 ini 파일 내에서 정의된 사용자 정의와 함께 ini 파일 내의 자체 스탠자 아래에 정의됩니다. 사용자 정의는 다음을 포함할 수 있습니다.

- 롤오버가 발생하기 전에 줄 바꾸는 로그 파일의 크기. Syslog에는 적용되지 않음
- 로그 메시지의 심각도에 기반으로한 모든 필터링 및
- 억제할 특정 메시지 코드.

IBM MQ는 특정 로그 스탠자가 특정 역할을 수행할 수 있도록 세 가지 유형의 로깅 엔드 포인트 중 하나 또는 전체에 쓰도록 구성할 수 있습니다. 마찬가지로, 다중 파일 서비스가 정의될 수 있습니다. 예를 들면, 다음과 같습니다.

- JSON 형식은 로컬 및 클라우드 환경에서 자동화된 도구를 통해 구문 분석을 쉽게 수행 할 수 있습니다.
- Syslog 출력은 IBM MQ 컴포넌트가 시스템의 다른 제품과 일치하는 공통 OS 로깅 위치에 진단 정보를 통합할 수 있도록 허용합니다.
- 심각도를 기반으로 필터링된 로그 엔드 포인트는 예를 들어, 시스템에서 심각한 오류만 기록할 특정 로그 파일을 허용합니다.

구성되는 진단 로그 기록 스타일과 관계없이 기존 진단 파일은 IBM MQ 시스템 로그 디렉토리(`/var/mqm/errors/AMQERRnn.log`)에 있고, 사용되는 다른 로깅 구성과 함께 특정 큐 관리자 로그 디렉토리(`/var/mqm/qmgrs/<qmgr_name>/errors/AMQERRnn.log`)가 항상 기록됩니다.

큐 관리자의 경우, 필수 로그의 선택적 구성은 120 페이지의 『진단 메시지 서비스 스탠자』의 속성을 지정하여 수행할 수 있습니다.

다양한 스탠자 영역

추가 스탠자는 IBM MQ의 다른 영역에 적용할 수 있습니다.

Qmgr(qm.ini)
: 큐 관리자가 생성한 로그 메시지에 적용

System(mqs.ini)
: 시스템에 의해 적용된 로그 메시지에 적용됩니다. 이 옵션은 큐 관리자가 자체 로그에 액세스하거나 쓸 수 없는 경우를 제외하고는 큐 관리자와 관련이 없습니다.

Templates(mqs.ini)
: 큐 관리자가 작성될 때 `qm.ini`에 복사되는 하나 이상의 스탠자를 템플리트로 사용합니다.

Client(mqclient.ini)
: 리모트 큐 관리자에 대한 클라이언트 모드, 예를 들어, **`runmqsc`**와 같은 클라이언트 조작에 적용됩니다.

JSON 형식화된 로그와 기존 형식화된 로그 간의 변환

mqrc 명령은 JSON과 기존 형식화된 로그 간의 많은 변환과 다양한 언어 간의 변환을 허용하도록 개선되었습니다.

관련 참조

120 페이지의 『진단 메시지 서비스 스탠자』
사용 가능한 진단 메시지 서비스 옵션은 IBM MQ 진단 로깅을 사용자 정의하여 로그 출력을 IBM MQ의 다른 컴포넌트에서 다른 로그 끝점으로 보낼 수 있습니다.

120 페이지의 『UNIX, Linux, and Windows의 QMErrorLog 스탠자』
QMErrorLog 서비스는 큐 관리자와 관련된 진단 메시지를 출력하기 위해 사용되는 기존의 IBM MQ 진단 로깅 서비스입니다. QMErrorLog 서비스는 지속적으로 실행되며, 끌 수는 없지만 어느 정도 사용자 정의할 수 있습니다.

123 페이지의 『진단 메시지 서비스』
구성 파일의 DiagnosticSystemMessages, DiagnosticMessages 및 DiagnosticMessagesTemplate 스탠자에 지정된 다음 진단 메시지 서비스 및 해당 서비스별 속성을 정의할 수 있습니다.

ULW *UNIX, Linux, and Windows*의 *QMErrorLog* 스탠자

QMErrorLog 서비스는 큐 관리자와 관련된 진단 메시지를 출력하기 위해 사용되는 기존의 IBM MQ 진단 로깅 서비스입니다. QMErrorLog 서비스는 지속적으로 실행되며, 끌 수는 없지만 어느 정도 사용자 정의할 수 있습니다.

이 스탠자는 IBM MQ 시스템 또는 클라이언트 ini 파일 구성에 적용할 수 없습니다.

IBM MQ Explorer에서 확장 큐 관리자 속성 페이지 또는 qm.ini 파일의 QMErrorLog 스탠자를 사용하여 IBM MQ 오류 로그의 조작 및 컨텐츠를 수정하십시오.

주의: Windows 플랫폼에서 로컬 큐 관리자를 사용하는 경우에만 IBM MQ Explorer 를 사용하여 변경할 수 있습니다.

V 9.0.4 **ErrorLogSize=** ***maxsize***

백업으로 복사할 큐 관리자 오류 로그의 크기를 지정합니다. *maxsize*는 32768 - 2147483648바이트 범위에 있어야 합니다. **ErrorLogSize**를 지정하지 않은 경우 기본값인 33554432바이트(32MB)가 사용됩니다.

필요한 경우 이 속성을 사용하여 최대 크기를 다시 이전 최대 크기인 2MB로 줄일 수 있습니다.

중요사항: IBM MQ 9.0.4에서는 **ErrorLogSize** 속성이 기본 크기가 늘어났습니다. 이는 IBM MQ 9.0.3에서 변경된 것입니다.

MQMAXERRORLOGSIZE 환경 변수를 사용하여 로그의 크기를 설정할 수 있습니다.

ExcludeMessage= ***msgIds***

큐 관리자 오류 로그에 기록하지 않을 메시지를 지정합니다.

자세한 정보는 120 페이지의 『진단 메시지 서비스 스탠자』의 ExcludeMessage를 참조하십시오.

SuppressMessage= ***msgIds***

지정된 시간 간격에 한 번만 큐 관리자 오류 로그에 기록되는 메시지를 지정합니다. `SuppressMessage`와 `ExcludeMessage`에 동일한 메시지 ID가 지정될 경우 해당 메시지가 제외됩니다.

이 옵션은 `MQ client.ini`에서 정의된 진단 메시지 서비스에 적용할 수 없습니다.

자세한 정보는 120 페이지의 『진단 메시지 서비스 스탠자』의 SuppressMessage를 참조하십시오.

SuppressInterval= ***length***

`SuppressMessage`에 지정된 메시지가 큐 관리자 오류 로그에 한 번만 기록되는 시간 간격(초)을 지정합니다. *length*는 1 - 86400초 범위에 있어야 합니다. `SuppressInterval`이 지정되지 않으면 기본값 30초가 사용됩니다.

관련 참조

120 페이지의 『진단 메시지 서비스 스탠자』

사용 가능한 진단 메시지 서비스 옵션은 IBM MQ 진단 로깅을 사용자 정의하여 로그 출력을 IBM MQ의 다른 컴포넌트에서 다른 로그 끝점으로 보낼 수 있습니다.

123 페이지의 『진단 메시지 서비스』

구성 파일의 DiagnosticSystemMessages, DiagnosticMessages 및 DiagnosticMessagesTemplate 스탠자에 지정된 다음 진단 메시지 서비스 및 해당 서비스별 속성을 정의할 수 있습니다.

V 9.0.5 진단 메시지 서비스 스탠자

사용 가능한 진단 메시지 서비스 옵션은 IBM MQ 진단 로깅을 사용자 정의하여 로그 출력을 IBM MQ의 다른 컴포넌트에서 다른 로그 끝점으로 보낼 수 있습니다.

다음 이름 중 하나로 된 스탠자를 사용하여 추가 진단 메시지 서비스를 사용할 수 있습니다.

- **DiagnosticSystemMessages**

  시스템 오류 로그로 이동할 진단 메시지가 생성될 때 사용할 서비스를 정의합니다. `mqs.ini` 또는 `mqclient.ini` 파일에서 유효합니다.

  클라이언트 애플리케이션은 mqclient.ini 파일 및 in mqs.ini 파일에서 **DiagnosticSystemMessages** 스탠자를 사용하는데, **DiagnosticSystemMessages** 스탠자는 큐 관리자 컨텍스트가 없는 서버 애플리케이션의 메시지를 제어합니다.

모든 메시지를 syslog 서비스에 추가적으로 기록하는 큐 관리자 및 애플리케이션을 구성할 수 있습니다.

- **`DiagnosticMessages`**

  큐 관리자 오류 로그로 이동할 진단 메시지가 생성될 때 사용할 서비스를 정의합니다. `qm.ini` 파일에서만 유효합니다.

- **`DiagnosticMessagesTemplate`**

  큐 관리자가 작성될 때 `mqs.ini` 파일에서 `qm.ini` 파일의 **`DiagnosticMessages`**로 복사되는 스탠자입니다.

진단 메시지를 표시하려면 mqrc 명령을 사용하십시오.

스탠자 속성

주의: `Service` 및 스탠자 이름은 필수입니다.

name=<stanzaname>
스탠자의 이름. 값은 ini 파일에서 고유해야 합니다.

Service= *type of service*

이 속성은 이 스탠자에서 사용 가능한 서비스를 정의하며 이때 서비스 이름은 대소문자를 구분하지 않습니다.

예를 들어 `syslog`를 추가 서비스로 사용하려면 다음과 같이 입력하십시오.

```
Service=syslog
```

진단 메시지 서비스 스탠자에 사용할 수 있는 123 페이지의 『진단 메시지 서비스』 및 해당 속성을 참조하십시오.

다음 선택적 속성을 스탠자에 추가할 수 있습니다.

- ExcludeMessage
- SuppressMessage
- SuppressInterval
- 122 페이지의 『심각도』

ExcludeMessage= *msgIds*
큐 관리자 오류 로그에 기록하지 않을 메시지를 지정합니다. IBM MQ 시스템이 과도하게 사용된 경우, 많은 채널이 중지 및 시작되고 많은 정보 메시지가 z/OS 콘솔 및 하드카피 로그로 전송됩니다. IBM MQ - IMS 브릿지 및 버퍼 관리자는 많은 수의 정보 메시지를 생성할 수도 있으므로, 메시지를 제외하면 사용자가 필요한 경우 많은 수의 메시지를 수신하지 못하게 합니다. *msgIds*에는 다음의 쉼표로 구분된 메시지 ID 목록이 포함됩니다.

5211 - 최대 특성 이름 길이가 초과되었습니다.
5973 - 분배된 발행/구독 등록이 억제됨
5974 - 분배된 발행/구독 등록이 억제됨
6254 - 시스템이 공유 라이브러리를 동적으로 로드할 수 없음
7234 - 로드된 메시지 수
8245 - 엔티티의 권한이 오브젝트를 표시하기에 충분하지 않습니다.
9001 - 채널 프로그램이 정상적으로 종료됨
9002 - 채널 프로그램이 시작됨
9202 - 원격 호스트가 사용 불가능함
9208 - 호스트로부터 수신 시 오류
9209 - 연결 종료됨
9228 - 채널 응답자를 시작할 수 없음
9489 - SVRCONN 최대 인스턴스 한계 초과
9490 - 클라이언트당 SVRCONN 최대 인스턴스 한계 초과

9508 - 큐 관리자에 연결할 수 없음
9524 - 리모트 큐 관리자가 사용 불가능함
9528 - 사용자가 채널을 닫도록 요청함
9545 - 연결 끊기 간격 만기
9558 - 원격 채널 사용 불가능
9637 - 채널에 인증서 결핍
9776 - 채널이 사용자 ID에 의해 차단됨
9777 - 채널이 NOACCESS 맵에 의해 차단됨
9782 - 연결이 주소에 의해 차단됨
9999 - 채널 프로그램이 비정상적으로 종료됨

SuppressMessage= ***msgIds***

지정된 시간 간격에 한 번만 큐 관리자 오류 로그에 기록되는 메시지를 지정합니다. IBM MQ 시스템이 과도하게 사용된 경우, 많은 채널이 중지 및 시작되고 많은 정보 메시지가 z/OS 콘솔 및 하드카피 로그로 전송됩니다. IBM MQ - IMS 브릿지 및 버퍼 관리자는 많은 수의 정보 메시지를 생성할 수도 있으므로, 메시지를 억제하면 필요한 경우 많은 반복 메시지를 수신하지 못하게 됩니다. 시간 간격은 `SuppressInterval`에 의해 지정됩니다. *msgIds*에는 다음의 쉼표로 구분된 메시지 ID 목록이 포함됩니다.

5211 - 최대 특성 이름 길이가 초과되었습니다.
5973 - 분배된 발행/구독 등록이 억제됨
5974 - 분배된 발행/구독 등록이 억제됨
6254 - 시스템이 공유 라이브러리를 동적으로 로드할 수 없음
7234 - 로드된 메시지 수
8245 - 엔티티의 권한이 오브젝트를 표시하기에 충분하지 않습니다.
9001 - 채널 프로그램이 정상적으로 종료됨
9002 - 채널 프로그램이 시작됨
9202 - 원격 호스트가 사용 불가능함
9208 - 호스트로부터 수신 시 오류
9209 - 연결 종료됨
9228 - 채널 응답자를 시작할 수 없음
9489 - SVRCONN 최대 인스턴스 한계 초과
9490 - 클라이언트당 SVRCONN 최대 인스턴스 한계 초과
9508 - 큐 관리자에 연결할 수 없음
9524 - 리모트 큐 관리자가 사용 불가능함
9528 - 사용자가 채널을 닫도록 요청함
9545 - 연결 끊기 간격 만기
9558 - 원격 채널 사용 불가능
9637 - 채널에 인증서 결핍
9776 - 채널이 사용자 ID에 의해 차단됨
9777 - 채널이 NOACCESS 맵에 의해 차단됨
9782 - 연결이 주소에 의해 차단됨
9999 - 채널 프로그램이 비정상적으로 종료됨

`SuppressMessage`와 `ExcludeMessage`에 동일한 메시지 ID가 지정될 경우 해당 메시지가 제외됩니다.

이 옵션은 `MQ client.ini`에서 정의된 진단 메시지 서비스에 적용할 수 없습니다.

SuppressInterval= ***length***

`SuppressMessage`에 지정된 메시지가 큐 관리자 오류 로그에 한 번만 기록되는 시간 간격(초)을 지정합니다. *length*는 1 - 86400초 범위에 있어야 합니다. **`SuppressInterval`**이 지정되지 않으면 기본값 30초가 사용됩니다.

심각도

쉼표로 구분된 심각도 레벨 목록입니다. 심각도 레벨의 이름은 대소문자를 구분하지 않습니다. 허용되는 값은 다음과 같습니다.

- I(정보 또는 0)

- W(경고 또는 10)
- E(오류 또는 20 및 30)
- S(중지 또는 40)
- T(시스템 또는 50)

참고사항:

1. 기본값은 `all`입니다.
2. 선택한 심각도 레벨의 메시지만 서비스에 제공됩니다.

   다른 방법으로, 지정된 오류 레벨 및 그 이상의 모든 레벨을 표시하는 더하기 문자(+)를 사용할 수 있습니다. 예를 들어 오류를 모두 표시하려면 다음과 같이 입력하십시오.

```
Severities=E+
```

관련 참조

120 페이지의 『UNIX, Linux, and Windows의 QMErrorLog 스탠자』
QMErrorLog 서비스는 큐 관리자와 관련된 진단 메시지를 출력하기 위해 사용되는 기존의 IBM MQ 진단 로깅 서비스입니다. QMErrorLog 서비스는 지속적으로 실행되며, 끌 수는 없지만 어느 정도 사용자 정의할 수 있습니다.

123 페이지의 『진단 메시지 서비스』
구성 파일의 DiagnosticSystemMessages, DiagnosticMessages 및 DiagnosticMessagesTemplate 스탠자에 지정된 다음 진단 메시지 서비스 및 해당 서비스별 속성을 정의할 수 있습니다.

V 9.0.5 진단 메시지 서비스

구성 파일의 DiagnosticSystemMessages, DiagnosticMessages 및 DiagnosticMessagesTemplate 스탠자에 지정된 다음 진단 메시지 서비스 및 해당 서비스별 속성을 정의할 수 있습니다.

다음과 같은 진단 메시지 서비스가 정의되었습니다.

파일

이 서비스는 QMErrorLog 서비스와 유사한 방식으로 필터링되지 않은 메시지를 파일로 전송합니다. 지정된 **Format**에 따라 기존 텍스트 형식 또는 특정 JSON 형식이 사용됩니다. By default, there are three files called `AMQERR01.LOG`, `AMQERR02.LOG`, and `AMQERR03.LOG` or `AMQERR01.json`, `AMQERR02.json`, and `AMQERR03.json`, depending upon the **Format** property, and these rollover based on the configured size.

다음 속성은 파일 스탠자에서만 지원됩니다.

FilePath

로그 파일을 기록하는 경로입니다. 기본값은 AMQERR01.log 파일(즉, 시스템 또는 큐 관리자)와 동일한 위치입니다. 경로는 절대 경로여야 하지만, 대체 가능한 삽입을 포함할 수 있습니다. 예를 들면, 다음과 같습니다.

+MQ_Q_MGR_DATA_PATH+

큐 관리자 진단 메시지 디렉토리의 상위에 대한 전체 경로. 기본적으로 UNIX 플랫폼의 경우: `/var/mqm/qmgrs/<QM_name>`, Windows, `C:\Program Data\IBM\MQ\qmgrs\<QM_name>`입니다.

+MQ_DATA_PATH+

시스템 진단 메시지 디렉토리의 상위에 대한 전체 경로. 기본적으로 UNIX 플랫폼의 경우:`/var/mqm`, Windows, `C:\Program Data\IBM\MQ`입니다.

기존 오류 디렉토리를 사용하고 있지 않으면, 적절한 권한으로 이 경로를 작성해야 합니다.

FilePrefix

로그 파일의 접두부입니다. 기본값은 AMQERR입니다.

FileSize

로그가 롤오버되는 크기입니다. 시멘틱적으로 동일한 120 페이지의 『UNIX, Linux, and Windows의 QMErrorLog 스탠자』의 ErrorLogSize 특성과 마찬가지로 기본값은 32MB입니다.

참고: ErrorLogSize 특성은 사용자 정의 진단 서비스가 아닌 기본 오류 로그 서비스에만 적용됩니다.

IBM MQ 9.0.5에서 **MQMAXERRORLOGSIZE** 환경 변수를 사용하여 로그 크기를 설정할 수 있습니다.

Format

이 파일의 형식입니다. 값은 *text*(추가 QMErrorLog 스타일 서비스의 경우) 또는 기본값인 *json*일 수 있습니다.

이 속성의 설정에 따라 파일 접미부가 .LOG이거나 .json입니다.

예를 들어 큐 관리자의 qm.ini 파일을 편집하고 다음 스탠자를 추가하십시오.

```
DiagnosticMessages:
   Service = File
   Name = JSONLogs
   Format = json
   FilePrefix = AMQERR
```

다시 시작 이후 큐 관리자의 ERRORS 디렉토리에는 AMQERR0x.json 파일이 있습니다.

파일 서비스는 여러 개 정의할 수 있습니다. 이 경우 다음 예에 표시된 구성이 허용됩니다. 여기서 서로 다른 태그의 메시지는 다른 세트의 로그로 분할됩니다.

```
DiagnosticMessages:
  Name=ErrorsToFile
  Service=File
  Severities=E+
  FilePrefix=OnlyErrors

DiagnosticMessages:
  Name=NonErrorstoFile
  Service=File
  Severities=1 W
  FilePrefix=Information
```

시스템 로그

Syslog 서비스는 Windows 또는 IBM i에서 사용 불가능함

하나의 Syslog 서비스만 정의할 수 있으며 Syslog 서비스는 JSON 형식 진단 메시지 스펙을 사용하여 필터링되지 않은 모든 메시지를 syslog로 보냅니다. 정보는 msgID 및 삽입에서 시작하여 표에 표시된 순서대로 syslog에 추가됩니다.

메시지의 심각도는 다음과 같은 방식으로 syslog 레벨에 맵핑됩니다.

심각도(Severity)	레벨
0	LOG_INFO
10	LOG_WARNING
20	LOG_ERR
30	LOG_ERR
40	LOG_ALERT
50	LOG_ALERT

다음 속성은 syslog 스탠자에서만 지원됩니다.

Ident

syslog 항목과 연관된 **ident** 값을 정의합니다. 기본값은 *ibm-mq*입니다.

다음 예는 Syslog로 전송되는 오류 메시지를 표시합니다.

```
DiagnosticMessages:
  Name=ErrorsToSyslog
  Service=Syslog
  Severities=E +
  FilePrefix=AllErrors
```

일반 스탠자 속성에 대한 자세한 정보는 120 페이지의 『진단 메시지 서비스 스탠자』의 내용을 참조하십시오.

참고사항:

1. 파일 서비스의 경우에만 각기 다른 이름을 사용하는 여러 개의 스탠자가 있을 수 있습니다. 시퀀스의 최종 이름을 사용하는 정의만 적용됩니다.
2. 스탠자 값에 대한 변경사항은 큐 관리자를 재시작할 경우에만 적용됩니다.

큐 관리자 기본 바인드 유형

IBM MQ Explorer의 Extended 큐 관리자 특성 페이지 또는 qm.ini 파일의 `Connection` 스탠자를 사용하여 기본 바인드 유형을 지정하십시오. 하나가 필요한 경우 `Connection` 스탠자를 작성해야 함을 참고하십시오.

DefaultBindType=SHARED|ISOLATED

DefaultBindType이 ISOLATED로 설정되는 경우, 애플리케이션과 큐 관리자가 별도의 프로세스에서 실행하며 그들 사이의 자원이 공유되지 않습니다.

DefaultBindType이 SHARED로 설정되는 경우, 애플리케이션과 큐 관리자가 별도의 프로세스에서 실행하지만 그들 사이에 일부 자원이 공유됩니다.

기본값은 SHARED입니다.

주의: **`DefaultBindType`** 는 모든 MQCONN 호출과 MQCNO_STANDARD_BINDING을 사용하여 MQCONNX를 사용하는 경우에 적용됩니다.

`DefaultBindType`을 변경하면 일부 애플리케이션의 성능이 저하됩니다.

큐 관리자 구성 파일의 SSL 스탠자

큐 관리자에서 TLS 채널을 구성하려면 큐 관리자 구성 파일의 SSL 스탠자를 사용하십시오.

OCSP(Online Certificate Status Protocol)

인증서는 AuthorityInfoAccess 확장을 포함할 수 있습니다. 이 확장은 OCSP(Online Certificate Status Protocol)를 통해 접속될 서버를 지정합니다. 큐 관리자의 SSL 또는 TLS 채널이 AuthorityInfoAccess 확장을 사용하도록 허용하려면, 그 안에서 이름 지정된 OCSP 서버가 사용 가능하고 올바르게 구성되고 네트워크를 통해 액세스 가능한지 확인하십시오. 자세한 정보는 폐지된 인증서에 대한 작업을 참조하십시오.

CrlDistributionPoint(CDP)

인증서는 CrlDistributionPoint 확장을 포함할 수 있습니다. 이 확장은 인증서 폐기 목록(CRL)을 다운로드하는 데 사용되는 프로토콜 및 접속될 서버를 둘 다 지정하는 URL을 포함합니다.

큐 관리자의 SSL 또는 TLS 채널이 CrlDistributionPoint 확장을 사용하도록 허용하려면, 그 안에서 이름 지정된 CDP 서버가 사용 가능하고 올바르게 구성되고 네트워크를 통해 액세스 가능한지 확인하십시오.

SSL 스탠자

큐 관리자의 TLS 채널이 다음 기능을 사용하려고 시도하는 방법 및 사용 시 문제점이 발생하는 경우 대응하는 방법을 구성하려면 `qm.ini` 파일에서 SSL 스탠자를 사용하십시오.

다음의 각 경우에서 제공되는 값이 나열된 올바른 값의 하나가 아닌 경우 기본값이 사용됩니다. 올바르지 않은 값이 지정됨을 언급하는 오류 메시지가 기록되지 않습니다.

CDPCheckExtensions=YES|NO

CDPCheckExtensions는 이 큐 관리자의 TLS 채널이 CrlDistributionPoint 인증서 확장자에서 이름 지정되는 CDP 서버를 검사하려고 시도할지 여부를 지정합니다.

- YES: TLS 채널이 CDP 서버를 검사하여 디지털 인증서가 폐기되었는지 여부를 판별하려고 시도합니다.
- NO: TLS 채널이 CDP 서버를 검사하려고 시도하지 않습니다. 이 값은 기본값입니다.

OCSPAuthentication=REQUIRED|WARN|OPTIONAL

OCSPAuthentication은 OCSP 서버로부터 폐기 상태를 판별할 수 없을 때 취할 조치를 지정합니다.

OCSP 검사가 가능한 경우 TLS 채널 프로그램이 OCSP 서버에 접속하려고 시도합니다.

채널 프로그램이 어떤 OCSP 서버에도 접속할 수 없는 경우 또는 인증서의 폐기 상태를 제공할 수 있는 서버가 없는 경우 OCSPAuthentication 매개변수의 값이 사용됩니다.

- `REQUIRED`: 폐기 상태를 판별하지 못하면 연결이 오류와 함께 닫힙니다. 이 값은 기본값입니다.
- `WARN`: 폐기 상태를 판별하지 못하면 경고 메시지가 큐 관리자 오류 로그에 기록되지만, 연결은 계속될 수 있습니다.
- `OPTIONAL`: 폐기 상태를 판별하지 못하면 연결이 자동으로 진행할 수 있습니다. 경고나 오류가 제공되지 않습니다.

OCSPCheckExtensions= YES|NO

OCSPCheckExtensions는 이 큐 관리자의 TLS 채널이 AuthorityInfoAccess 인증서 확장자에서 이름 지정된 OCSP 서버를 검사하려고 시도할지 여부를 지정합니다.

- `YES`: TLS 채널이 OCSP 서버를 검사하여 디지털 인증서가 폐기되었는지 여부를 판별하려고 시도합니다. 이 값은 기본값입니다.
- `NO`: TLS 채널이 OCSP 서버를 검사하려고 시도하지 않습니다.

SSLHTTPProxyName=*string*

문자열은 OCSP 검사를 위해 GSKit에서 사용될 HTTP 프록시 서버의 호스트 이름 또는 네트워크 주소입니다. 이 주소 뒤에 선택적 포트 번호를 괄호로 묶어 표시할 수 있습니다. 포트 번호가 지정되지 않으면 기본 HTTP 포트 80이 사용됩니다.

Solaris HP-UX AIX AIX, 및 Solaris SPARC 플랫폼, 및 HP-UX PA-RISC 플랫폼, 네트워크 주소는 IPv4 주소만 가능합니다.

다른 플랫폼에서는 네트워크 주소가 IPv4 또는 IPv6 주소가 될 수 있습니다.

예를 들어 방화벽이 OCSP 응답자의 URL에 대한 액세스를 막는 경우 이 속성이 필요할 수 있습니다.

엑시트 특성

큐 관리자의 엑시트 특성에 관한 정보를 지정하려면 IBM MQ Explorer의 클러스터 큐 관리자 특성 페이지 또는 qm.ini 파일의 ExitPropertiesLocal 스탠자를 사용하십시오. 그렇지 않으면 **amqmdain** 명령을 사용하여 설정할 수 있습니다.

기본적으로, 이 설정은 머신 전체 구성(100 페이지의 『엑시트 특성』에서 설명됨)의 ExitProperties 스탠자에 있는 CLWLMode 속성에서 상속됩니다. 이 큐 관리자를 다른 방법으로 구성하려는 경우에만 이 설정을 변경하십시오. 이 값은 클러스터 큐 관리자 특성 페이지의 클러스터 워크로드 모드 속성을 사용하여 개별 큐 관리자에 대해 대체될 수 있습니다.

CLWLMode= SAFE |FAST

클러스터 워크로드(CLWL) 엑시트를 사용하면 클러스터에서 MQI 호출(예: `MQOPEN`, `MQPUT`)에 대한 응답으로 열리는 클러스터 큐를 지정할 수 있습니다. CLWL 엑시트는 CLWLMode 속성에 지정한 값에 따라 FAST 모드 또는 SAFE 모드로 실행됩니다. CLWLMode 속성을 생략하면 클러스터 워크로드 엑시트가 SAFE 모드로 실행됩니다.

안전

큐 관리자와는 별도의 프로세스로 CLWL 엑시트를 실행하십시오. 기본값입니다.

SAFE 모드로 실행할 때 사용자 작성 CLWL 엑시트에서 문제점이 발생하면 다음이 발생합니다.

- CLWL 서버 프로세스(amqzlwa0)가 실패합니다.
- 큐 관리자가 CLWL 서버 프로세스를 재시작합니다.
- 오류 로그에 오류가 보고됩니다. MQI 호출이 진행 중인 경우 리턴 코드 양식으로 된 알림이 수신됩니다.

큐 관리자의 무결성은 보존됩니다.

참고: 별도의 프로세스로 CLWL 엑시트를 실행하면 성능에 영향을 줄 수 있습니다.

FAST
큐 관리자 프로세스에서 클러스터 엑시트 인라인을 실행합니다.

이 옵션을 지정하면 SAFE 모드 실행과 연관된 프로세스 전환 비용이 발생하는 것을 방지하여 성능을 향상시키지만 큐 관리자 무결성은 손상됩니다. CLWL 엑시트에 문제점이 **없음**을 확신하고 성능에 대해 특히 걱정되는 경우에만 CLWL 엑시트를 FAST 모드로 실행해야 합니다.

CLWL 엑시트를 FAST 모드로 실행하는 동안 문제점이 발생하면 큐 관리자는 실패하고 큐 관리자의 무결성이 손상될 위험이 있습니다.

서브풀

이 스탠자는 IBM MQ에 의해 작성됩니다. 변경하지 마십시오.

Subpool 스탠자 및 해당 스탠자 내 ShortSubpoolName 속성은 큐 관리자를 작성할 때 IBM MQ에 의해 자동으로 기록됩니다. IBM MQ가 ShortSubpoolName에 대한 값을 선택합니다. 이 값을 변경하지 마십시오.

이름은 /var/mqm/sockets 디렉토리 안에 작성되는 디렉토리 및 기호 링크에 대응하며, IBM MQ는 실행 중인 프로세스 사이의 내부 통신에 사용합니다.

IBM i Linux UNIX qm.ini 파일의 Filesystem 스탠자

오류 로그 파일에 설정된 기본 권한은 대부분의 상황에서 유용할 것으로 예상되므로 대부분의 IBM MQ 관리자가 변경할 필요가 없습니다.

그러나 IBM MQ 관리자는 오류 로그 파일에 대한 권한을 변경할 수 있습니다. 이 경우에는 Filesystem 스탠자 옵션 **ValidateAuth**=*No*를 설정해야 하며, 이 경우 큐 관리자는 이후에 변경되지 않은 권한을 그대로 둘 수 있습니다.

기본 동작(**ValidateAuth**=*No*)은 큐 관리자가 큐 관리자 오류 로그의 파일 권한을 확인하고 이를 기본값으로 다시 변경하는 것입니다. 이 확인은 큐 관리자 종료 또는 시작 조작을 포함하여 언제든지 발생할 수 있습니다.

보안

qm.ini 파일의 Security 스탠자를 사용하여 오브젝트 권한 관리자(OAM)에 대한 옵션을 지정합니다.

ClusterQueueAccessControl=RQMName|Xmitq
클러스터 큐 관리자에서 호스팅되는 완전한 큐 또는 클러스터 큐의 액세스 제어를 검사하기 위해 이 속성을 설정합니다.

RQMName
원격으로 호스팅되는 큐의 액세스 제어에 대해 검사된 프로파일을 큐 또는 큐 관리자 프로파일이라고 합니다.

Xmitq
원격으로 호스팅되는 큐의 액세스 제어에 대해 검사된 프로파일이 SYSTEM.CLUSTER.TRANSMIT.QUEUE로 해석됩니다.

Xmitq가 기본값입니다.

GroupModel=GlobalGroups
이 속성은 Windows에서 사용자의 그룹 멤버십을 판별할 때 OAM이 글로벌 그룹을 검사하는지 여부를 판별합니다.

기본값은 글로벌 그룹을 검사하지 않는 것입니다.

GlobalGroups
OAM이 글로벌 그룹을 검사합니다.

GlobalGroups가 설정된 경우 권한 명령 **setmqaut**, **dspmqaut** 및 **dmpmqaut**가 글로벌 그룹 이름을 승인합니다. **setmqaut -g** 매개변수를 참조하십시오.

참고: ClusterQueueAcessControl=RQMName을 설정하고 MQZAS_VERSION_6 이전 권한 서비스의 사용자 정의 구현이 있으면 큐 관리자가 시작되지 않습니다. 이 경우 ClusterQueueAcessControl=Xmitq를 설정하거나 사용자 정의 권한 서비스를 MQZAS_VERSION_6 이상으로 업그레이드하십시오.

V 9.0.5 성능 조정 매개변수

큐 관리자의 성능 조정을 위해 옵션을 지정하려면 qm.ini 파일에서 TuningParameters 스탠자를 사용하십시오.

ImplSyncOpenOutput=*value*
: **ImplSyncOpenOutput**은 동기점 외부에서 지속적 넣기(PUT)에 내재된 동기점을 사용할 수 있기 전에 넣기(PUT)를 위해 큐가 열려 있는 최소 애플리케이션 수입니다. **ImplSyncOpenOutput**의 기본값은 2입니다.

  이는 넣기(PUT) 조작을 위해 큐가 열려 있는 애플리케이션이 하나일 경우 **ImplSyncOpenOutput**이 꺼짐으로 전환되는 효과가 있습니다.

  ImplSyncOpenOutput=1을 지정하면 내재된 동기화가 항상 고려됩니다.

  양의 정수 값을 설정할 수 있습니다. 내재된 동기점을 추가하지 않으려면 ImplSyncOpenOutput=OFF로 설정하십시오.

V 9.0.0.12 **OAMLdapConnectTimeout=최대 시간**
: 서버에 대한 TCP 연결을 설정하기 위해 LDAP 클라이언트가 대기하는 최대 시간(초)입니다. 연결 이름 목록을 통해 여러 LDAP 서버를 제공하는 경우 제한시간이 각 개별 연결 시도에 적용되므로 이 제한시간에 도달하면 이름 목록의 다음 항목에 대한 연결이 시도됩니다.

V 9.0.0.12 **OAMLdapQueryTimeLimit=최대 시간**
: 연결이 설정되고 LDAP 요청이 전송되면 LDAP 클라이언트가 서버로부터 LDAP 요청에 대한 응답을 수신하기를 기다리는 최대 시간(초)입니다.

관련 정보

암시적 동기점

분산 큐잉 구성

이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

시작하기 전에

이 섹션을 읽기 전에 분산 큐잉 및 클러스터에 소개된 채널, 큐 및 기타 개념을 이해하는 것이 좋습니다.

프로시저

- 다음 하위 주제의 정보를 활용하여 분산 큐잉을 사용하는 애플리케이션을 연결하십시오.
  - 129 페이지의 『IBM MQ 분산 큐잉 기술』
  - 147 페이지의 『분산 큐 관리 소개』
  - 149 페이지의 『다른 큐 관리자로 메시지를 보내는 방법』
  - 169 페이지의 『채널 트리거』
  - 167 페이지의 『메시지의 안전성』
  - ULW 175 페이지의 『UNIX, Linux, and Windows에서 채널 모니터링 및 제어』
  - IBM i 196 페이지의 『IBM i에서 채널 모니터링 및 제어』

관련 개념

597 페이지의 『IBM MQ for z/OS 설정』
이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

관련 태스크

14 페이지의 『서버와 클라이언트 간의 연결 구성』
IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

216 페이지의 『큐 관리자 클러스터 구성』
클러스터는 초기 구성 및 진행 중인 관리를 모두 단순화하는 방식으로 큐 관리자를 상호 연결하기 위한 메커니즘을 제공합니다. 클러스터 컴포넌트를 정의하고 클러스터를 작성 및 관리할 수 있습니다.

78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』
설치 요구에 맞게 IBM MQ 또는 개별 큐 관리자의 작동을 변경할 수 있습니다.

592 페이지의 『z/OS에서 큐 관리자 구성』
IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

IBM MQ 분산 큐잉 기술

이 절의 하위 주제에서는 채널 계획 시 사용되는 기술을 설명합니다. 이 하위 주제에서는 큐 관리자와 함께 연결하는 방법을 계획하고 애플리케이션 간 메시지 플로우를 관리하는 데 도움이 되는 기술을 설명합니다.

메시지 채널 계획 예는 다음을 참조하십시오.

- ULW UNIX, Linux, and Windows의 메시지 채널 계획 예
- IBM i IBM i의 메시지 채널 계획 예
- z/OS z/OS의 메시지 채널 계획 예
- z/OS 큐 공유 그룹을 사용하는 z/OS의 메시지 채널 계획 예

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

관련 정보

채널
메시지 큐잉 소개
분산 큐잉 및 클러스터
구성 정보 예

메시지 플로우 제어

메시지 플로우 제어는 큐 관리자 사이의 메시지 라우트 설정 및 유지보수를 처리하는 태스크로, 많은 큐 관리자를 통해 멀티호핑하는 라우트에 중요합니다. 이 절에서는 메시지 플로우를 제어하기 위해 시스템에서 큐, 알리어스 큐 정의, 메시지 채널을 사용하는 방법을 설명합니다.

128 페이지의 『분산 큐잉 구성』에 소개되어 있는 여러 기술을 사용하여 메시지 플로우를 제어할 수 있습니다. 큐 관리자가 클러스터에 있으면 다른 기술을 사용하여 메시지 플로우가 제어됩니다(129 페이지의 『메시지 플로우 제어』의 설명 참조). z/OS 큐 관리자가 큐 공유 그룹에 있고 그룹 내 큐잉(IGQ)을 사용할 수 있으면 IGQ 에이전트로 메시지 플로우를 제어할 수 있습니다. 이러한 에이전트는 그룹 내 큐잉에 설명되어 있습니다.

메시지 플로우를 제어하기 위해 사용할 수 있는 오브젝트는 다음과 같습니다.

- 전송 큐
- 메시지 채널
- 리모트 큐 정의
- 큐 관리자 알리어스 정의
- 응답 대상 큐 알리어스 정의

큐 관리자 및 큐 오브젝트는 오브젝트 유형에 설명되어 있습니다. 메시지 채널은 분산 큐잉 컴포넌트에 설명되어 있습니다. 다음 기술은 위의 오브젝트를 사용하여 시스템에서 메시지 플로우를 작성합니다.

- 리모트 큐에 메시지 넣기
- 특정 전송 큐를 통한 라우팅
- 메시지 수신
- 사용자 시스템을 통한 메시지 전달
- 메시지 플로우 분리
- 메시지 플로우를 다른 목적지로 전환
- 응답 대상 큐 이름을 알리어스 이름으로 해석

참고

이 절에서 설명하는 모든 개념은 네트워크의 모든 노드와 관련이 있으며 메시지 채널의 송신측과 수신측을 포함합니다. 따라서 대부분의 예에는 하나의 노드만 나와 있습니다. 예외는 메시지 채널의 다른 측에서 관리자의 명시적인 협업이 필요한 경우입니다.

개별 기술에 대한 내용으로 진행하기 전에 이름 분석의 개념과 리모트 큐 정의를 사용하는 세 가지 방법에 대한 기본 사항을 정리하는 것이 좋습니다. 분산 큐잉 및 클러스터를 참조하십시오.

관련 개념

130 페이지의 『전송 헤더의 큐 이름』
목적지 큐 이름은 목적지 큐에 도달할 때까지 전송 헤더에서 메시지와 함께 이동합니다.

130 페이지의 『큐 관리자 및 응답 대상 알리어스 작성 방법』
이 주제는 리모트 큐 정의를 작성할 수 있는 세 가지 방법을 설명합니다.

전송 헤더의 큐 이름

목적지 큐 이름은 목적지 큐에 도달할 때까지 전송 헤더에서 메시지와 함께 이동합니다.

애플리케이션이 사용하는 큐 이름인 논리적 큐 이름은 큐 관리자가 목적지 큐 이름으로 해석합니다(즉, 물리적 큐 이름). 이 목적지 큐 이름은 목적지 큐에 도달할 때까지 개별 데이터 영역인 전송 헤더에서 메시지와 함께 이동합니다. 그런 다음 전송 헤더가 제거됩니다.

서비스의 병렬 클래스를 작성할 때 이 큐 이름의 큐 관리자 파트를 변경합니다. 서비스 클래스 전환 끝에 도달하면 큐 관리자 이름을 원래 이름으로 되돌려야 합니다.

큐 관리자 및 응답 대상 알리어스 작성 방법

이 주제는 리모트 큐 정의를 작성할 수 있는 세 가지 방법을 설명합니다.

리모트 큐 정의 오브젝트는 세 가지 다른 방법으로 사용됩니다. 131 페이지의 표 14은(는) 세 가지 방법 각각을 정의하는 방법을 설명합니다.

- 리모트 큐 정의를 사용하여 로컬 큐 이름 재정의.

  애플리케이션이 큐를 열 때 큐 이름만 제공하며 이 큐 이름은 리모트 큐 정의의 이름입니다.

  리모트 큐 정의에는 대상 큐와 큐 관리자의 이름이 포함됩니다. 선택적으로, 이 정의에는 사용될 전송 큐의 이름이 포함됩니다. 전송 큐 이름이 제공되지 않는 경우에는 큐 관리자가 전송 큐 이름에 대해 리모트 큐 정의에서 가져온 큐 관리자 이름을 사용합니다. 이 이름의 전송 큐가 정의되지 않지만 기본 전송 큐가 정의된 경우에는 기본 전송 큐를 사용합니다.
- 리모트 큐 정의를 사용하여 큐 관리자 이름 재정의.

  애플리케이션 또는 채널 프로그램은 큐를 열 때 리모트 큐 관리자 이름과 함께 큐 이름을 제공합니다.

  큐 관리자 이름과 동일한 이름을 리모트 큐 정의에 제공하고 정의의 큐 이름을 공백으로 남겨둔 경우에는 큐 관리자가 열린 호출의 큐 관리자 이름을 정의의 큐 관리자 이름으로 대체합니다.

또한 정의에는 사용될 전송 큐의 이름이 포함될 수 있습니다. 전송 큐 이름이 제공되지 않는 경우에는 큐 관리자가 전송 큐 이름에 대해 리모트 큐 정의에서 가져온 큐 관리자 이름을 사용합니다. 이 이름의 전송 큐가 정의되지 않지만 기본 전송 큐가 정의된 경우에는 기본 전송 큐를 사용합니다.

• 리모트 큐 정의를 사용하여 응답 대상 큐 이름 재정의.

  애플리케이션이 큐에 메시지를 둘 때마다 응답 메시지에 대한 응답 대상 큐 이름을 제공할 수 있지만 큐 관리자 이름은 공백으로 둡니다.

  응답 대상 큐와 동일한 이름을 가진 리모트 큐 정의를 제공하는 경우, 로컬 큐 관리자는 응답 대상 큐 이름을 정의의 큐 이름으로 바꿉니다.

  정의에서 큐 관리자 이름은 제공할 수 있지만 전송 큐 이름은 제공할 수 없습니다.

표 *14.* 리모트 큐 정의 오브젝트를 사용하는 세 가지 방법

사용법	**큐 관리자 이름**	**큐 이름**	**전송 큐 이름**
1. 리모트 큐 정의(OPEN 호출 시)			
호출에 제공	공백 또는 로컬 QM	(*) 필수	해당 사항 없음
정의에 제공	필수	필수	선택사항
2. 큐 관리자 알리어스(OPEN 호출 시)			
호출에 제공	(*) 필수 및 비로컬 QM	필수	해당 사항 없음
정의에 제공	필수	공백	선택사항
3. 응답 대상 큐 알리어스(PUT 호출 시)			
호출에 제공	공백	(*) 필수	해당 사항 없음
정의에 제공	선택사항	선택사항	공백

참고: (*)는 이 이름이 정의 오브젝트의 이름임을 의미

정식 설명은 큐 이름 해석을 참조하십시오.

리모트 큐에 메시지 넣기

리모트 큐 정의 오브젝트를 사용하여 큐 이름을 인접 큐 관리자의 전송 큐로 해석할 수 있습니다.

분산 큐잉 환경에서 전송 큐와 채널은 메시지가 로컬 시스템에 있는 애플리케이션에서 수신되거나 또는 인접 시스템으로부터 채널을 통해 도착하거나 한 위치로의 모든 메시지에 대한 중심점입니다. 132 페이지의 그림 7에서는 논리적 큐 'QA_norm'에 메시지를 넣는 애플리케이션을 보여줍니다. 이름 해석은 리모트 큐 정의 'QA_norm'을 사용하여 전송 큐 QMB를 선택합니다. 그런 다음 'QA_norm at QMB' 메시지에 전송 헤더를 추가합니다.

예를 들어, 'Channel_back'의 인접 시스템에서 도착하는 메시지의 전송 헤더에는 물리적 큐 이름 'QA_norm at QMB'가 포함됩니다. 이러한 메시지는 변경되지 않은 채로 전송 큐 QMB에 놓여집니다.

채널은 메시지를 인접 큐 관리자로 이동합니다.

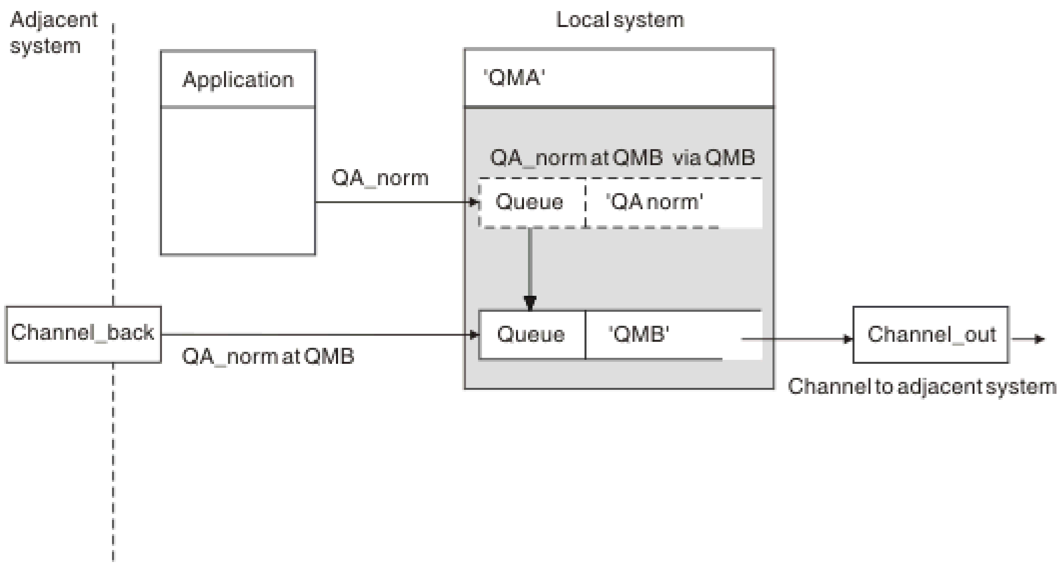


그림 7. 리모트 큐 정의는 큐 이름을 인접 큐 관리자의 전송 큐로 해석하는 데 사용됩니다.

IBM MQ 시스템 관리자는 다음을 수행해야 합니다.

- 인접 시스템으로부터의 메시지 채널 정의
- 인접 시스템으로의 메시지 채널 정의
- 전송 큐 QMB 작성
- 애플리케이션이 사용하는 큐 이름을 목적지 큐 이름, 목적지 큐 관리자 이름, 전송 큐 이름으로 해석하려면 리모트 큐 오브젝트 'QA_norm' 정의

클러스터링 환경의 경우, 로컬 큐 관리자에서만 클러스터 수신자 채널을 정의하면 됩니다. 전송 큐 또는 리모트 큐 오브젝트는 정의하지 않아도 됩니다. 클러스터를 참조하십시오.

이름 해석 정보

리모트 큐를 정의하면 물리적 목적지 큐 이름과 큐 관리자 이름이 정의됩니다. 이러한 이름은 메시지의 전송 헤더에 놓입니다.

인접 시스템으로부터의 수신 메시지는 원래 큐 관리자가 수행한 이름 분석 유형을 이미 갖고 있습니다. 따라서 전송 헤더에 물리적 목적지 큐 이름과 큐 관리자 이름이 표시됩니다. 이러한 메시지는 리모트 큐 정의의 영향을 받지 않습니다.

전송 큐 선택

리모트 큐 정의를 사용하여 다른 전송 큐에서 동일한 인접 큐 관리자로 메시지를 보내도록 허용할 수 있습니다.

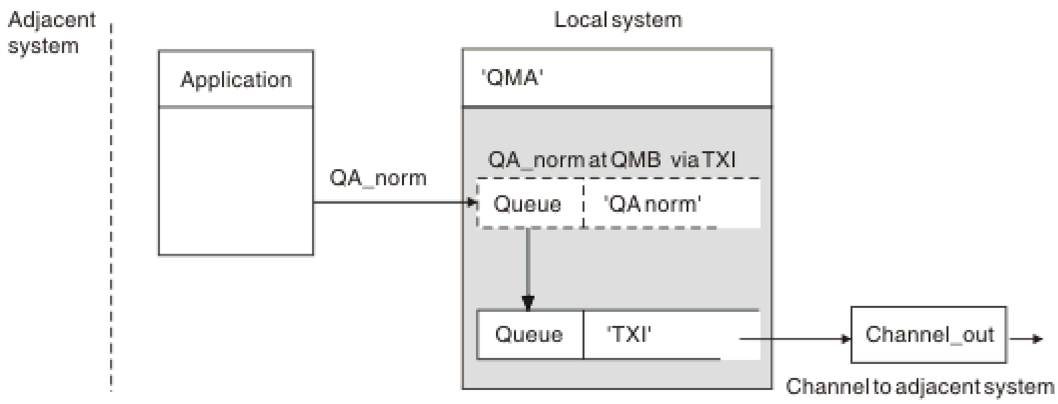


그림 *8.* 리모트 큐 정의에서는 다른 전송 큐를 사용할 수 있습니다.

분산 큐잉 환경에서 한 채널에서 다른 채널로 메시지 흐름을 변경해야 하는 경우 131 페이지의 『리모트 큐에 메시지 넣기』의 132 페이지의 그림 7과(와) 동일한 시스템 구성을 사용합니다. 이 주제의 133 페이지의 그림 8은(는) 원격 큐 정의를 사용하여 다른 전송 큐를 통해, 따라서 다른 채널을 통해 동일한 인접 큐 관리자에게 메시지를 보내는 방법을 보여줍니다.

133 페이지의 그림 8에 표시된 구성의 경우, 리모트 큐 오브젝트 'QA_norm'과 전송 큐 'TX1'을 제공해야 합니다. 리모트 큐 관리자의 'QA_norm' 큐, 전송 큐 'TX1', 큐 관리자 'QMB_priority'를 선택하려면 'QA_norm'을 제공해야 합니다. 시스템에 인접한 채널의 정의에 'TX1'을 지정하십시오.

메시지는 전송 헤더에 'QA_norm at QMB_priority'가 포함되는 전송 큐 'TX1'에 배치되며 채널을 통해 인접 시스템으로 보내집니다.

이 예에서는 큐 관리자 알리어스가 필요하므로 channel_back이 제외되었습니다.

클러스터링 환경에서는 전송 큐 또는 리모트 큐 정의를 정의하지 않아도 됩니다. 추가 정보는 216 페이지의 『클러스터 큐 정의』의 내용을 참조하십시오.

메시지 수신

큐 관리자가 다른 큐 관리자로부터 메시지를 수신하도록 구성할 수 있습니다. 의도하지 않은 이름 해석이 발생하지 않는지 확인해야 합니다.

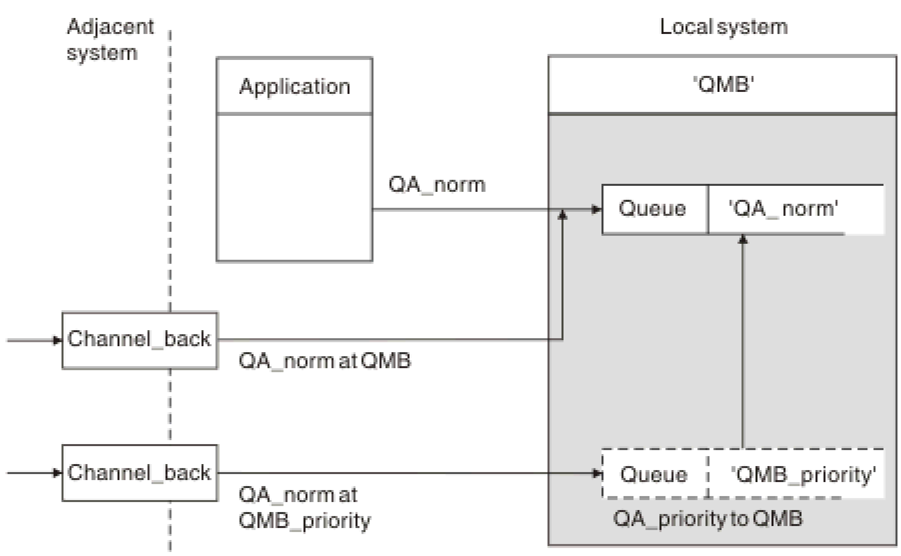


그림 *9.* 직접적인 메시지 수신과 알리어스 큐 관리자 이름 해석

시스템 관리자는 송신할 메시지에 대한 정렬은 물론, 인접 큐 관리자로부터 수신할 메시지도 정렬해야 합니다. 수신된 메시지의 전송 헤더에는 목적지 큐 관리자 및 큐의 물리적 이름이 포함됩니다. 이 메시지는 큐 관리자 이름

과 큐 이름을 둘 다 지정하는 로컬 애플리케이션으로부터의 메시지와 동일한 것으로 간주됩니다. 따라서 시스템에 들어가는 메시지에 고의적이지 않은 이름 해석이 수행되지 않는지 확인해야 합니다. 이 시나리오에 대해서는 133 페이지의 그림 9의 내용을 참조하십시오.

이 구성의 경우 다음 준비가 필요합니다.

- 인접 큐 관리자로부터 메시지를 수신할 메시지 채널
- 수신 메시지 플로우, 'QMB_priority'를 로컬 큐 관리자 이름, 'QMB'로 해석하기 위한 큐 관리자 알리어스 정의
- 로컬 큐, 'QA_norm'(존재하지 않는 경우)

알리어스 큐 관리자 이름 수신

이 예에서 큐 관리자 알리어스 정의 사용 시 다른 목적지 큐 관리자를 선택하지 않았습니다. 이 로컬 큐 관리자를 통해 전달되고 'QMB_priority'에 주소 지정된 메시지는 큐 관리자 'QMB'용입니다. 알리어스 큐 관리자 이름은 별도의 메시지 플로우를 작성하는 데 사용됩니다.

사용자 시스템을 통한 메시지 전달

세 가지 방법(위치 이름 사용, 큐 관리자의 알리어스 사용 또는 전송 큐 선택)으로 사용자 시스템을 통해 메시지를 전달할 수 있습니다.

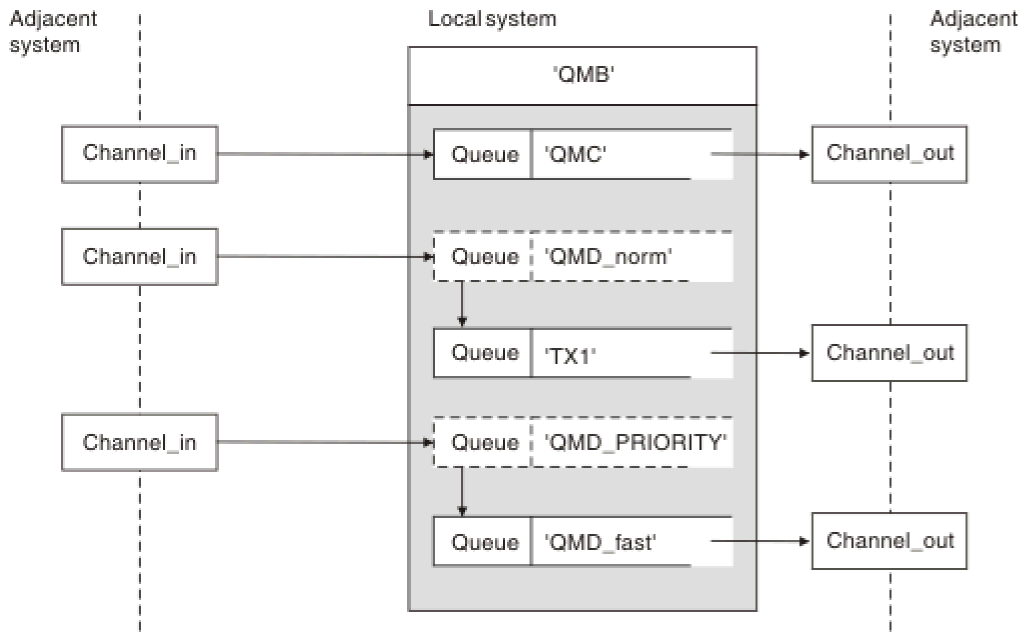


그림 *10.* 시스템을 통한 세 가지 메시지 전달 방법

133 페이지의 『메시지 수신』의 133 페이지의 그림 9에 표시된 기술은 별명 플로우가 캡처되는 방법을 보여줍니다. 134 페이지의 그림 10은(는) 이전에 설명한 기술을 결합하여 네트워크가 구성되는 방법을 보여줍니다.

구성은 목적지가 다른 다음 세 개 메시지를 전달하는 채널을 보여줍니다.

1. QMC의 QB
2. QMD_norm의 QB
3. QMD_PRIORITY의 QB

첫 번째 메시지 플로우는 변경되지 않은 시스템을 통해 전달해야 합니다. 두 번째 메시지 플로우는 다른 전송 큐와 채널을 통해 전달해야 합니다. 두 번째 메시지 플로우의 경우, 또한 알리어스 큐 관리자 이름 QMD_norm에 대한 메시지를 큐 관리자 QMD로 해석해야 합니다. 세 번째 메시지 플로우는 다른 변경사항 없이 다른 전송 큐를 선택합니다.

클러스터링 환경에서는 클러스터 전송 큐를 통해 메시지가 전달됩니다. 일반적으로 단일 전송 큐 SYSTEM.CLUSTER.TRANSMIT.queue는 큐 관리자가 구성원인 모든 클러스터의 모든 큐 관리자에게 모든 메시지를 전송합니다. 큐 관리자 클러스터의 내용을 참조하십시오. 큐 관리자가 멤버인 클러스터에 속하는 일부 또는 모든 큐 관리자에 개별 전송 큐를 정의할 수 있습니다.

다음 방법은 분산 큐잉 환경에 적용 가능한 기술을 설명합니다.

다음 방법 사용

이러한 구성의 경우 다음을 준비해야 합니다.

- 입력 채널 정의
- 출력 채널 정의
- 전송 큐:
  - QMC
  - TX1
  - QMD_fast
- 큐 관리자 알리어스 정의:
  - TX1을 통한 QMD_norm에서 QMD까지의 QMD_norm
  - QMD_fast를 통한 QMD_PRIORITY에서 QMD_PRIORITY까지의 QMD_PRIORITY

참고: 이 예에 표시된 어떤 메시지 플로우도 목적지 큐를 변경하지 않습니다. 큐 관리자 이름 알리어스는 메시지 플로우를 분리하기 위해 제공됩니다.

방법 1: 수신 위치 이름 사용

다른 위치 이름(예: QMC)을 포함하는 전송 헤더로 메시지를 수신하려고 합니다. 가장 단순한 구성은 해당 이름, QMC를 사용하는 전송 큐를 작성하는 것입니다. 전송 큐를 지원하는 채널은 메시지를 변경 없이 다음 목적지로 전달합니다.

방법 2: 큐 관리자 알리어스 사용

두 번째 방법은 큐 관리자 알리어스 오브젝트 정의를 사용하되 새 위치 이름, QMD와 특정 전송 큐, TX1을 지정하는 것입니다. 이 조치는 다음과 같이 수행됩니다.

- 큐 관리자 이름 알리어스 QMD_norm(즉, 이름 지정된 서비스 클래스 QMD_norm)으로 설정된 알리어스 메시지 플로우를 종료합니다.
- 이 메시지의 전송 헤더를 QMD_norm에서 QMD로 변경합니다.

방법 3: 전송 큐 선택

세 번째 방법은 목적지 위치, QMD_PRIORITY와 동일한 이름으로 큐 관리자 알리어스 오브젝트를 정의하는 것입니다. 특정 전송 큐, QMD_fast와 그에 따라 다른 채널을 선택하려면 큐 관리자 알리어스 정의를 사용하십시오. 이러한 메시지의 전송 헤더는 변경되지 않습니다.

메시지 플로우 분리

큐 관리자 알리어스를 사용하여 동일한 큐 관리자로 메시지를 보내는 별도 메시지 플로우를 작성할 수 있습니다.

분산 큐잉 환경에서 여러 가지 이유로 인해 동일한 큐 관리자에 대한 메시지를 서로 다른 메시지 플로우로 분리해야 할 수 있습니다. 예를 들면 다음과 같습니다.

- 대형, 중형, 소형 메시지에 개별 플로우를 제공해야 할 수 있습니다. 이 요구 사항은 또한 클러스터링 환경에서 적용되며 이 경우 중첩되는 클러스터를 작성할 수 있습니다. 이렇게 해야 하는 이유는 다음과 같은 여러 가지가 있습니다.

- 서로 다른 조직이 각자 고유의 관리를 할 수 있도록 하기 위해
- 독립 애플리케이션을 개별적으로 관리할 수 있도록 하기 위해
- 서비스 클래스를 작성하기 위해. 예를 들어, STUDENTS 클러스터의 서브세트인 STAFF 클러스터가 있을 수 있습니다. STAFF 클러스터에서 알려진 큐에 메시지를 넣을 때는 제한된 채널이 사용됩니다. STUDENTS 클러스터에서 알려진 큐에 메시지를 넣을 때는 일반 채널 또는 제한된 채널을 사용할 수 있습니다.
- 테스트 환경 및 프로덕션 환경을 생성하기 위해

- 로컬로 생성된 메시지의 경로에서 다른 경로가 수신 메시지를 라우트해야 할 수 있습니다.
- 설치에서 특정 시점(예를 들어, 야간)에 메시지 이동을 스케줄링해야 할 수 있으며 이 경우 메시지는 스케줄링될 때까지 예약된 큐에 저장되어야 합니다.

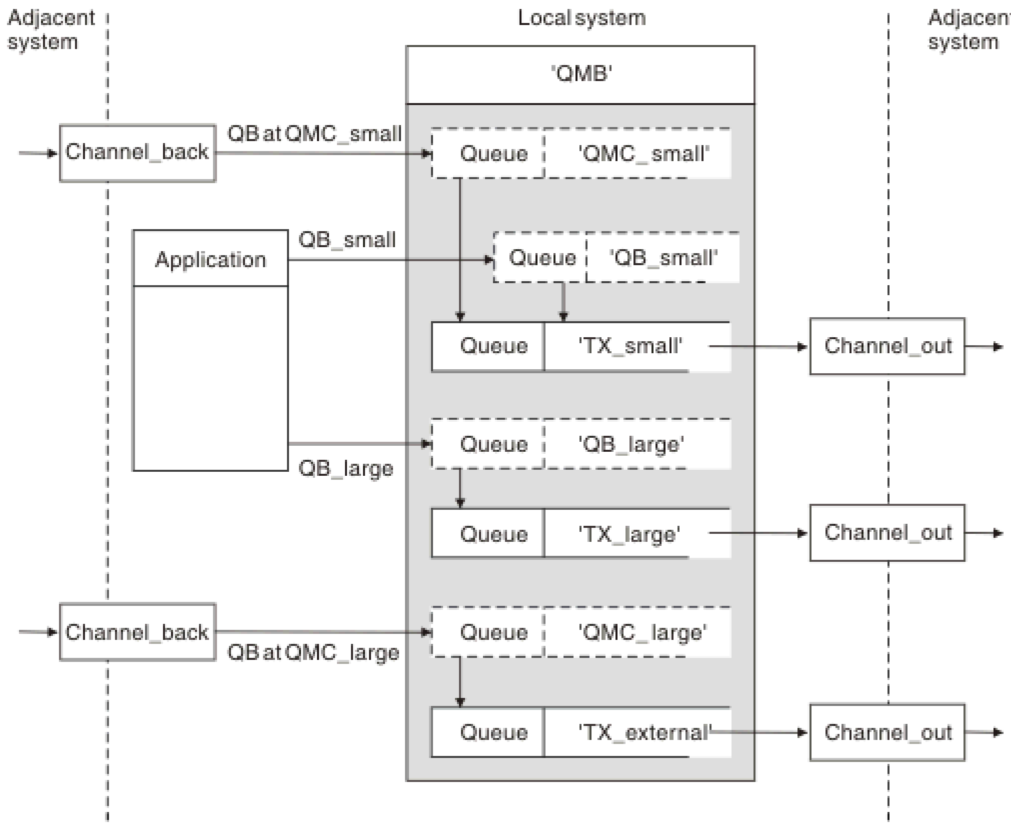


그림 *11.* 메시지 플로우 분리

136 페이지의 그림 11에 표시된 예에는 알리어스 큐 관리자 이름 'QMC_small' 및 'QMC_large'로의 두 개 수신 플로우가 있습니다. 이러한 플로우에는 로컬 큐 관리자를 위해 플로우를 캡처하기 위한 큐 관리자 알리어스 정의를 제공합니다. 두 개 리모트 큐에 대한 주소를 지정하는 애플리케이션이 있으며 이 메시지 플로우를 별도로 유지해야 합니다. 동일한 위치, 'QMC'를 지정하지만 다른 전송 큐를 지정하는 두 가지 리모트 큐 정의를 제공합니다. 이 정의는 플로우를 개별적으로 유지하므로 다른 한쪽에서 추가 요소가 필요하지 않습니다. 전송 헤더에 동일한 목적지 큐 관리자 이름이 있기 때문입니다. 다음을 제공합니다.

- 수신 채널 정의
- 두 개의 리모트 큐 정의(QB_small과 QB_large)
- 두 개의 큐 관리자 알리어스 정의(QMC_small과 QMC_large)
- 세 개의 송신 채널 정의
- 세 개의 전송 큐: TX_small, TX_large, TX_external

인접 시스템과의 조정

큐 관리자 알리어스를 사용하여 별도 메시지 플로우를 작성하는 경우, 메시지 채널의 원격 끝 시스템 관리자와 함께 이 활동을 조정하여 해당 큐 관리자 알리어스를 사용할 수 있도록 해야 합니다.

다양한 위치로 분산될 메시지 집중

다양한 위치를 목적지로 하는 메시지를 단일 채널에 집중시킬 수 있습니다.

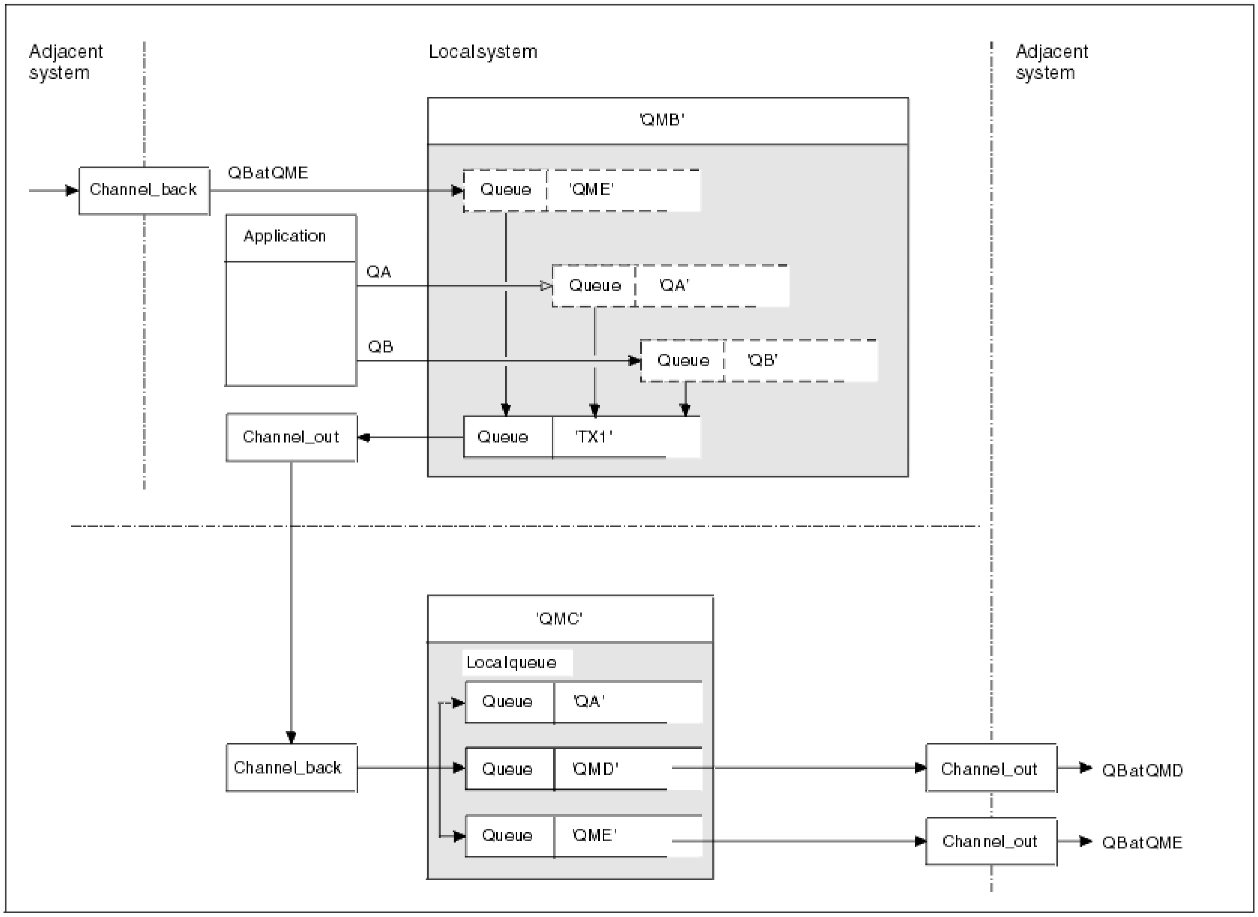


그림 *12.* 채널에 메시지 플로우 결합

137 페이지의 그림 12에서는 다양한 위치를 목적지로 하는 메시지를 단일 채널로 집중시키기 위한 분산 큐잉 기법을 보여줍니다. 두 가지 가능한 방법은 다음과 같습니다.

- 게이트웨이를 통해 메시지 트래픽 집중
- 노드 간 광대역 하이웨이 사용

이 예에서는 다른 목적지 큐와 큐 관리자가 다른 로컬 및 인접 소스의 메시지가 전송 큐 'TX1'을 통해 큐 관리자 QMC로 이동합니다. 큐 관리자 QMC는 목적지에 따라 메시지를 전달합니다. 한 메시지는 큐 관리자 QMD로 전송되는 전송 큐 'QMD'로 설정되고 또 다른 메시지는 큐 관리자 QME로 전송되는 전송 큐 'QME'로 설정됩니다. 다른 메시지는 로컬 큐 'QA'에 넣습니다.

다음을 제공해야 합니다.

- 채널 정의
- 전송 큐 TX1
- 리모트 큐 정의:
  - QA - 'TX1을 통한 QMC의 QA'
  - QB - 'TX1을 통한 QMD의 QB'
- 큐 관리자 알리어스 정의:

– QME - 'TX1을 통한 QME'

QMC를 구성하는 보조 관리자는 다음을 제공해야 합니다.

- 채널 이름이 같은 수신 채널 정의
- 연관된 전송 채널 정의가 있는 전송 큐 QMD
- 연관된 전송 채널 정의가 있는 전송 큐 QME
- 로컬 큐 오브젝트 QA

메시지 플로우를 다른 목적지로 전환

큐 관리자 알리어스와 전송 큐를 사용하여 특정 메시지의 목적지를 재정의할 수 있습니다.

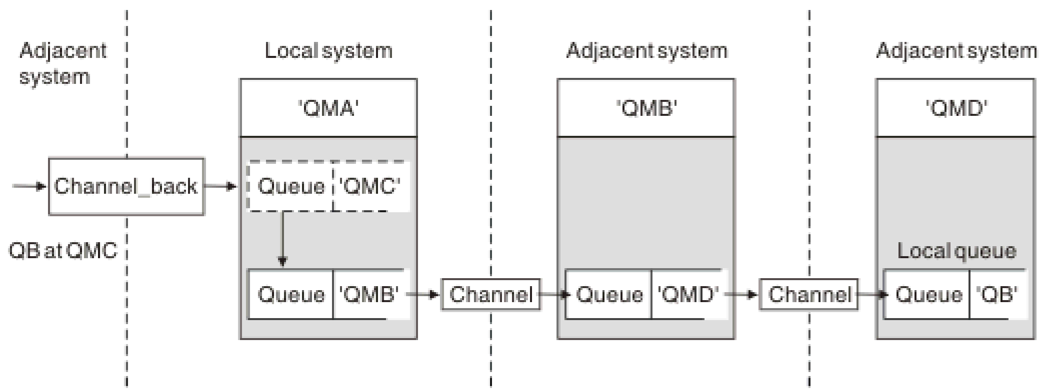


그림 *13.* 메시지 스트림을 다른 목적지로 전환

138 페이지의 그림 13에서는 특정 메시지의 목적지를 재정의할 수 있는 방법을 보여줍니다. QMA에 대한 수신 메시지는 'QMC의 QB'용으로 지정됩니다. 이러한 메시지는 일반적으로 QMA에 도달하며 QMC 채널의 일부인 전송 큐 QMC에 배치됩니다. QMA는 QMD로 메시지를 보내야 하지만 이 경우 QMB를 통해서만 QMD에 도달할 수 있습니다. 이 방법은 한 위치에서 다른 위치로 서비스를 이동해야 하고 메시지를 새 주소로 조정할 때까지 신청자에게 임시로 메시지 송신을 계속하도록 허용해야 할 때 유용합니다.

특정 큐 관리자를 수신지로 하여 들어오는 메시지를 다른 큐 관리자로 다시 라우팅하는 메소드는 다음이 사용됩니다.

- 목적지 큐 관리자를 다른 큐 관리자로 변경하고, 인접 시스템으로 향하는 전송 큐를 선택하기 위해 큐 관리자 알리어스 사용
- 인접 큐 관리자에게 서비스를 제공하기 위해 전송 큐 사용
- 목적지 큐 관리자로의 지속적인 라우팅을 위해 인접 큐 관리자에 있는 전송 큐 사용

다음을 제공해야 합니다.

- Channel_back 정의
- QMB을 통한 QMD에 있는 QB을 가진 큐 관리자 별칭 오브젝트 정의 QMC
- Channel_out 정의
- 연관된 전송 큐 QMB

QMB를 구성하는 보조 관리자는 다음을 제공해야 합니다.

- 해당 channel_back 정의
- 전송 큐, QMD
- QMD에 대한 관련 채널 정의

클러스터링 환경에서 알리어스를 사용할 수 있습니다. 관련 정보는 304 페이지의 『큐 관리자 알리어스 및 클러스터』의 내용을 참조하십시오.

분배 목록으로 메시지 송신

단일 MQPUT 호출을 사용하여 애플리케이션이 여러 대상으로 메시지를 보내도록 할 수 있습니다.

z/OS를 제외한 모든 플랫폼의 IBM MQ의 경우, 애플리케이션이 단일 MQPUT 호출로 여러 대상에 메시지를 보낼 수 있습니다. 이 기능은 분산 큐잉 환경과 클러스터링 환경에서 모두 수행할 수 있습니다. 목적지는 분배 목록에 정의해야 합니다(분배 목록의 설명 참조).

모든 큐 관리자가 분배 목록을 지원하는 것은 아닙니다. MCA가 파트너와의 연결을 설정할 때 파트너가 분배 목록을 지원하는지 여부를 판별하며 그에 따라 전송 큐에 플래그를 설정합니다. 애플리케이션이 분배 목록을 목적지로 하는 메시지를 보내려고 시도하지만 파트너가 분배 목록을 지원하지 않는 경우에는 송신 MCA가 메시지를 가로채 정해진 각 목적지에 맞는 전송 큐에 메시지를 한번 넣습니다.

수신 MCA는 분배 목록으로 보낸 메시지가 지정된 모든 목적지에서 안전하게 수신되는지 확인합니다. 목적지가 실패하면 MCA가 실패한 목적지를 설정합니다. 그런 다음 해당 목적지에 대한 예외 보고서를 생성하고 다시 메시지를 보내도록 시도할 수 있습니다.

응답 대상 큐

응답 대상 큐를 사용하여 전체 리모트 큐 처리 루프를 작성할 수 있습니다.

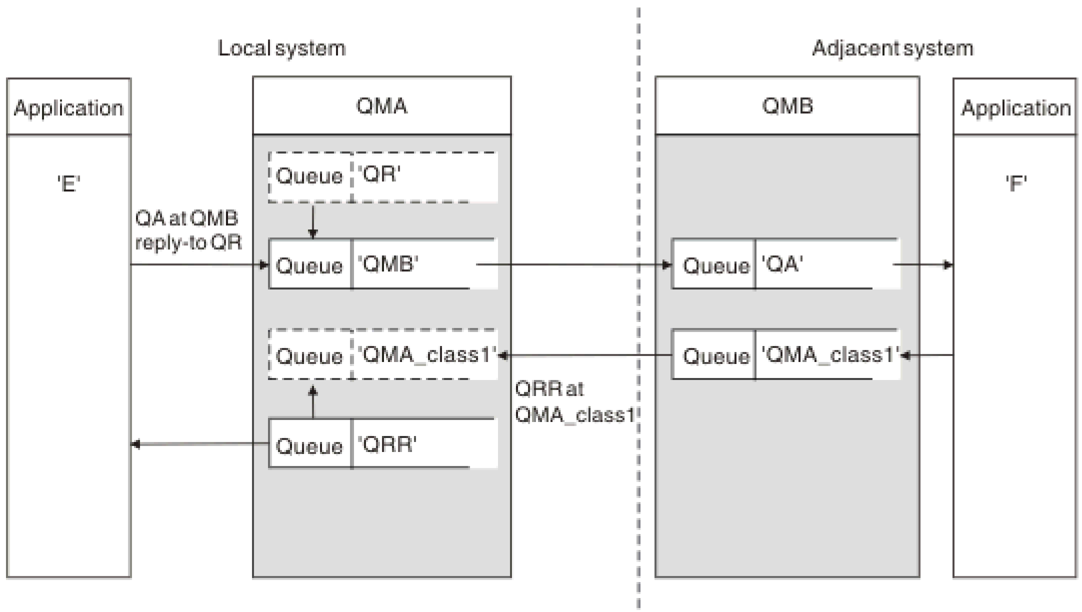


그림 *14. PUT* 호출 시 응답 대상 큐 이름 대체

응답 대상 큐를 사용하는 전체 리모트 큐 처리 루프가 139 페이지의 그림 14에 나와 있습니다. 이 루프는 분산 큐잉 환경과 클러스터링 환경에서 모두 적용됩니다. 자세한 내용은 145 페이지의 표 18에 나와 있습니다.

애플리케이션은 QMB에서 QA를 열고 해당 큐에 메시지를 넣습니다. 큐 관리자 이름이 지정되지 않고 응답 대상 큐 이름 QR이 메시지에 제공됩니다. 큐 관리자 QMA는 응답 대상 큐 오브젝트 QR을 찾고, 여기서 알리어스 이름 QRR과 큐 관리자 이름 QMA_class1을 추출합니다. 이러한 이름은 메시지의 응답 대상 필드에 들어갑니다.

QMB의 애플리케이션에서 들어온 응답 메시지는 QMA_class1의 QRR로 지정됩니다. 큐 관리자는 큐 관리자 알리어스 이름 정의 QMA_class1을 사용하여 메시지를 자신 또는 큐 QRR로 이동합니다.

이 시나리오는 응답 메시지에 대한 서비스 클래스를 선택하는 기능을 애플리케이션에 제공합니다. 이 클래스는 QMA의 큐 관리자 알리어스 정의, QMA_class1과 함께 QMB의 전송 큐 QMA_class1로 구현됩니다. 이 방식으로, 애플리케이션과 관계 없이 플로우가 분리될 수 있도록 애플리케이션의 응답 대상 큐를 변경할 수 있습니다. 애플리케이션은 이 특정 서비스 클래스에 항상 QR을 선택합니다. 응답 대상 큐 정의 QR로 서비스 클래스를 변경할 수 있습니다.

다음을 작성해야 합니다.

- 응답 대상 큐 정의 QR

- 전송 큐 오브젝트 QMB
- Channel_out 정의
- Channel_back 정의
- 큐 관리자 알리어스 정의 QMA_class1
- 로컬 큐 오브젝트가 존재하지 않을 경우, 로컬 큐 오브젝트 QRR

인접 시스템의 보조 관리자는 다음을 작성해야 합니다.

- 수신 채널 정의
- 전송 큐 오브젝트 QMA_class1
- 연관된 송신 채널
- 로컬 큐 오브젝트 QA

애플리케이션은 프로그램은 다음을 사용합니다.

- 넣기 호출에서 응답 대상 큐 이름 QR
- 가져오기 호출에서 큐 이름 QRR

이 방식을 사용하면 애플리케이션과 관계 없이 필요에 따라 서비스 클래스를 변경할 수 있습니다. 전송 큐 'QMA_class1' 및 큐 관리자 알리어스 'QMA_class1'과 함께 응답 대상 알리어스 'QR'을 변경할 수 있습니다.

큐에 메시지를 넣을 때 응답 대상 알리어스 오브젝트가 없으면 비어 있는 응답 대상 큐 관리자 이름 필드에 로컬 큐 관리자 이름이 삽입됩니다. 응답 대상 큐 이름은 변경되지 않습니다.

이름 해석 제한사항

이름 해석은 원래 메시지를 넣을 때 'QMA'의 응답 대상 큐에 대해 수행되므로 'QMB'에서는 더 이상 이름 해석이 허용되지 않습니다. 응답 애플리케이션이 응답 대상 큐의 실제 이름으로 메시지를 넣습니다.

애플리케이션은 응답 대상 큐에 사용하는 이름이 리턴 메시지를 찾게 될 실제 큐의 이름과 다르다는 사실을 알아야 합니다.

예를 들어, 응답 대상 큐 알리어스 이름이 'C1_alias' 및 'C2_alias'인 애플리케이션을 사용하기 위해 두 가지 서비스 클래스가 제공되는 경우 애플리케이션은 해당 이름을 메시지 넣기 호출에서 응답 대상 큐 이름으로 사용합니다. 그러나 실제로는 애플리케이션이 큐 'C1'('C1_alias'의 경우) 및 'C2'('C2_alias'의 경우)에 메시지가 나타나는 것으로 예상합니다.

그러나 애플리케이션은 응답 대상 알리어스 큐에서 조회 호출을 하여 응답 메시지를 가져오는 데 사용해야 하는 실제 큐의 이름을 직접 검사할 수 있습니다.

관련 개념

130 페이지의 『큐 관리자 및 응답 대상 알리어스 작성 방법』
이 주제는 리모트 큐 정의를 작성할 수 있는 세 가지 방법을 설명합니다.

140 페이지의 『응답 대상 큐 알리어스 예』
이 예는 응답 대상 큐 알리어스를 사용하여 리턴된 메시지에 서로 다른 라우트(전송 큐)를 선택하는 경우를 보여 줍니다. 이 기능을 사용하려면 애플리케이션과 함께 응답 대상 큐 이름이 변경되어야 합니다.

142 페이지의 『이 예의 작동 방법』
큐 관리자가 응답 대상 큐 알리어스를 사용하는 방법과 그 예에 대한 설명입니다.

143 페이지의 『응답 대상 큐 알리어스 개요』
리모트 큐에 메시지를 넣는 애플리케이션에서 알리어스 응답 대상 큐에서 응답 메시지를 제거하는 동일한 애플리케이션까지의 프로세스 개요

응답 대상 큐 알리어스 예

이 예는 응답 대상 큐 알리어스를 사용하여 리턴된 메시지에 서로 다른 라우트(전송 큐)를 선택하는 경우를 보여 줍니다. 이 기능을 사용하려면 애플리케이션과 함께 응답 대상 큐 이름이 변경되어야 합니다.

141 페이지의 그림 15와 같이, 전송 큐, 채널, 큐 관리자 알리어스를 포함하여 응답 메시지에 리턴 라우트를 사용할 수 있어야 합니다.

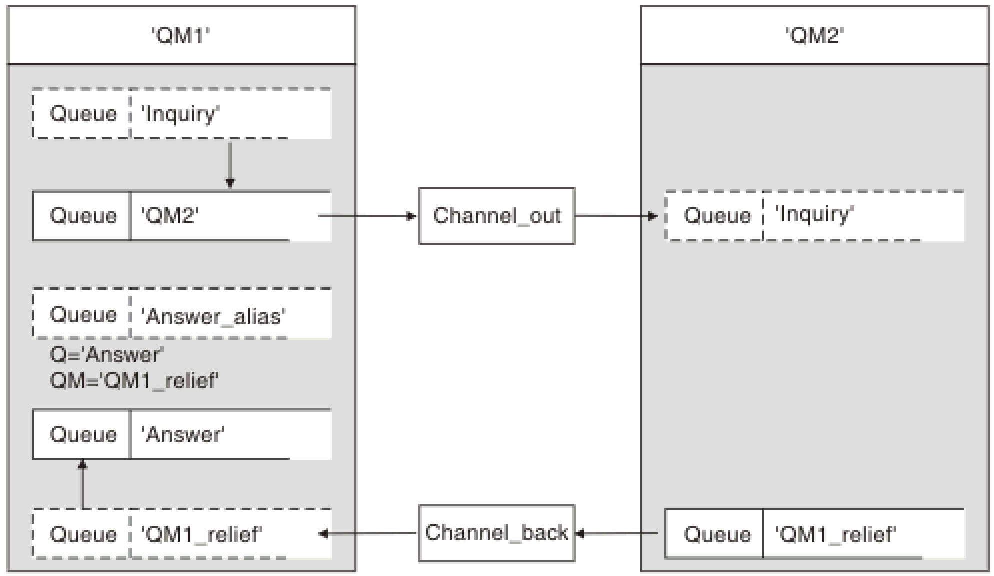


그림 *15.* 응답 대상 큐 알리어스 예

이 예는 'QM2'의 서버 애플리케이션으로 메시지를 보내는 'QM1'의 요청자 애플리케이션에 대한 것입니다. 서버의 메시지는 전송 큐 'QM1_relief'를 사용하여 대체 채널을 통해 리턴됩니다. 기본 리턴 채널은 전송 큐 'QM1'과 함께 제공됩니다.

응답 대상 큐 알리어스는 리모트 큐 정의 'Answer_alias'의 특정 용도입니다. QM1의 애플리케이션은 'Inquiry' 큐에 넣는 모든 메시지의 응답 대상 필드에 이 이름, 'Answer_alias'를 포함합니다.

응답 대상 큐 정의 'Answer_alias'는 'Answer at QM1_relief'로 정의됩니다. QM1의 애플리케이션은 해당 응답이 로컬 큐 'Answer'에 나타나는 것으로 예상합니다.

QM2의 서버 애플리케이션은 수신된 메시지의 응답 대상 필드를 사용하여 QM1의 요청자의 응답 메시지에 대한 큐 및 큐 관리자 이름을 획득합니다.

이 예에서 QM1에 사용된 정의

QM1의 IBM MQ 시스템 관리자는 응답 대상 큐 'Answer'가 다른 오브젝트와 함께 작성되는지 확인해야 합니다. '*' 표시가 있는 큐 관리자 알리어스의 이름은 또한 '*' 표시가 있는 응답 대상 큐 알리어스 정의의 큐 관리자 이름과 일치해야 합니다.

오브젝트	**정의**	
로컬 전송 큐	QM2	
리모트 큐 정의	오브젝트 이름	조회
	리모트 큐 관리자 이름	QM2
	리모트 큐 이름	조회
	전송 큐 이름	QM2 (DEFAULT)
큐 관리자 알리어스	오브젝트 이름	QM1_relief *
	큐 관리자 이름	QM1
	큐 이름	(공백)
응답 대상 큐 알리어스	오브젝트 이름	Answer_alias
	리모트 큐 관리자 이름	QM1_relief *
	리모트 큐 이름	응답

QM1에 정의 지정

애플리케이션은 응답 대상 필드를 응답 대상 큐 알리어스 이름으로 채우고 큐 관리자 이름 필드는 공백으로 둡니다.

필드	컨텐츠
큐 이름	조회
큐 관리자 이름	(공백)
응답 대상 큐 이름	Answer_alias
응답 대상 큐 관리자	(공백)

이 예에서 QM2에 사용된 정의

QM2의 IBM MQ 시스템 관리자는 수신 메시지에 대한 로컬 큐가 존재하고 올바른 이름이 지정된 전송 큐를 응답 메시지에 사용할 수 있는지 확인해야 합니다.

오브젝트	정의
로컬 큐	조회
전송 큐	QM1_relief

QM2에 정의 지정

QM2의 애플리케이션은 원래 메시지에서 응답 대상 큐 이름과 큐 관리자 이름을 검색하고 응답 대상 큐에 응답 메시지를 둘 때 해당 이름을 사용합니다.

필드	컨텐츠
큐 이름	응답
큐 관리자 이름	QM1_relief

이 예의 작동 방법

큐 관리자가 응답 대상 큐 알리어스를 사용하는 방법과 그 예에 대한 설명입니다.

이 예에서는 QM1의 요청자 애플리케이션이 항상 'Answer_alias'를 넣기 호출의 관련 필드에서 응답 대상 큐로 사용합니다. 애플리케이션은 항상 'Answer' 큐에서 메시지를 검색합니다.

응답 대상 큐 알리어스 정의는 QM1 시스템 관리자가 응답 대상 큐 'Answer'와 응답 라우트 'QM1_relief'의 이름을 변경하는 데 사용할 수 있습니다.

QM1 애플리케이션은 이 큐에서 응답을 예상하므로 큐 이름 'Answer'를 변경하는 것은 일반적으로 유용하지 않습니다. 그러나 QM1 시스템 관리자는 필요에 따라 리턴 라우트(서비스 클래스)를 변경할 수 있습니다.

큐 관리자가 응답 대상 큐 알리어스를 사용하는 방식

애플리케이션의 넣기 호출에 포함된 응답 대상 큐 이름이 응답 대상 큐 알리어스와 동일하고 큐 관리자 파트가 공백인 경우 큐 관리자 QM1이 응답 대상 큐 알리어스에서 정의를 검색합니다.

큐 관리자는 넣기 호출의 응답 대상 큐 이름을 정의의 큐 이름으로 바꿉니다. 또한 넣기 호출의 공백 큐 관리자 이름을 정의의 큐 관리자 이름으로 바꿉니다.

이러한 이름은 메시지 디스크립터에서 메시지와 함께 이동합니다.

표 *15.* 응답 대상 큐 알리어스

필드 이름	넣기(put) 호출	전송 헤더
응답 대상 큐 이름	Answer_alias	응답
응답 대상 큐 관리자 이름	(공백)	QM1_relief

응답 대상 큐 알리어스 개요

리모트 큐에 메시지를 넣는 애플리케이션에서 알리어스 응답 대상 큐에서 응답 메시지를 제거하는 동일한 애플리케이션까지의 프로세스 개요

이 예를 완료하려면 프로세스를 확인하십시오.

1. 애플리케이션은 'Inquiry' 큐를 열고 메시지를 큐에 넣습니다. 애플리케이션은 메시지 디스크립터의 응답 대상 큐를 다음으로 설정합니다.

응답 대상 큐 이름	**Answer_alias**
응답 대상 큐 관리자 이름	(공백)

2. 이름 'Answer_alias'로 리모트 큐 정의를 확인하여 큐 관리자 'QM1'이 공백 큐 관리자 이름에 응답합니다. 해당 항목이 없으면 큐 관리자가 자체 이름 'QM1'을 메시지 디스크립터의 응답 대상 큐 관리자 필드에 넣습니다.
3. 큐 관리자가 이름 'Answer_alias'로 리모트 큐 정의를 찾으면 정의에서 큐 이름과 큐 관리자 이름을 추출합니다(큐 이름 = 'Answer', 큐 관리자 이름 = 'QM1_relief'). 그런 다음 해당 이름을 메시지 디스크립터의 응답 대상 필드에 넣습니다.
4. 큐 관리자 'QM1'은 리모트 큐 정의 '조회'를 사용하여 의도한 목적지 큐가 큐 관리자 'QM2'에 있고 메시지가 전송 큐 'QM2'에 배치되었는지 판별합니다. 'QM2'는 큐 관리자 'QM2'에서 큐로 예정된 메시지의 기본 전송 큐 이름입니다.
5. 큐 관리자 'QM1'이 전송 큐에 메시지를 넣으면 메시지에 전송 헤더를 추가합니다. 이 헤더에는 목적지 큐, 'Inquiry'의 이름과 목적지 큐 관리자, 'QM2'가 포함됩니다.
6. 메시지가 큐 관리자 'QM2'에 도착하고 'Inquiry' 로컬 큐에 배치됩니다.
7. 애플리케이션이 이 큐에서 메시지를 가져와 처리합니다. 애플리케이션은 응답 메시지를 준비하고 해당 응답 메시지를 원래 메시지의 메시지 디스크립터에서 응답 대상 큐에 넣습니다.

응답 대상 큐 이름	**응답**
응답 대상 큐 관리자 이름	QM1_relief

8. 큐 관리자 'QM2'가 넣기 명령을 수행합니다. 큐 관리자 이름, 'QM1_relief'가 리모트 큐 관리자임을 발견한 후 동일한 이름 'QM1_relief'로 전송 큐에 메시지를 배치합니다. 목적지 큐 이름, 'Answer'와 목적지 큐 관리자, 'QM1_relief'를 포함하는 전송 헤더가 메시지에 제공됩니다.
9. 메시지는 큐 관리자 'QM1'로 전송됩니다. 큐 관리자가 큐 관리자 이름 'QM1_relief'가 알리어스임을 인식하고 알리어스 정의 'QM1_relief'에서 물리적 큐 관리자 이름 'QM1'을 추출합니다.
10. 그런 다음 큐 관리자 'QM1'이 전송 헤더에 포함된 큐 이름, 'Answer'에 메시지를 넣습니다.
11. 애플리케이션은 큐 'Answer'에서 응답 메시지를 추출합니다.

네트워킹 고려사항

분산 큐잉 환경에서는 큐 이름과 큐 관리자 이름만으로 메시지 목적지의 주소를 지정하므로 특정 규칙이 적용됩니다.

1. 큐 관리자 이름이 지정되는 경우 해당 이름은 로컬 큐 관리자 이름과 다릅니다.
   - 이름이 같은 전송 큐를 사용할 수 있어야 합니다. 이 전송 큐는 다른 큐 관리자로 메시지를 이동하는 메시지 채널의 일부여야 합니다. 또는
   - 큐 관리자 이름을 동일하거나 다른 큐 관리자 이름과 선택적 전송 큐로 해석하기 위해서는 큐 관리자 알리어스 정의가 존재해야 합니다. 또는
   - 전송 큐 이름을 해석할 수 없고 기본 전송 큐가 정의된 경우에는 기본 전송 큐를 사용합니다.
2. 큐 이름만 제공된 경우 동일한 이름을 사용하는 임의 유형의 큐를 로컬 큐 관리자에서 사용할 수 있어야 합니다. 이 큐는 인접 큐 관리자로의 전송 큐, 큐 관리자 이름, 선택적 전송 큐로 해석되는 리모트 큐 정의일 수 있습니다.

클러스터링 환경에서 이 작업이 작동하는 방법을 보려면 클러스터를 참조하십시오.

z/OS 큐 관리자가 큐 공유 그룹(QSG)에서 실행되고 그룹 내 큐잉(IGQ)을 사용하는 경우에는 SYSTEM.QSG.TRANSMIT.QUEUE를 사용할 수 있습니다. 자세한 정보는 그룹 내 큐잉의 내용을 참조하십시오.

분산 큐잉 환경 내 특정 큐 관리자에서 다른 큐 관리자로 메시지를 이동하는 메시지 채널 시나리오를 고려하십시오.

이동하는 메시지는 네트워크의 다른 큐 관리자에서 송신된 것이며 알 수 없는 큐 관리자 이름을 목적지로 갖는 일부 메시지가 도착할 수 있습니다. 이 문제는 예를 들어, 큐 관리자 이름이 변경되었거나 시스템에서 제거된 경우 발생할 수 있습니다.

채널 프로그램은 해당 메시지의 전송 큐를 찾을 수 없을 때 이 상황을 인식하고 미전달 메시지(데드-레터) 큐에 메시지를 놓습니다. 해당 메시지를 찾아 올바른 목적지로 전달되도록 조정해야 합니다. 또는 진원지를 알 수 있는 경우 진원지로 돌려보냅니다.

원래 메시지에서 보고 메시지가 요청된 경우에는 이러한 상황에서 예외 보고서가 생성됩니다.

이름 해석 규칙

목적지 큐의 ID를 변경(즉, 논리적 이름을 물리적 이름으로 변경)하는 이름 해석은 원래 큐 관리자에서 한 번만 발생합니다.

다양한 알리어스 가능성의 후속 사용은 메시지 플로우를 분리하고 결합할 때만 수행됩니다.

리턴 라우팅

메시지에는 큐 및 큐 관리자 이름 양식의 리턴 주소가 포함될 수 있습니다. 이 리턴 주소 양식은 분산 큐잉 환경과 클러스터링 환경에서 모두 사용할 수 있습니다.

이 주소는 일반적으로 메시지를 작성하는 애플리케이션으로 지정됩니다. 이 주소는 사용자 엑시트 애플리케이션을 포함하여 메시지를 처리하는 애플리케이션으로 수정될 수 있습니다.

이 주소의 소스에 관계 없이, 메시지를 처리하는 애플리케이션이 원래 애플리케이션으로 응답, 상태 또는 보고서 메시지를 리턴하기 위해 이 주소를 사용하도록 선택할 수 있습니다.

이러한 응답 메시지의 경로 방식은 원래의 메시지 경로 방식과 동일합니다. 다른 큐 관리자에 작성하는 메시지 플로우에 해당 리턴 플로우가 필요함에 유의하십시오.

물리적 이름 충돌

이 목적지 응답 대상 큐 이름은 원래 큐 관리자에서 물리적 큐 이름으로 해석되었습니다. 이 이름은 응답 큐 관리자에서 다시 해석되어서는 안 됩니다.

물리적 큐 이름과 논리적 큐 이름에 대한 네트워크 범위 동의로만 방지될 수 있는 이름 충돌 문제점이 존재할 수 있습니다.

큐 이름 변환 관리

큐 관리자 알리어스 정의 또는 리모트 큐 정의를 작성하는 경우, 해당 이름을 포함하는 모든 메시지에 대해 이름 분석이 수행됩니다. 이 상황은 관리가 필요합니다.

이 설명은 인접 시스템에 대한 메시지 채널이 있는 개별 시스템에 관심이 있는 채널 기획자와 애플리케이션 설계자에게 제공됩니다. 채널 기획과 제어에 대한 로컬 관점을 갖습니다.

큐 관리자 알리어스 정의 또는 리모트 큐 정의를 작성하면, 메시지의 소스에 관계 없이 해당 이름을 사용하는 모든 메시지에 대해 이름 해석이 수행됩니다. 큐 관리자 네트워크에 여러 개의 큐가 있는 이러한 상황을 감독하기 위해 다음 내용의 테이블을 유지해야 합니다.

- 해석된 큐 이름, 해석된 큐 관리자 이름, 그리고 해석된 전송 큐 이름과 관련하여 소스 큐 및 소스 큐 관리자의 이름과 그 해석 방법
- 다음과 관련된 소스 큐의 이름
  - 해석된 목적지 큐 이름
  - 해석된 목적지 큐 관리자 이름

- 전송 큐
- 메시지 채널 이름
- 인접 시스템 이름
- 응답 대상 큐 이름

참고: 이 컨텍스트에서 소스라는 용어의 사용은 애플리케이션이 제공하는 큐 이름이나 큐 관리자 이름 또는 메시지를 넣기 위해 큐를 열 때의 채널 프로그램을 의미합니다.

이들 표에 대한 각각의 예가 145 페이지의 표 16, 145 페이지의 표 17, 145 페이지의 표 18에 나와 있습니다.

이들 표의 이름은 이 절의 예에서 도출된 것이며 이 표는 단일 노드의 실제 큐 이름 분석 예로 작성되지 않았습니다.

표 *16.* 큐 관리자 *QMA*에서의 큐 이름 해석

큐를 열 때 지정된 소스 큐	**큐를 열 때 지정된 소스 큐 관리자**	**해석된 큐 이름**	**해석된 큐 관리자 이름**	**해석된 전송 큐 이름**	**해석 유형**
QA_norm	-	QA_norm	QMB	QMB	리모트 큐
(임의)	QMB	-	-	QMB	(없음)
QA_norm	-	QA_norm	QMB	TX1	리모트 큐
QB	QMC	QB	QMD	QMB	큐 관리자 알리어스

표 *17.* 큐 관리자 *QMB*에서의 큐 이름 해석

큐를 열 때 지정된 소스 큐	**큐를 열 때 지정된 소스 큐 관리자**	**해석된 큐 이름**	**해석된 큐 관리자 이름**	**해석된 전송 큐 이름**	**해석 유형**
QA_norm	-	QA_norm	QMB	-	(없음)
QA_norm	QMB	QA_norm	QMB	-	(없음)
QA_norm	QMB_PRIORITY	QA_norm	QMB	-	큐 관리자 알리어스
(임의)	QMC	(임의)	QMC	QMC	(없음)
(임의)	QMD_norm	(임의)	QMD_norm	TX1	큐 관리자 알리어스
(임의)	QMD_PRIORITY	(임의)	QMD_PRIORITY	QMD_fast	큐 관리자 알리어스
(임의)	QMC_small	(임의)	QMC_small	TX_small	큐 관리자 알리어스
(임의)	QMC_large	(임의)	QMC_large	TX_external	큐 관리자 알리어스
QB_small	QMC	QB_small	QMC	TX_small	리모트 큐
QB_large	QMC	QB_large	QMC	TX_large	리모트 큐
(임의)	QME	(임의)	QME	TX1	큐 관리자 알리어스
QA	QMC	QA	QMC	TX1	리모트 큐
QB	QMD	QB	QMD	TX1	리모트 큐

표 *18.* 큐 관리자 *QMA*에서의 응답 대상 큐 이름 해석

애플리케이션 설계		**응답 대상 알리어스 정의**	
로컬 QMGR	**메시지의 큐 이름**	**응답 대상 큐 알리어스 이름**	**다음으로 재정의**
QMA	QRR	QR	QMA_class1의 QRR

채널 메시지 순서 매기기

채널은 순서 번호를 사용하여 메시지가 전송 큐에서 가져온 것과 동일한 순서로 전달되는지 확인합니다.

채널이 시작될 때 채널 순서 번호가 검사되고 불일치가 발생해야 하는 경우 채널의 양쪽에서 지속적 동기화 데이터가 유실되었음을 의미합니다. 예를 들어, 재해 복구 (DR) 구성 또는 채널이 인다우트 상태인 경우 배치 처리 종료가 인터럽트되었음을 의미합니다.

RESET CHANNEL 명령을 발행하는 것은 메시지들의 손실 또는 중복을 야기하지 않는다. RESET은 어떤 것이 올바른 것으로 보이지 않는다는 경고를 IBM MQ 에서 확인합니다. 지속적 상태가 손실된 인다우트 채널은 RESOLVE CHANNEL 명령을 실행할 때까지 RESET후 시작하는 데 계속 실패하며, 이는 배치를 유실하거나 복제할 가능성이 있는 조치이다.

이 정보는 DISPLAY CHSTATUS를 사용하여 표시될 수 있습니다. 순서 번호와 ID, LUWID는 배치에서 전송된 마지막 메시지에 대한 영구 스토리지에 저장됩니다. 채널 시작 시 이 값을 사용함으로써 링크 양측이 메시지가 성공적으로 전송되었음을 확인할 수 있습니다.

순차적 메시지 검색

다음 조건이 충족되는 경우, 애플리케이션이 동일한 목적지 큐에 일련의 메시지를 넣으면 **단일** 애플리케이션의 일련의 MQGET 조작으로 해당 메시지를 순차적으로 검색할 수 있습니다.

- 모든 넣기 요청이 동일한 애플리케이션에서 수행되었습니다.
- 모든 넣기 요청이 동일한 작업 단위에서 시작되었거나 모든 넣기 요청이 작업 단위 밖에서 작성되었습니다.
- 모든 메시지는 우선순위가 같습니다.
- 메시지의 지속성이 모두 같습니다.
- 리모트 큐잉의 경우, 넣기 요청을 작성하는 애플리케이션부터 해당 큐 관리자와 상호통신을 통해 목적지 큐 관리자 및 대상 큐까지 한 가지 경로만 존재하도록 구성됩니다.
- 데드-레터 큐에는 메시지를 넣지 않습니다(예를 들어, 큐가 일시적으로 가득 찬 경우).
- 메시지를 가져오는 애플리케이션은 특정 *MsgId* 또는 *CorrelId*를 지정하거나, 메시지 우선순위를 사용하는 등의 방법을 사용하여 고의로 검색의 순서를 변경하지 않습니다.
- 목적지 큐에서 메시지를 검색하는 가져오기 조작은 하나의 애플리케이션만 수행합니다. 애플리케이션이 여러 개인 경우, 해당 애플리케이션은 송신 애플리케이션이 각 순서의 모든 메시지를 가져오도록 설계되어야 합니다.

참고: 다른 태스크와 작업 단위로부터의 메시지는 단일 작업 단위 내에서 시퀀스를 넣은 경우라도 시퀀스와 함께 배치될 수 있습니다.

이러한 조건이 충족되지 않고 대상 큐의 메시지 순서가 중요한 경우에는 자체 메시지 순서 번호를 메시지의 일부로 사용하도록 애플리케이션을 코드화하여 메시지 순서를 유지할 수 있습니다.

빠른, 비지속 메시지의 검색 순서

빠른 채널의 비지속 메시지는 동일한 채널의 지속 메시지를 추월하지 못해 순서가 유지되지 않을 수 있습니다. 수신 MCA는 목적지 큐에 즉시 비지속 메시지를 넣어 표시 가능하게 만듭니다. 지속 메시지는 다음 동기점까지 표시되지 않습니다.

루프백 테스트

루프백 테스트는 다른 시스템에 실제로 링크하지 않고 통신 링크를 테스트할 수 있는 비z/OS 플랫폼에 대한 기술입니다.

두 큐 관리자가 개별 시스템에 있는 것처럼 연결을 설정하지만 동일한 시스템의 다른 프로세스로 루프백하여 연결을 테스트합니다. 이 기술은 활성 네트워크 없이 통신 코드를 테스트할 수 있음을 의미합니다.

이 방법은 사용하는 제품과 프로토콜에 따라 다릅니다.

Windows 시스템에서 "루프백" 어댑터를 사용할 수 있습니다.

자세한 정보는 사용하고 있는 제품 설명서를 참조하십시오.

라우트 추적 및 활동 레코드

메시지 라우트가 두 가지 방식으로 일련의 큐 관리자를 통해 수행되는지 확인할 수 있습니다.

제어 명령 **dspmqrte**를 통해 사용 가능한 IBM MQ 표시 라우트 애플리케이션을 사용하거나, 활동 기록을 사용할 수 있습니다. 이 두 주제는 모두 모니터링 참조에서 설명됩니다.

분산 큐 관리 소개

분산 큐 관리(DQM)는 큐 관리자 간 통신을 정의하고 제어하는 데 사용됩니다.

분산 큐 관리:

- 큐 관리자 간 통신 채널을 정의하고 제어할 수 있습니다.
- 메시지 채널 서비스를 제공하여 전송 큐로 알려진 로컬 큐의 유형에서 로컬 시스템의 통신 링크로, 통신 링크에서 목적지 큐 관리자의 로컬 큐로 메시지를 이동합니다.
- 패널, 명령, 프로그램을 사용하여 채널 작업을 모니터링하고 문제점을 진단하는 기능을 제공합니다.

채널 정의는 채널 이름을 전송 큐, 통신 링크 ID, 채널 속성과 연관시킵니다. 다른 플랫폼에서는 채널 정의가 다른 방식으로 구현됩니다. 메시지 송신 및 수신은 메시지 채널 에이전트(MCA)라는 프로그램에서 제어하는데, 이 프로그램은 채널 정의를 사용하여 통신을 시작하고 제어합니다.

MCA는 차례로 DQM이 직접 제어합니다. 구조는 플랫폼 의존적이지만 일반적으로 운영자 명령 및 패널과 함께 리스너와 트리거 모니터를 포함합니다.

메시지 채널은 특정 큐 관리자에서 다른 큐 관리자로 메시지를 이동시키는 단방향 파이프입니다. 따라서 메시지 채널은 MCA 쌍으로 표시되는 두 개의 엔드 포인트를 갖습니다. 각 엔드 포인트에는 메시지 채널 끝에 대한 정의가 있습니다. 예를 들어, 한쪽 끝은 송신자를 정의하고 다른 측은 수신자를 정의합니다.

채널을 정의하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

- ULW 175 페이지의 『UNIX, Linux, and Windows에서 채널 모니터링 및 제어』
- z/OS 660 페이지의 『z/OS에서 채널 모니터링 및 제어』
- IBM i 196 페이지의 『IBM i에서 채널 모니터링 및 제어』

메시지 채널 계획 예는 다음을 참조하십시오.

- ULW UNIX, Linux, and Windows의 메시지 채널 계획 예
- IBM i IBM i의 메시지 채널 계획 예
- z/OS z/OS의 메시지 채널 계획 예
- z/OS 큐 공유 그룹을 사용하는 z/OS의 메시지 채널 계획 예

채널 엑시트에 대한 정보는 메시지 채널을 위한 채널 엑시트 프로그램을 참조하십시오.

관련 개념

148 페이지의 『메시지 송신 및 수신』
다음 그림은 메시지가 전송될 때 엔티티 간의 관계를 나타내는 분산 큐 관리 모델을 보여줍니다. 또한 제어를 위한 플로우를 보여줍니다.

154 페이지의 『채널 제어 기능』
채널 제어 기능은 채널을 정의, 모니터, 제어할 수 있는 기능을 제공합니다.

167 페이지의 『메시지를 전달할 수 없을 경우의 메시지 처리』
메시지를 전달할 수 없는 경우 MCA가 여러 가지 방법으로 메시지를 처리할 수 있습니다. 다시 시도하거나 송신자로 돌아가거나 데드-레터 큐에 넣을 수 있습니다.

172 페이지의 『초기화 및 구성 파일』
채널 초기화 데이터 처리는 IBM MQ 플랫폼에 따라 다릅니다.

173 페이지의 『데이터 변환』
IBM MQ 메시지를 다른 큐 관리자의 큐 사이에 보내려면 데이터 변환이 필요할 수 있습니다.

173 페이지의 『사용자 고유의 메시지 채널 에이전트 작성』
IBM MQ를 사용하여 사용자 고유의 메시지 채널 에이전트(MCA) 프로그램을 작성하거나 독립 소프트웨어 벤더의 프로그램을 설치할 수 있습니다.

174 페이지의 『분산 큐 관리를 위해 고려해야 할 기타 사항』
분산 큐 관리를 위한 IBM MQ 준비 시 고려해야 할 기타 주제. 이 주제에서는 미전달 메시지 큐, 사용 중인 큐, 시스템 확장 기능, 사용자 엑시트 프로그램과 실행 중인 채널 및 리스너를 신뢰성 있는 애플리케이션으로 다룹니다.

관련 정보

구성 정보 예

메시지 송신 및 수신

다음 그림은 메시지가 전송될 때 엔티티 간의 관계를 나타내는 분산 큐 관리 모델을 보여줍니다. 또한 제어를 위한 플로우를 보여줍니다.

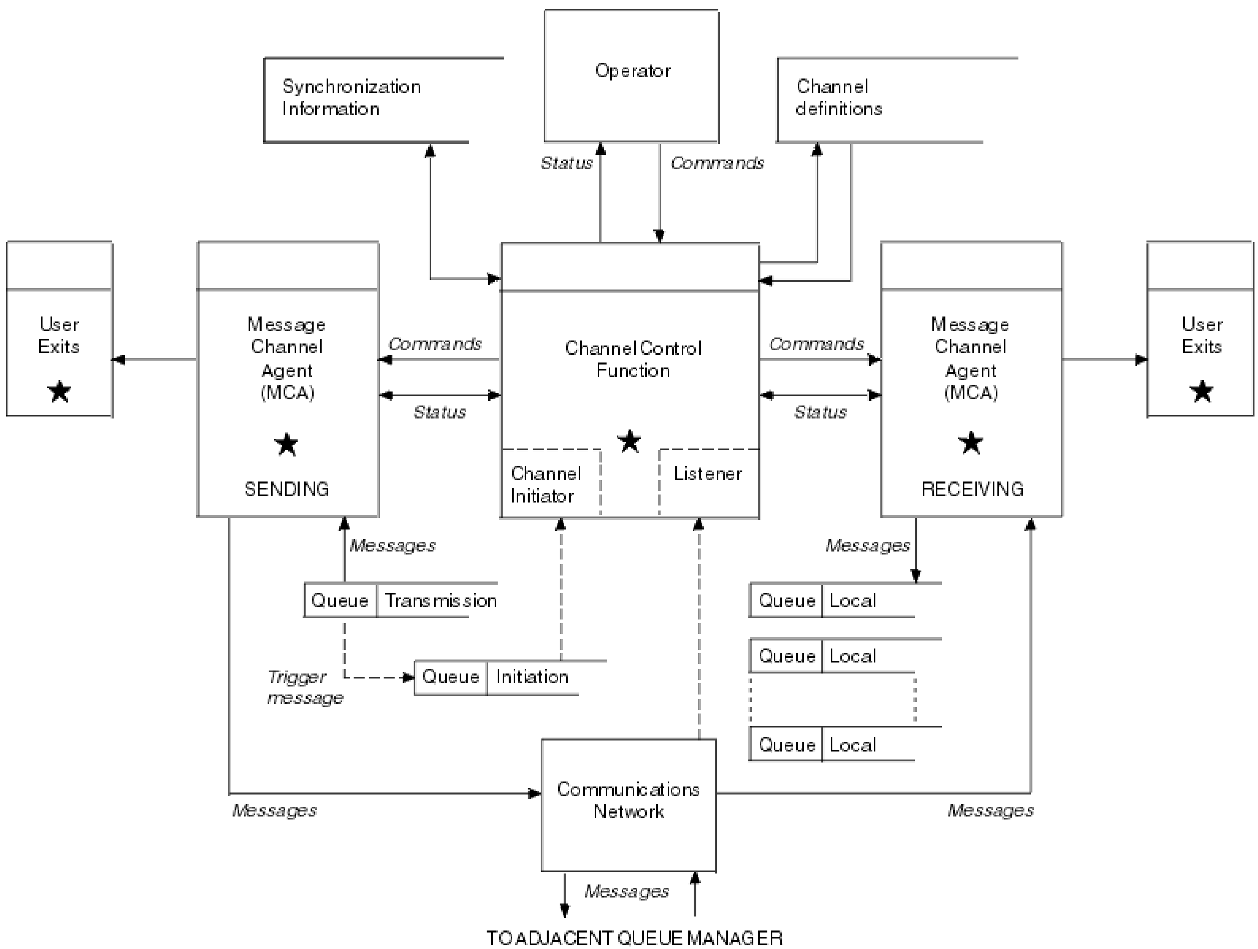


그림 *16.* 분산 큐 관리 모델

참고:

1. 플랫폼에 따라 채널당 MCA가 하나 있습니다. 특정 큐 관리자의 경우 하나 이상의 채널 제어 기능이 존재할 수 있습니다.
2. MCA 및 채널 제어 기능 구현은 플랫폼에 따라 크게 다릅니다. 이는 프로그램이나 프로세스 또는 스레드일 수 있으므로 여러 독립 또는 링크된 파트로 구성되는 단일 또는 다수의 엔티티가 될 수 있습니다.
3. 별표로 표시된 모든 구성요소는 MQI를 사용할 수 있습니다.

채널 매개변수

MCA는 다음 몇 가지 방법 중 하나로 매개변수를 수신합니다.

- 명령으로 시작되면, 채널 이름이 데이터 영역으로 전달됩니다. 그런 다음 MCA가 직접 채널 정의를 읽어 속성을 확보합니다.
- 송신자의 경우, 또한 몇몇 상황에서는 서버 채널의 경우, 큐 관리자 트리거에 의해 MCA가 자동으로 시작될 수 있습니다. 채널 이름은 사용 가능한 경우 트리거 프로세스 정의로부터 검색되며, MCA로 전달됩니다. 나머지 처리는 이전에 설명한 것과 같습니다. 서버 채널은 완전히 규정된 경우, 즉 연결될 CONNAME을 지정하는 경우에만 트리거되도록 설정되어야 합니다.
- 송신자, 서버, 요청자 또는 클라이언트 연결이 원격으로 시작된 경우, 채널 이름은 파트너 메시지 채널 에이전트로부터 초기 데이터로 전달됩니다. MCA는 그 속성을 확보하기 위해 채널 정의를 직접 읽습니다.

또한 채널 정의에 정의되어 있지 않은 특정 속성도 조정 가능합니다.

메시지 분할
: 한쪽이 분할 메시지를 지원하지 않으면 분할 메시지를 보내지 않습니다.

변환 기능
: 한쪽에서 필요한 코드 페이지 변환이나 숫자 인코딩 변환을 필요할 때 수행하지 못하면 다른 쪽이 이를 처리해야 합니다. 필요할 때 어느 한쪽에서도 지원하지 않으면 채널을 시작할 수 없습니다.

분배 목록 지원
: 한쪽에서 분배 목록을 지원하지 않으면 파트너 MCA가 알 수 있도록 전송 큐에서 플래그를 설정하여 여러 목적지를 대상으로 하는 메시지를 가로챕니다.

채널 상태 및 순서 번호

메시지 채널 에이전트 프로그램은 현재 순서 번호와 각 채널의 논리 작업 단위 그리고 채널의 일반 상태 레코드를 보관합니다. 일부 플랫폼은 이 상태 정보를 표시하여 패널 제어에 도움을 줍니다.

다른 큐 관리자로 메시지를 보내는 방법

이 절에서는 전제조건, 필요한 권한 등 큐 관리자 간에 메시지를 보내는 가장 간단한 방법을 설명합니다. 리모트 큐 관리자로 메시지를 보내기 위해 다른 방법을 사용할 수도 있습니다.

특정 큐 관리자에서 다른 큐 관리자로 메시지를 보내려면 다음 단계를 수행해야 합니다.

1. 선택한 통신 프로토콜을 사용할 수 있는지 확인하십시오.
2. 큐 관리자를 시작하십시오.
3. 채널 시작기를 시작하십시오.
4. 리스너를 시작하십시오.

또한 필요한 오브젝트를 작성하려면 올바른 IBM MQ 보안 권한이 필요합니다.

특정 큐 관리자에서 다른 관리자로 메시지를 보내려면 다음을 수행하십시오.

- 소스 큐 관리자에서 다음 오브젝트를 정의하십시오.
  - 송신자 채널
  - 리모트 큐 정의
  - 이니시에이션 큐(z/OS required on z/OS, 다른 경우는 선택사항)
  - 전송 큐
  - 데드-레터 큐
- 대상 큐 관리자에서 다음 오브젝트를 정의하십시오.
  - 수신자 채널
  - 대상 큐
  - 데드-레터 큐

다음과 같이 IBM MQ 플랫폼에 따라 여러 가지 다른 방법을 사용하여 이러한 오브젝트를 정의할 수 있습니다.

- 모든 플랫폼에서, IBM MQ 스크립트 명령(MQSC)(MQSC 명령의 설명 참조), 프로그래밍 가능 명령 형식(PCF) 명령(관리 태스크 자동화의 설명 참조) 또는 IBM MQ 탐색기를 사용할 수 있습니다.

- z/OS z/OS의 경우 조작 및 제어판도 사용할 수 있습니다(IBM MQ for z/OS 관리의 설명 참조).
- IBM i IBM i의 경우 패널 인터페이스도 사용할 수 있습니다.

다른 큐 관리자로 메시지를 보내기 위한 컴포넌트 작성에 대한 자세한 정보는 다음 하위 주제를 참조하십시오.

관련 개념

129 페이지의 『IBM MQ 분산 큐잉 기술』
이 절의 하위 주제에서는 채널 계획 시 사용되는 기술을 설명합니다. 이 하위 주제에서는 큐 관리자와 함께 연결하는 방법을 계획하고 애플리케이션 간 메시지 플로우를 관리하는 데 도움이 되는 기술을 설명합니다.

147 페이지의 『분산 큐 관리 소개』
분산 큐 관리(DQM)는 큐 관리자 간 통신을 정의하고 제어하는 데 사용됩니다.

169 페이지의 『채널 트리거』
IBM MQ에서는 큐에 대한 특정 조건이 충족될 때 자동으로 애플리케이션을 시작하기 위한 기능을 제공합니다. 이 기능을 트리거라고 합니다.

167 페이지의 『메시지의 안전성』
IBM MQ의 일반 복구 기능 이외에 분산 큐잉 관리를 통해 메시지 채널의 양측 간에 조정된 동기점 프로시저를 사용함으로써 메시지를 올바르게 전달할 수 있습니다. 이 프로시저가 오류를 감지하면, 채널이 닫혀 문제를 조사하고 채널이 다시 시작될 때까지 메시지를 전송 큐에 안전하게 유지할 수 있습니다.

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

관련 태스크

5 페이지의 『Multiplatforms에서 큐 관리자 작성 및 관리』
메시지 및 큐를 사용하려면 먼저 최소한 하나의 큐 관리자 및 연관된 오브젝트를 작성하고 시작해야 합니다. 큐 관리자는 연관된 자원, 특히 큐에서 소유하는 자원을 관리합니다. 큐 관리자는 IBM MQ 오브젝트 작성, 수정, 표시, 삭제를 위한 메시지 큐잉 인터페이스(MQI) 호출 및 명령에 대한 큐잉 서비스를 애플리케이션에 제공합니다.

175 페이지의 『UNIX, Linux, and Windows에서 채널 모니터링 및 제어』
DQM의 경우 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어해야 합니다. 명령, 프로그램, IBM MQ Explorer, 채널 정의 파일, 동기화 정보 스토리지 영역을 사용하여 채널을 제어할 수 있습니다.

196 페이지의 『IBM i에서 채널 모니터링 및 제어』
DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다. 각 큐 관리자에는 호환 가능한 리모트 큐 관리자에 대한 상호 연결을 제어하기 위한 DQM 프로그램이 있습니다.

14 페이지의 『서버와 클라이언트 간의 연결 구성』
IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

216 페이지의 『큐 관리자 클러스터 구성』
클러스터는 초기 구성 및 진행 중인 관리를 모두 단순화하는 방식으로 큐 관리자를 상호 연결하기 위한 메커니즘을 제공합니다. 클러스터 컴포넌트를 정의하고 클러스터를 작성 및 관리할 수 있습니다.

채널 정의

특정 큐 관리자에서 다른 큐 관리자로 메시지를 보내려면 두 개 채널을 정의해야 합니다. 소스 큐 관리자에서 한 채널을 정의하고 대상 큐 관리자에 한 채널을 정의해야 합니다.

소스 큐 관리자에서

SENDER 유형의 채널을 정의하십시오. 다음을 지정해야 합니다.

- 사용될 전송 큐의 이름(XMITQ 속성)
- 파트너 시스템의 연결 이름(CONNAME 속성)
- 사용 중인 통신 프로토콜의 이름(TRPTYPE 속성). IBM MQ for z/OS의 경우 프로토콜이 TCP 또는 LU6.2여야 합니다. 다른 플랫폼에서는 이를 지정하지 않아도 됩니다. 그대로 두어 기본 채널 정의에서 값을 선택할 수 있습니다.

모든 채널 속성에 대한 자세한 내용은 채널 속성에 제공됩니다.

대상 큐 관리자에서

채널 유형이 RECEIVER이고 송신자 채널과 이름이 같은 채널을 정의하십시오.

사용 중인 통신 프로토콜의 이름을 지정하십시오(TRPTYPE 속성). IBM MQ for z/OS의 경우 프로토콜이 TCP 또는 LU6.2여야 합니다. 다른 플랫폼에서는 이를 지정하지 않아도 됩니다. 그대로 두어 기본 채널 정의에서 값을 선택할 수 있습니다.

수신자 채널 정의는 일반일 수 있습니다. 이는 동일 수신자와 통신하는 여러 큐 관리자가 있는 경우, 송신 채널이 수신자에 동일한 이름을 모두 지정할 수 있으며 하나의 수신자 정의가 모두에 적용된다는 것을 의미합니다.

채널을 정의하면 PING CHANNEL 명령을 사용하여 테스트할 수 있습니다. 이 명령은 송신자 채널에서 수신자 채널로 특별한 메시지를 보내고 리턴되는지 확인합니다.

참고: TRPTYPE 매개변수의 값은 응답 메시지 채널 에이전트에서 무시됩니다. 예를 들어, 송신자 채널 정의에 있는 TCP의 TRPTYPE은 수신자 채널 정의에 있는 LU62의 TRPTYPE을 파트너로 사용하여 시작됩니다.

큐 정의

특정 큐 관리자에서 다른 큐 관리자로 메시지를 송신하려면 최대 여섯 개의 큐를 정의해야 합니다. 소스 큐 관리자에서 최대 네 개의 큐, 대상 큐 관리자에서 최대 두 개의 큐를 정의해야 합니다.

소스 큐 관리자에서

- 리모트 큐 정의

  이 정의에서는 다음을 지정하십시오.

  리모트 큐 관리자 이름
  대상 큐 관리자의 이름

  리모트 큐 이름
  대상 큐 관리자의 대상 큐 이름

  전송 큐 이름
  전송 큐의 이름. 이 전송 큐 이름은 지정하지 않아도 됩니다. 이 이름을 지정하지 않으면 대상 큐 관리자와 이름이 같은 전송 큐를 사용합니다. 이 큐가 없으면 기본 전송 큐를 사용합니다. 기본적으로 큐가 표시되도록 전송 큐에 대상 큐 관리자와 동일한 이름을 지정하는 것이 좋습니다.

- 이니시에이션 큐 정의

  z/OS 필수입니다. 이니시에이션 큐 SYSTEM.CHANNEL.INITQ를 사용해야 합니다.

  Multi 선택적입니다. 이니시에이션 큐 이름으로 SYSTEM.CHANNEL.INITQ를 지정할 수 있습니다.

- 전송 큐 정의

  USAGE 속성이 XMITQ로 설정된 로컬 큐. IBM i IBM MQ for IBM i 기본 인터페이스를 사용하는 경우 USAGE 속성은 *TMQ입니다.

- 데드-레터 큐 정의

  미전달 메시지를 쓸 수 있는 데드-레터 큐를 정의하십시오.

대상 큐 관리자에서

- 로컬 큐 정의

  대상 큐. 이 큐의 이름은 소스 큐 관리자에 있는 리모트 큐 정의의 리모트 큐 이름 필드에 지정된 것과 같아야 합니다.

- 데드-레터 큐 정의

  미전달 메시지를 쓸 수 있는 데드-레터 큐를 정의하십시오.

관련 개념

152 페이지의 『전송 큐 작성』
채널(요청자 채널 이외)을 시작하려면 이 절의 설명대로 전송 큐가 정의되어야 합니다. 전송 큐는 채널 정의에 이름이 지정되어야 합니다.

152 페이지의 『IBM i에서 전송 큐 작성』
MQM 큐 작성 패널을 사용하여 IBM i 플랫폼에서 전송 큐를 작성할 수 있습니다.

전송 큐 작성
채널(요청자 채널 이외)을 시작하려면 이 절의 설명대로 전송 큐가 정의되어야 합니다. 전송 큐는 채널 정의에 이름이 지정되어야 합니다.

각 송신 메시지 채널에 대해 USAGE 속성을 XMITQ로 설정하여 로컬 큐를 정의합니다. 리모트 큐 정의에서 특정 전송 큐를 사용하려면 표시된 대로 리모트 큐를 작성하십시오.

전송 큐를 작성하려면 다음 예와 같이 IBM MQ 명령(MQSC)을 사용하십시오.

전송 큐 작성 예

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

리모트 큐 작성 예

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

예제에서와 같이 전송 큐를 원격 시스템의 큐 관리자 이름으로 지정해 보십시오.

*IBM i*에서 전송 큐 작성
MQM 큐 작성 패널을 사용하여 IBM i 플랫폼에서 전송 큐를 작성할 수 있습니다.

각 송신 메시지 채널마다 사용 필드 속성을 *TMQ로 설정하여 로컬 큐를 정의해야 합니다.

리모트 큐 정의를 사용하려면 동일한 명령을 사용하여 *RMT 유형과 *NORMAL 용도의 큐를 작성하십시오.

전송 큐를 작성하려면 명령행에서 CRTMQMQ 명령을 사용하여 첫 번째 큐 작성 패널을 표시하십시오(152 페이지의 그림 17 참조).

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

큐 이름...........

큐 유형..........  ____ * ALS, * LCL, * MDL, * RMT

메세지 큐 관리자 이름....  * DFT________________________________
-----









Bottom
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
+
```

그림 *17.* 큐 작성*(1)*

큐의 이름을 입력하고 작성하려는 큐의 유형(로컬, 리모트 또는 알리어스)을 지정하십시오. 전송 큐의 경우 이 패널에서 로컬(*LCL)을 지정하고 Enter를 누르십시오.

MQM 큐 작성 패널의 두 번째 페이지가 표시됩니다(153 페이지의 그림 18 참조).

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

큐 이름........... > HURS.2.HURS.PRIORIT

큐 유형.......... > * LCL * ALS, * LCL, * MDL, * RMT
메세지 큐 관리자 이름....  * DFT
대체.............  * 아니오, * 아니오, * 예
텍스트 '설명'........ ' '
넣기 사용..........  * 예 * SYSDFTQ, * NO, * 예
디폴트 메세지 우선순위......  0 0-9, * SYSDFTQ
기본 메시지 지속성...  * NO * SYSDFTQ, * NO, * YES
프로세스 이름.......... ' '
트리거 사용.......  * NO * SYSDFTQ, * NO, * YES
사용 가능...........  * 예 * SYSDFTQ, * NO, * 예
공유 사용........  * 예 * SYSDFTQ, * NO, * 예




More...
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *18.* 큐 작성*(2)*

표시된 기본값을 변경하십시오. 다음 페이지를 눌러 다음 화면으로 스크롤하십시오(153 페이지의 그림 19 참조).

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

디폴트 공유 옵션.......  * 예 * SYSDFTQ, * NO, * 예
메시지 전달 순서....  * PTY * SYSDFTQ, * PTY, * 50
하든 백아웃 계수.......  * NO * SYSDFTQ, * NO, * YES
트리거 유형.........  * FIRST * SYSDFTQ, * FIRST, * ALL ...
트리거 깊이.........  1-999999999, * SYSDFTQ
트리거 메세지 우선순위......  0 0-9, * SYSDFTQ
트리거 데이터.......... ' '
보유 간격.........  999999999 0-999999999, * SYSDFTQ
최대 큐 용량.......  5000 1-24000, * SYSDFTQ
최대 메시지 길이........  4194304 0-4194304, * SYSDFTQ
백아웃 임계값.......  0-999999999, * SYSDFTQ
백아웃 큐..... ' '.
이니시에이션 큐....... ' '.



More...
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *19.* 큐 작성*(3)*

전송 큐의 경우 이 패널의 사용 필드에 *TMQ를 입력하고 다른 필드에 표시된 기본값을 변경하십시오.

```
Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

사용법............  * TMQ * SYSDFTQ, * NORMAL, * TMQ
높은 큐 용량 임계값....  80 0-100, * SYSDFTQ
큐 용량 하한 임계값....  20 0-100, * SYSDFTQ
큐 가득 참 이벤트가 사용 가능합니다.  * 예 * SYSDFTQ, * NO, * 예
높은 큐 이벤트가 사용 가능합니다.  * 예 * SYSDFTQ, * NO, * 예
큐가 낮은 이벤트를 사용할 수 있습니다....  * 예 * SYSDFTQ, * NO, * 예
서비스 간격........  999999999 0-999999999, * SYSDFTQ
서비스 간격 이벤트.....  * 없음 * SYSDFTQ, * HIGH, * OK, * 없음
분배 리스트 지원.....  * NO * SYSDFTQ, * NO, * YES
클러스터 이름..........  * SYSDFTQ
클러스터 이름 목록........  * SYSDFTQ
기본 바인딩........  * SYSDFTQ * SYSDFTQ, * OPEN, * NOTFIXED




Bottom
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *20.* 큐 작성*(4)*

필드에 올바른 데이터가 포함된 것으로 만족하면 Enter를 눌러 큐를 작성하십시오.

채널 시작

소스 큐 관리자에 정의되어 있는 리모트 큐에 메시지를 넣으면 채널을 시작할 때까지 전송 큐에 저장됩니다. 채널이 시작되면 메시지가 리모트 큐 관리자의 대상 큐로 전달됩니다.

START CHANNEL 명령을 사용하여 송신 큐 관리자에서 채널을 시작하십시오. 송신 채널을 시작하면, 수신 채널이 자동으로(리스너에 의해) 시작되며 메시지가 대상 큐로 송신됩니다. 메시지를 전송하려면 메시지 채널의 양측이 실행 중이어야 합니다.

채널의 양측이 서로 다른 큐 관리자에 있으므로 서로 다른 속성으로 정의되어 있을 수 있습니다. 차이점을 해결하기 위해, 채널이 시작할 때 양쪽 사이에 초기 데이터 조정이 이루어집니다. 일반적으로 채널의 양측은 보다 적은 자원이 필요한 속성으로 작동합니다. 따라서 보다 큰 시스템이 메시지 채널의 다른 한쪽에서 작은 시스템의 적은 자원을 수용할 수 있습니다.

송신 MCA는 채널을 통해 대형 메시지를 송신하기 전에 이를 작게 분할합니다. 그리고 리모트 큐 관리자에서 이 메시지를 다시 조합합니다. 이는 사용자에게 표시되지 않습니다.

MCA는 다중 스레드를 사용하여 메시지를 전송할 수 있습니다. 파이프라이닝이라고 하는 이 프로세스는 MCA가 더 적게 대기하면서 더 효율적으로 메시지를 전송할 수 있도록 합니다. 파이프라이닝은 채널 성능을 향상시킵니다.

채널 제어 기능

채널 제어 기능은 채널을 정의, 모니터, 제어할 수 있는 기능을 제공합니다.

명령은 패널이나 프로그램을 통해 또는 명령행에서 채널 제어 기능으로 실행됩니다. 패널 인터페이스에는 또한 채널 상태와 채널 정의 데이터가 표시됩니다. 프로그래밍 가능 명령 형식 또는 해당 IBM MQ 명령(MQSC) 및 제어 명령(175 페이지의 『UNIX, Linux, and Windows에서 채널 모니터링 및 제어』의 설명 참조)을 사용할 수 있습니다.

명령은 다음 그룹으로 분류됩니다.

- 채널 관리
- 채널 제어
- 채널 상태 모니터링

채널 관리 명령은 채널 정의를 처리합니다. 이 명령은 다음 기능을 수행합니다.

- 채널 정의 작성
- 채널 정의 복사
- 채널 정의 대체
- 채널 정의 삭제

채널 제어 명령은 채널 조작을 관리합니다. 이 명령은 다음 기능을 수행합니다.

- 채널 시작
- 채널 중지
- 파트너와 재동기화(일부 구현에서)
- 메시지 순서 번호 재설정
- 메시지의 인다우트(in-doubt) 배치 해석
- ping, 채널에서 테스트 통신 송신

채널 모니터링은 다음과 같은 채널의 상태를 표시합니다.

- 현재 채널 설정
- 채널이 활성 또는 비활성인지 여부
- 채널이 동기화된 상태로 종료되었는지 여부

채널 정의, 제어, 모니터링에 대한 자세한 정보는 다음 하위 주제를 참조하십시오.

채널 준비

메시지 채널 또는 MQI 채널을 시작하려고 시도하기 전에 채널을 준비해야 합니다. 로컬 및 원격 채널 정의의 모든 속성이 올바르고 호환 가능한지 확인해야 합니다.

채널 속성은 채널 정의와 속성을 설명합니다.

명시적 채널 정의를 설정하더라도 채널이 시작될 때 수행된 채널 조정은 정의된 하나 이상의 다른 값을 대체할 수 있습니다. 이 동작은 정상적인 것으로 사용자에게 표시되지 않으며 호환되지 않는 정의가 함께 작동할 수 있도록 이 방식으로 배열되었습니다.

수신자 및 서버 연결 채널의 자동 정의

z/OS를 제외한 모든 플랫폼의 경우 IBM MQ에 적합한 채널 정의가 없으면 자동 정의를 사용할 수 있는 송신자 또는 서버 연결 채널에 대한 정의가 자동으로 작성됩니다. 정의는 다음을 사용하여 작성됩니다.

1. 적합한 모델 채널 정의, SYSTEM.AUTO.RECEIVER 또는 SYSTEM.AUTO.SVRCONN. 자동 정의를 위한 모델 채널 정의는 설명 필드("Auto-defined by" 다음에 49개 공백이 옴)를 제외하고는 시스템 기본값인 SYSTEM.DEF.RECEIVER, SYSTEM.DEF.SVRCONN과 동일합니다. 시스템 관리자는 제공된 모델 채널 정의의 일부를 변경할 수 있습니다.
2. 파트너 시스템에서 얻은 정보. 파트너의 값은 채널 이름과 순서 번호 줄바꿈 값에 사용됩니다.
3. 자동 정의로 작성된 값을 대체하는 데 사용할 수 있는 채널 엑시트 프로그램. 채널 자동 정의 엑시트 프로그램을 참조하십시오.

그런 다음 설명을 확인하여 자동 정의 엑시트에 의해 대체되었는지 또는 모델 정의가 변경되었는지 여부를 판별합니다. 처음 44개 문자가 계속 "Auto-defined by"와 그 다음의 29개 공백인 경우에는 큐 관리자 이름이 추가됩니다. 마지막 20자가 계속 모두 공백이면 현지 시간과 날짜가 추가됩니다.

정의가 작성 및 저장되면 정의가 항상 존재한 것처럼 채널 시작이 진행됩니다. 배치 크기, 전송 크기, 메시지 크기는 파트너와 함께 조정됩니다.

다른 오브젝트 정의

메시지 채널을 시작하려면 해당 큐 관리자에서 양쪽이 모두 정의되어야 합니다(또는 자동 정의에 사용 가능). 지원하는 전송 큐는 송신측에서 큐 관리자에 정의되어야 합니다. 통신 링크가 정의되고 사용 가능해야 합니다. 리모트 큐 정의, 큐 관리자 알리어스 정의, 응답 대상 큐 알리어스 정의와 같은 다른 IBM MQ 오브젝트가 128 페이지의 『분산 큐잉 구성』에 설명된 시나리오를 구현해야 할 수 있습니다.

MQI 채널 정의에 대한 정보는 27 페이지의 『MQI 채널 정의』의 내용을 참조하십시오.

전송 큐당 복수 메시지 채널

전송 큐당 여러 채널을 정의할 수 있지만 한 번에 해당 채널 중 하나만 활성화될 수 있습니다. 트래픽 밸런싱 및 링크 실패 정정 조치를 위해 큐 관리자 간에 대체 라우트를 제공하는 경우 이 옵션을 고려하십시오. 전송 큐를 사용하려는 이전의 채널이 송신측에서 메시지 배치가 인다우트(in-doubt)인 채로 종료되면 다른 채널에서 전송 큐를 사용할 수 없습니다. 추가 정보는 165 페이지의 『인다우트(in-doubt) 채널』의 내용을 참조하십시오.

채널 시작

채널을 사용하여 다음 네 가지 방법 중 하나로 메시지 전송을 시작할 수 있습니다. 채널은 다음과 같습니다.

- 운영자가 시작할 수 있습니다. 수신자, 클러스터 수신자 또는 서버 연결 채널은 시작할 수 없습니다.
- 전송 큐에서 트리거될 수 있습니다. 이 방법은 송신자 채널과 완전히 규정된 서버 채널(CONNAME을 지정하는 채널)에만 적용됩니다. 채널 트리거에 필요한 오브젝트를 준비해야 합니다.
- 애플리케이션 프로그램에서 시작할 수 있습니다. 수신자, 클러스터 수신자 또는 서버 연결 채널은 시작할 수 없습니다.
- 송신자, 클러스터 송신자, 요청자, 서버 또는 클라이언트 연결 채널에 의해 네트워크로부터 원격으로 시작됩니다. 수신자, 클러스터 수신자, 서버 및 요청자 채널 전송은 이 방식으로 시작됩니다. 서버 연결 채널도 마찬가지입니다. 채널은 이미 시작된 상태여야 합니다(즉, 사용 가능한 상태).

참고: 채널이 '시작'되었으므로 반드시 메시지를 전송하지는 않습니다. 대신 이전에 설명한 네 가지 이벤트 중 하나가 발생할 때 전송을 시작할 수 있도록 '설정'될 수 있습니다. 채널 사용과 사용 불가능은 START 및 STOP 운영자 명령을 사용하여 수행됩니다.

채널 상태

채널은 언제든지 여러 상태 중 하나일 수 있습니다. 일부 상태에는 하위 상태도 있습니다. 채널은 제공된 상태에서 다른 상태로 이동할 수 있습니다.

157 페이지의 그림 21에서는 가능한 모든 채널 상태의 계층과 각 채널 상태에 적용되는 하위 상태를 보여줍니다.

158 페이지의 그림 22에서는 채널 상태 간 링크를 보여줍니다. 이들 링크는 모든 유형의 메시지 채널과 서버 연결 채널에 적용됩니다.

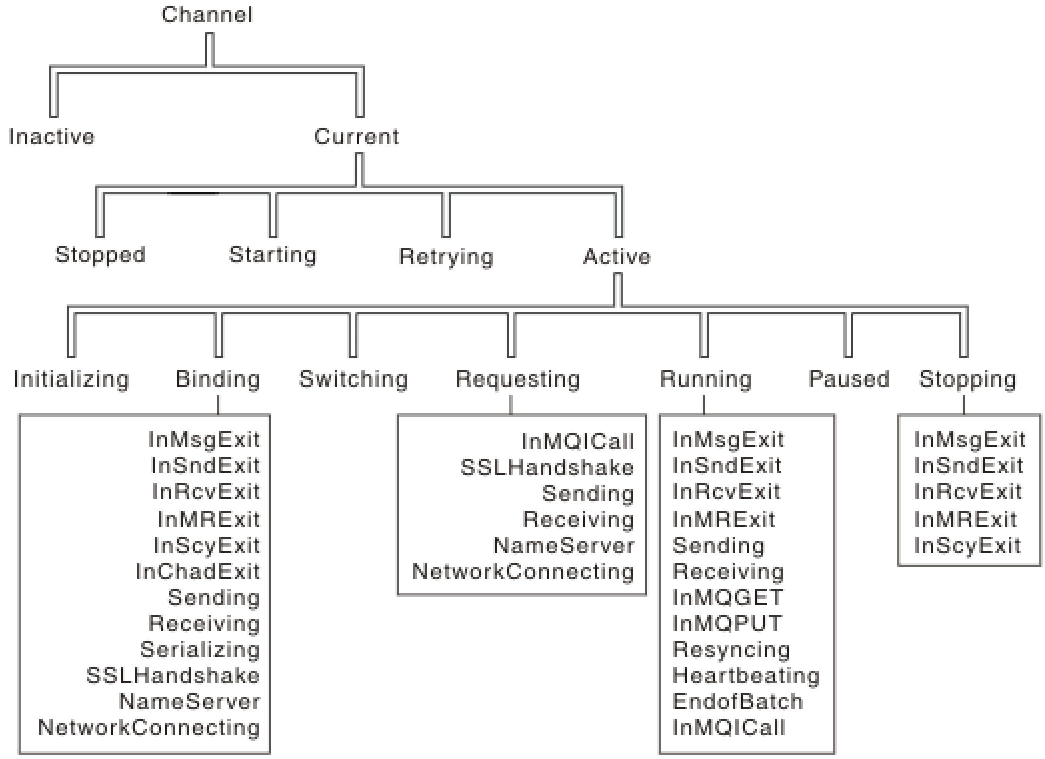


그림 *21.* 채널 상태 및 하위 상태

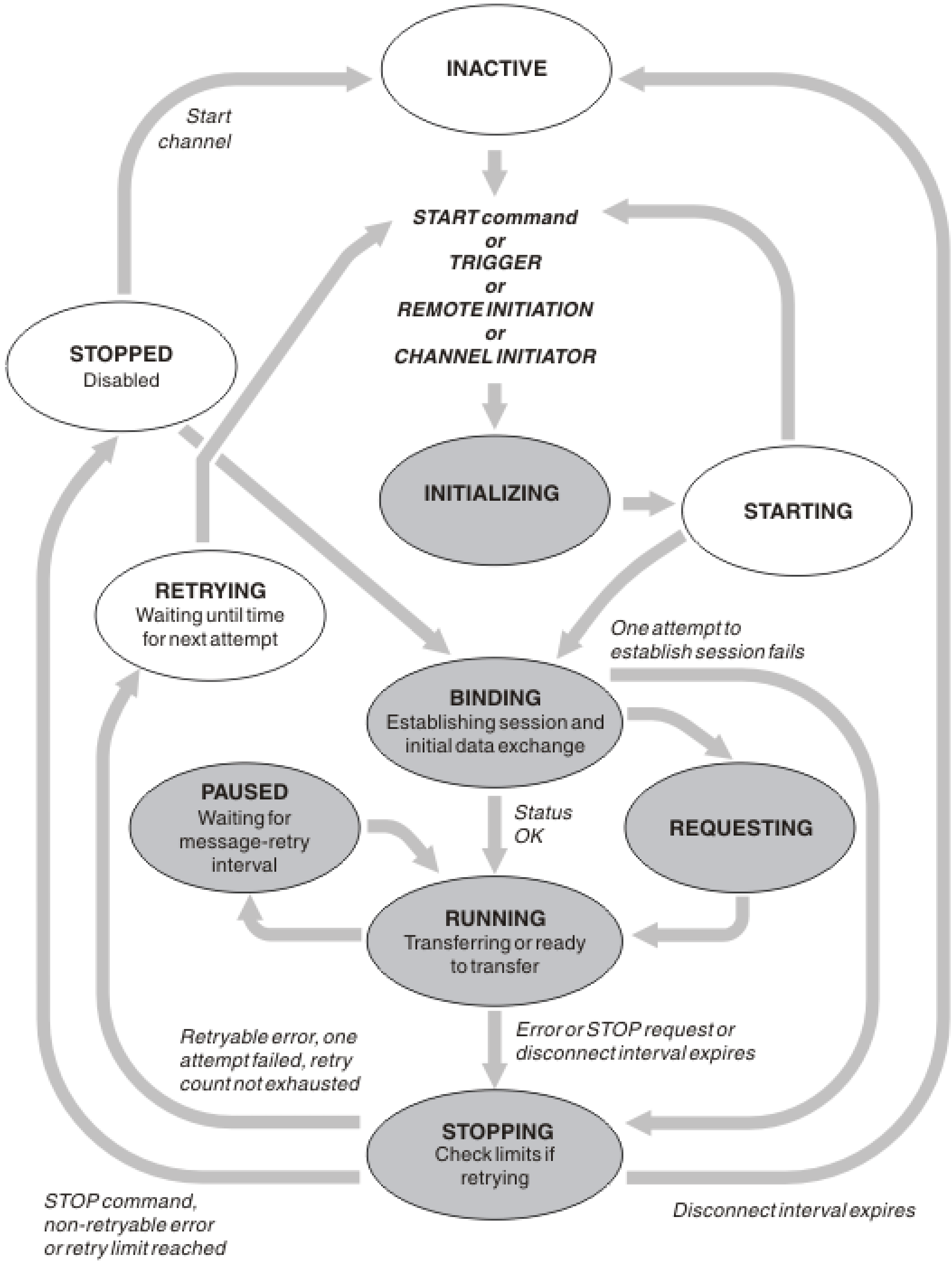


그림 *22.* 채널 상태 간 플로우

현재 및 활성

비활성 이외의 상태인 경우 채널은 현재 상태입니다. RETRYING, STOPPED 또는 STARTING 상태에 있지 않는 한 현재 채널은 활성입니다. 채널이 활성 상태이면 자원을 이용하고 프로세스나 스레드가 실행 중입니다. 활성 채널의 일곱 가지 가능한 상태(INITIALIZING, BINDING, SWITCHING, REQUESTING, RUNNING, PAUSED 또는 STOPPING)가 158 페이지의 그림 22에 강조표시되어 있습니다.

활성 채널은 채널이 수행하고 있는 작업의 자세한 정보를 정확하게 제공하는 하위 상태를 표시할 수도 있습니다. 각 상태의 하위 상태가 157 페이지의 그림 21에 나와 있습니다.

현재 및 활성
비활성 이외의 상태인 경우 채널은 "현재" 상태입니다. RETRYING, STOPPED 또는 STARTING 상태에 있지 않는 한 현재 채널은 "활성"입니다.

"활성" 채널은 채널이 수행하고 있는 내용의 세부사항을 자세히 제공하는 하위 상태를 표시할 수도 있습니다.

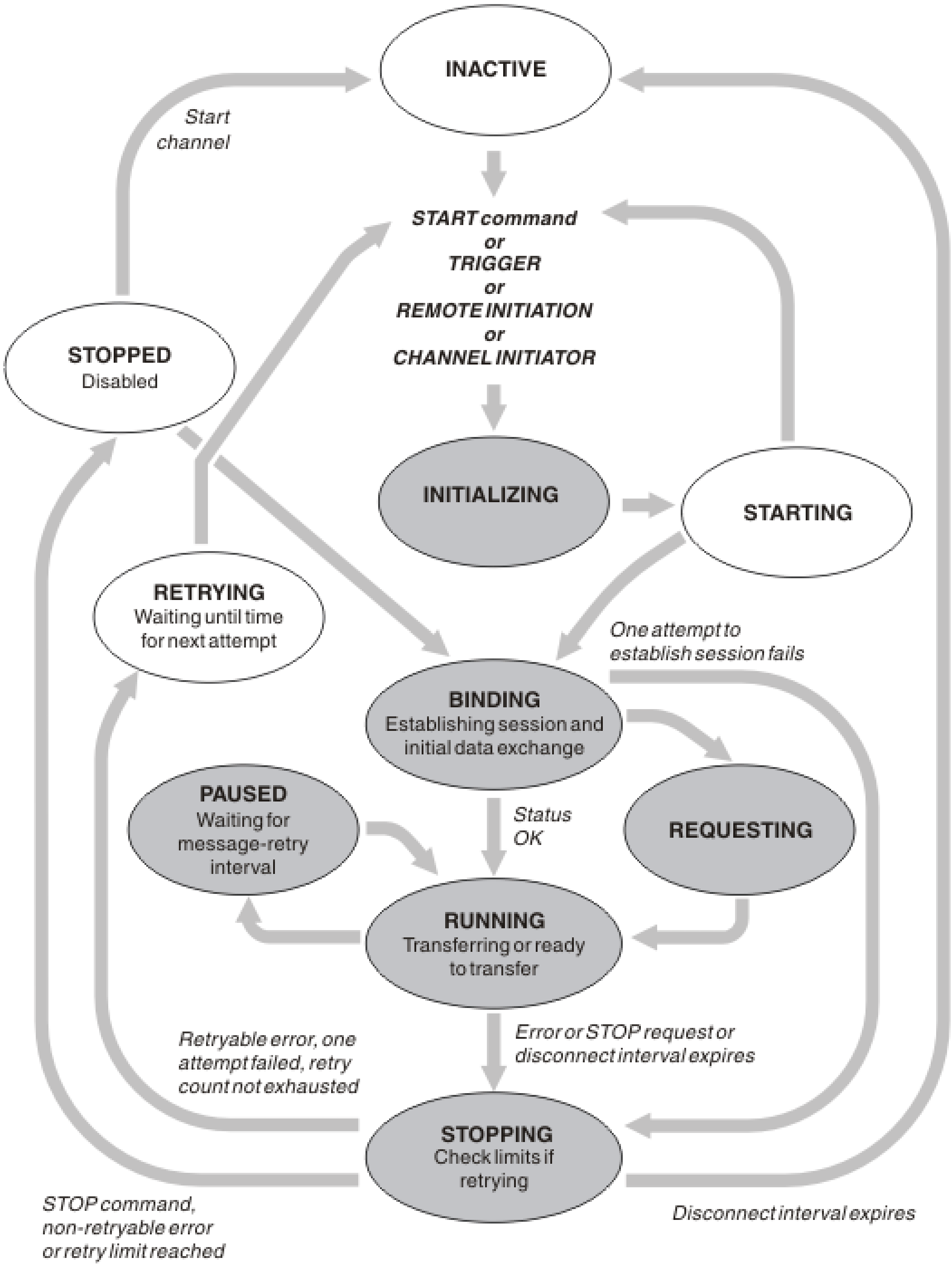


그림 *23.* 채널 상태 간 플로우

참고:

1. 채널이 159 페이지의 그림 23에서 강조표시된 6가지 상태(INITIALIZING, BINDING, REQUESTING, RUNNING, PAUSED 또는 STOPPING) 중 하나이면 자원을 이용하고 프로세스나 스레드가 실행 중인 상태입니다. 즉, 채널이 활성 상태입니다.
2. 채널이 STOPPED 상태이면 다음 상태를 아직 알지 못해 세션이 활성 상태일 수 있습니다.

최대 현재 채널 수 지정

한 번에 현재 상태일 수 있는 최대 채널 수를 지정할 수 있습니다. 이 수는 재시도 중인 채널과 중지된 채널을 포함하여 채널 상태 테이블에 항목이 있는 채널의 수입니다. 플랫폼에 대해 다음을 지정하십시오.

- z/OS ALTER QMGR MAXCHL 명령을 사용하십시오.
- IBM i 큐 관리자 초기화 파일을 편집하십시오.
- Linux UNIX 큐 관리자 구성 파일을 편집하십시오.
- IBM MQ Explorer를 사용합니다.

초기화 또는 구성 파일을 사용하여 설정되는 값에 대한 자세한 정보는 분산 큐잉에 대한 구성 파일 스탠자를 참조하십시오. 최대 채널 수 지정에 대한 자세한 정보는 다음 주제를 참조하십시오.

- ULW IBM MQ 관리.
- IBM i IBM MQ for IBM i 관리.
- z/OS IBM MQ for z/OS 관리.

참고:

1. 이 수에는 서버 연결 채널이 포함됩니다.
2. 채널이 활성화되려면 먼저 현재 상태여야 합니다. 채널이 시작되었지만 현재가 될 수 없으면 시작이 실패합니다.

최대 활성 채널 수 지정

또한 많은 시작 채널로 인한 시스템 과부하를 방지하기 위해 최대 활성 채널 수를 지정할 수 있습니다. 이 방법을 사용하는 경우, 연결 끊기 간격 속성을 낮은 값으로 설정하여 다른 채널이 종료되는 즉시 대기 채널이 시작될 수 있도록 하십시오.

재시도하는 채널이 파트너와의 연결 설정 시도할 때마다 활성 채널이 되어야 합니다. 시도가 실패하면 다음 시도 전까지 활성 상태가 아닌 현재 채널로 남게 됩니다. 채널 재시도 횟수와 빈도는 재시도 횟수 및 재시도 간격 채널 속성으로 결정됩니다. 이 속성에는 둘 다 짧은 값과 긴 값이 있습니다. 자세한 정보는 채널 속성을 참조하십시오.

START 명령이 실행 또는 트리거되었거나 다른 재시도 시간이 되어 채널이 활성 채널이 되었지만 활성 채널이 이미 최대값에 도달하여 활성화될 수 없는 경우, 채널은 다른 채널 인스턴스가 활성화를 멈추어 활성 슬롯이 비워질 때까지 대기합니다. 그러나 채널이 시작되고(원격으로 시작되기 때문에) 해당 시점에 사용 가능한 활성 슬롯이 없으면 원격 시작이 거부됩니다.

요청자 채널 이외의 채널이 활성화를 시도할 때마다 때마다 STARTING 상태가 됩니다. 이 상태는 즉시 사용 가능한 활성 슬롯이 있는 경우에도 발생하며 STARTING 상태는 잠시 동안만 유지됩니다. 그러나 채널이 활성 슬롯을 대기해야 하는 경우에는 대기 중에 STARTING 상태가 됩니다.

요청자 채널은 시작 상태가 되지 않습니다. 활성 채널 수가 이미 한계에 도달하여 요청자 채널을 시작할 수 없는 경우에는 채널이 비정상적으로 종료됩니다.

요청자 채널을 제외한 채널이 활성 슬롯을 가져올 수 없어 대기할 때마다 로그 z/OS 또는 z/OS 콘솔에 메시지가 기록되며 이벤트가 생성됩니다. 나중에 슬롯이 비워져 채널이 확보할 수 있으면 다른 메시지와 이벤트가 생성됩니다. 채널이 슬롯을 즉시 확보할 수 없으면 이러한 이벤트와 메시지 모두 생성되지 않습니다.

채널이 활성화될 때까지 대기하는 동안 STOP CHANNEL 명령을 실행하면 채널이 STOPPED 상태가 됩니다. 채널-중지 이벤트가 생성됩니다.

서버 연결 채널은 최대 활성 채널 수에 포함됩니다.

최대 활성 채널 수 지정에 대한 자세한 정보는 다음 주제를 참조하십시오.

- ULW IBM MQ 관리.
- IBM i IBM MQ for IBM i 관리.
- z/OS IBM MQ for z/OS 관리.

채널 오류
채널 오류가 발생하면 채널이 더 이상 전송 작업을 수행하지 않습니다. 채널이 송신자 또는 서버인 경우에는 문제점이 명확할 수 있으므로 재시도 상태로 전환됩니다. 재시도 상태로 전환할 수 없는 경우에는 채널이 STOPPED 상태가 됩니다.

송신 채널의 경우, 연관된 전송 큐가 GET(DISABLED)로 설정되고 트리거가 꺼집니다. (STATUS(STOPPED) 상태의 STOP 명령은 명령을 실행한 측을 STOPPED 상태로 전환합니다. 연결 끊기 간격이 만료되거나 STATUS(INACTIVE) 상태의 STOP 명령은 정상적으로 종료하여 비활성화됩니다.) STOPPED 상태의 채널은 운영자가 개입해야 재시작할 수 있습니다(165 페이지의 『중지된 채널 재시작』 참조).

참고: IBM i IBM i, UNIX, Linux, and Windows 시스템의 경우 재시도하려면 채널 시작기가 실행 중이어야 합니다. 채널 시작기를 사용할 수 없으면 채널이 비활성화되므로 수동으로 재시작해야 합니다. 스크립트를 사용하여 채널을 시작하는 경우에는 스크립트 실행을 시도하기 전에 채널 시작기가 실행 중인지 확인하십시오.

긴 재시도 횟수(LONGRTY)는 재시도 과정을 설명합니다. 오류가 제거되면 채널은 자동으로 재시작되며, 전송 큐를 다시 사용할 수 있습니다. 오류를 제거하지 않고 재시도 한계에 도달하면, 채널이 STOPPED 상태가 됩니다. 중지된 채널은 운영자가 수동으로 재시작해야 합니다. 오류가 해결되지 않으면 다시 재시도하지 않습니다. 성공적으로 시작되면 전송 큐를 다시 사용합니다.

z/OS 채널이 RETRYING 또는 STOPPED 상태일 때 채널 시작기가 중지되면 채널 시작기가 재시작될 때 채널 상태가 기억됩니다. 그러나 채널이 STOPPED 상태일 때 채널 시작기가 중지되면 SVRCONN 채널 유형의 채널 상태가 재설정됩니다.

Multi 채널이 RETRYING 또는 STOPPED 상태일 때 큐 관리자가 중지되면 큐 관리자가 재시작될 때 채널 상태가 기억됩니다. IBM MQ 8.0 이상에서는 SVRCONN 채널에도 적용됩니다. 이전에는 채널이 STOPPED 상태일 때 채널 시작기가 중지되면 SVRCONN 채널 유형에 대한 채널 상태가 재설정되었습니다.

대상 큐가 가득 차거나 금지 상태가 되어 채널이 해당 큐에 메시지를 넣을 수 없으면 채널이 시간 간격(메시지 재시도 간격 속성에 지정)에 따라 조작을 여러 번(메시지 재시도 횟수 속성에 지정됨) 재시도할 수 있습니다. 또는 재시도를 야기하는 상황과 시도 횟수를 판별하는 메시지 재시도 엑시트를 작성할 수 있습니다. 채널은 메시지 재시도 간격이 완료할 때까지 기다리는 동안 일시정지 상태가 됩니다.

채널 속성에 대한 정보는 채널 속성을, 메시지 재시도 엑시트에 대한 정보는 메시징 채널을 위한 채널 엑시트 프로그램을 참조하십시오.

서버 연결 채널 한계

서버 연결 채널 한계를 설정하여 클라이언트 애플리케이션이 큐 관리자 채널 자원, **MAXINST**를 이용하는 것을 방지하고 단일 클라이언트 애플리케이션이 서버 연결 채널 용량, **MAXINSTC**를 이용하는 것을 방지할 수 있습니다.

MAXINST 및 **MAXINSTC** 을 **DEFINE CHANNEL** 명령으로 설정합니다.

개별 큐 관리자에서 언제라도 최대 총 채널 수가 활성화될 수 있습니다. 총 서버 연결 채널 인스턴스 수는 최대 활성 채널 수에 포함됩니다.

시작될 수 있는 서버 연결 채널의 최대 동시 인스턴스 수를 지정하지 않으면 단일 클라이언트 애플리케이션이 단일 서버 연결 채널에 연결되어 사용 가능한 최대 활성 채널 수를 이용할 수 있습니다. 최대 활성 채널 수에 도달하면 다른 채널이 큐 관리자에서 시작될 수 없습니다. 이러한 상황을 피하려면 채널을 시작한 클라이언트에 관계 없이, 시작될 수 있는 개별 서버 연결 채널의 동시 인스턴스의 수를 제한해야 합니다.

한계 값이 현재 실행 중인 서버 연결 채널의 인스턴스 수 미만으로 감소하면(0까지) 실행 중인 채널이 영향을 받지 않습니다. 충분한 수의 기존 인스턴스가 실행을 중지하여 현재 실행 중인 인스턴스 수가 한계 값 미만이 될 때까지 새 인스턴스가 시작될 수 없습니다.

또한 다수의 서로 다른 클라이언트 연결 채널을 개별 서버 연결 채널에 연결할 수 있습니다. 채널을 시작한 클라이언트에 관계 없이 시작될 수 있는 개별 서버 연결 채널의 동시 인스턴스 수 한계로 인해 클라이언트가 큐 관리자의 최대 활성 채널 용량을 이용할 수 없게 됩니다. 또한 개별 클라이언트에서 시작될 수 있는 개별 서버 연결 채널의 동시 인스턴스 수를 제한하지 않으면, 결함이 있는 단일 클라이언트 애플리케이션이 너무 많은 연결을 열어 개별 서버 연결 채널에 할당된 채널 용량을 이용하고 결과적으로 채널을 사용해야 하는 다른 클라이언트가 연결하지 못하게 됩니다. 이러한 상황을 피하려면 개별 클라이언트에서 시작될 수 있는 개별 서버 연결 채널의 동시 인스턴스의 수를 제한해야 합니다.

개별 클라이언트 한계 값이 현재 개별 클라이언트에서 실행 중인 서버 연결 채널의 인스턴스 수 미만으로 감소하면(0까지) 실행 중인 채널이 영향을 받지 않습니다. 그러나 해당 클라이언트로부터의 충분한 수의 기존 인스턴스

가 실행을 중지하여 현재 실행 중인 인스턴스 수가 이 매개변수의 값보다 적어질 때까지 새 한계를 초과하는 개별 클라이언트에서 서버 연결 채널의 새 인스턴스를 시작할 수 없습니다.

관련 정보

채널 속성 및 채널 유형

DEFINE CHANNEL

채널의 다른 쪽이 여전히 사용 가능한지 검사

하트비트 간격, 활성 유지(keepalive) 간격, 수신 제한시간을 사용하여 채널의 다른 한 쪽을 사용할 수 있는지 확인할 수 있습니다.

하트비트

하트비트 간격 채널 속성을 사용하여 전송 큐에 메시지가 없을 때 송신 MCA에서 플로우가 전달되는 것으로 지정할 수 있습니다(하트비트 간격(HBINT)의 설명 참조).

활성 유지(keepalive)

IBM MQ for z/OS에서 TCP/IP를 전송 프로토콜로 사용하는 경우, 또한 **Keepalive** 간격 채널 속성(KAINT)에 대한 값을 지정할 수 있습니다. **Keepalive** 간격에 하트비트 간격보다 큰 값을 지정하고 연결 끊기 값보다 작은 값을 지정하는 것이 좋습니다. 이 속성을 사용하여 각 채널의 제한시간 값을 지정할 수 있습니다(활성 유지(keepalive) 간격(KAINT)의 설명 참조).

IBM MQ for IBM i, UNIX, Linux, and Windows 시스템에서 TCP를 전송 프로토콜로 사용하는 경우 `keepalive=yes`를 설정할 수 있습니다. 이 옵션을 지정하면 TCP가 연결의 다른 한 쪽을 계속 사용할 수 있는지 주기적으로 확인합니다. 사용할 수 없으면 채널이 종료됩니다. 이 옵션에 대한 설명은 활성 유지(keepalive) 간격(KAINT)을 참조하십시오.

TCP 오류를 보고하는 신뢰할 수 없는 채널이 있는 경우 **Keepalive** 옵션을 사용하면 채널을 복구할 수 가능성이 높음을 의미합니다.

Keepalive 옵션 작동을 제어하는 시간 간격을 지정할 수 있습니다. 시간 간격을 변경하면 변경 이후 시작된 TCP/IP 채널만 영향을 받습니다. 시간 간격에 선택하는 값은 채널의 연결 끊기 간격 값보다 작아야 합니다.

Keepalive 옵션 사용에 대한 자세한 정보는 DEFINE CHANNEL 명령의 KAINT 매개변수를 참조하십시오.

수신 제한시간

TCP를 전송 프로토콜로 사용하는 경우 일정 기간 동안 데이터가 수신되지 않으면 유휴 비MQI 채널 연결의 수신 측 또한 닫힙니다. 이 기간에 해당하는 *receive time-out* 값은 HBINT(하트비트 간격) 값에 따라 판별됩니다.

IBM MQ for IBM i, UNIX, Linux, and Windows 시스템에서 *receive time-out* 값은 다음과 같이 설정됩니다.

1. 초기 플로우 수의 경우, 조정이 수행되기 전 *receive time-out* 값은 채널 정의의 HBINT 값의 두 배입니다.
2. HBINT가 60초 미만으로 설정되는 경우 채널이 HBINT 값을 조정하면 *receive time-out* 값이 이 값의 두 배로 설정됩니다. HBINT가 60초 이상으로 설정되면 *receive time-out* 값이 HBINT 값보다 60초 더 큰 값으로 설정됩니다.

IBM MQ for z/OS의 경우 *receive time-out* 값은 다음과 같이 설정됩니다.

1. 초기 플로우 수의 경우, 조정이 수행되기 전 *receive time-out* 값은 채널 정의의 HBINT 값의 두 배입니다.
2. RCVTIME이 설정되면 제한시간이 다음 중 하나로 설정됩니다.
   - 상수를 곱한 조정 HBINT
   - 조정된 HBINT + 상수(초)
   - 상수(초)

   이 경우 제한시간은 RCVTTYPE 매개변수에 따라 다르며 적용되는 경우 RCVTMIN으로 부과되는 한계가 적용됩니다. RCVTTYPE(EQUAL)이 구성된 경우에는 RCVTMIN이 적용되지 않습니다. RCVTIME의 상수값과 하트비트 간격을 사용하는 경우에는 RCVTIME을 하트비트 간격 미만으로 지정하지 마십시오. RCVTIME, RCVTMIN, RCVTTYPE 속성에 대한 세부사항은 ALTER QMGR 명령을 참조하십시오.

참고:

1. 다음 값 중 하나가 0이면 제한시간이 없습니다.
2. 하트비트를 지원하지 않는 연결의 경우, HBINT 값이 2단계에서 0으로 조정되어 제한시간이 없으므로 TCP/IP KEEPALIVE를 사용해야 합니다.
3. 공유 대화를 사용하는 클라이언트 연결의 경우, 하트비트가 MQGET이 미해결 상태인 경우뿐 아니라 항상 채널 양측에서 전달될 수 있습니다.
4. 공유 대화를 사용하지 않는 클라이언트 연결의 경우, 클라이언트가 대기와 함께 MQGET 호출을 발행하는 경우에만 서버에서 하트비트가 전달됩니다. 따라서 클라이언트 채널의 경우 하트비트 간격을 너무 작게 설정하는 것은 바람직하지 않습니다. 예를 들어, 하트비트가 10초로 설정되면 커미트 시간이 20초보다 길게 소요되는 경우 MQRC_CONNECTION_BROKEN과 함께 MQCMIT 호출이 실패합니다. 이 시간 동안 데이터가 전달되지 않기 때문입니다. 이는 작업 단위가 큰 경우에 해당할 수 있습니다. 그러나 대기가 있는 MQGET만 상당한 시간이 소요되므로 하트비트 간격에 적절한 값을 선택하면 이에 해당되지 않습니다.

   SHARECNV가 0이 아니면 클라이언트가 양방향 전송 연결을 사용하며 이는 클라이언트가 모든 MQI 호출 중에 하트비트를 수행할 수 있고 실제로 수행함을 의미합니다.
5. IBM WebSphere MQ 7 클라이언트 채널에서는 서버 측과 클라이언트 측에서 모두 하트비트가 전달될 수 있습니다. 양측의 제한시간은 2*HBINT(HBINT가 60초 미만인 경우) 및 HBINT+60(HBINT가 60초보다 긴 경우)을 기반으로 합니다.
6. 최소한 모든 하트비트 간격에서 데이터 또는 하트비트 플로우가 예상되므로 두 배의 하트비트 간격 후 연결을 취소하는 것은 올바르지 않습니다. 그러나 하트비트 간격을 너무 작게 설정하면 특히 채널 엑시트를 사용하는 경우 문제가 발생할 수 있습니다. 예를 들어, HBINT 값이 1초이고 송신 또는 수신 엑시트를 사용하는 경우에는 채널을 취소하기 전에 2초 동안만 수신측이 대기합니다. MCA가 메시지 암호화와 같은 태스크를 수행하는 경우에는 이 값이 너무 짧을 수도 있습니다.

제안된 설정

IBM MQ for z/OS

초기 시작점으로 다음을 사용할 수 있습니다.

```
/cpf ALTER QMGR TCPKEEP(YES) RCVTTYPE(ADD) RCVTIME(60) ADOPTMCA(ALL) ADOPTCHK(ALL)
```

여기서, `cpf`는 큐 관리자 서브시스템의 명령 접두부입니다.

다양한 매개변수에 대한 자세한 정보는 ALTER QMGR 및 IBM MQ 네트워크 가용성을 참조하십시오.

송신자의 IP 주소가 둘 이상의 주소로 변환될 수 있는 경우, ADOPTCHK를 ALL이 아닌 QMNAME으로 설정해야 할 수 있습니다.

IBM MQ for Multiplatforms

`qm.ini`에서 다음 정보를 추가하십시오.

```
TCP:
KeepAlive=Yes
CHANNELS:
AdoptNewMCA=ALL
AdoptNewMCACheck=ALL
```

자세한 정보는 ALTER QMGR, 분산 큐잉을 위한 구성 파일 스탠자 및 111 페이지의 『Channels 스탠자의 속성』의 내용을 참조하십시오.

송신자의 IP 주소가 둘 이상의 주소로 변환될 수 있는 경우, **AdoptNewMCACheck** 를 `ALL`이 아닌 `QMNAME` 으로 설정해야 합니다.

MCA 채택

IBM MQ는 MCA 채택 기능으로 수신자 채널을 취소하고 올바른 위치에서 새 채널을 시작할 수 있습니다.

채널 접속이 끊어지면 수신자 채널이 "통신 수신" 상태가 될 수 있습니다. 통신이 재설정되면 송신자 채널이 다시 연결을 시도합니다. 리모트 큐 관리자가 수신자 채널이 이미 실행 중인 사실을 발견하면 동일한 수신자 채널의 다른 버전이 시작되도록 허용하지 않습니다. 이 문제를 해결하려면 사용자가 개입하여 문제를 수정하거나 시스템 활성 유지(keepalive)를 사용해야 합니다.

MCA 채택 기능은 문제점을 자동으로 해결합니다. IBM MQ는 이 기능으로 수신자 채널을 취소하고 올바른 위치에서 새 채널을 시작할 수 있습니다.

관련 정보

IBM MQ 관리

IBM MQ for z/OS 관리

IBM MQ for IBM i 관리

채널 중지 및 일시정지

연결 끊기 시간 간격이 만기되기 전에 채널을 중지하고 일시정지할 수 있습니다.

메시지 채널은 큐 관리자 간에 오래 실행되는 연결이 되도록, 또한 질서 있는 종료는 연결 끊기 간격 채널 속성으로만 제어되도록 설계되었습니다. 이 메커니즘은 연결 끊기 시간 간격이 만기되기 전에 운영자가 채널을 종료하지 않아도 되는 경우에도 적용됩니다. 이러한 요구는 다음 상황에서 발생할 수 있습니다.

- 시스템 일시정지
- 자원 유지
- 채널의 한쪽 끝에서의 일방적인 조치

이 경우 채널을 중지할 수 있습니다. 다음을 사용하여 이를 수행할 수 있습니다.

- STOP CHANNEL MQSC 명령
- Stop Channel PCF 명령
- IBM MQ 탐색기
- IBM i z/OS 기타 플랫폼별 메커니즘은 다음과 같습니다.

  z/OS **z/OS의 경우:**

  채널 중지 패널

  IBM i **IBM i의 경우:**

  ENDMQMCHL CL 명령 또는 WRKMQMCHL 패널의 END 옵션

이 명령을 사용한 채널 중지 옵션은 다음과 같은 세 가지가 있습니다.

QUIESCE

QUIESCE 옵션은 채널을 중지하기 전에 현재 메시지 배치를 종료하려고 시도합니다.

FORCE

FORCE 옵션은 채널을 즉시 종료하려고 시도하며, 채널이 인다우트(in-doubt) 상태로 남아 있을 수 있으므로 재시작 시 채널을 재동기화해야 할 수 있습니다.

z/OS IBM MQ for z/OS의 경우, FORCE는 진행 중인 메시지 재할당을 방해하여 BIND_NOT_FIXED 메시지가 부분적으로 재할당되거나 적용되지 않을 수 있습니다.

TERMINATE

TERMINATE 옵션은 채널을 즉시 중지하려고 시도하며 채널의 스레드 또는 프로세스를 종료합니다.

z/OS IBM MQ for z/OS의 경우, TERMINATE는 진행 중인 메시지 재할당을 방해하여 BIND_NOT_FIXED 메시지가 부분적으로 재할당되거나 적용되지 않을 수 있습니다.

이러한 옵션은 모두 채널을 STOPPED 상태로 유지하므로 재시작하려면 운영자가 개입해야 합니다.

송신측에서 채널을 중지할 수 있지만 재시작하려면 운영자 개입이 필요합니다. 채널의 수신측에서는 MCA가 송신측에서 들어오는 데이터를 기다리고 수신측에서 질서 있는 채널 종료를 시작할 수 없으므로 상황이 훨씬 더 어려워집니다. 중지 명령은 MCA가 데이터 대기에서 리턴할 때까지 보류 상태가 됩니다.

결과적으로 필요한 조작 특성에 따라 다음과 같은 세 가지 채널 사용 방법이 권장됩니다.

- 채널을 오래 실행하려면 송신측의 질서 있는 종료만 존재할 수 있다는 점에 유의하십시오. 채널이 인터럽트, 즉 중지되는 경우 재시작하려면 운영자 개입(START CHANNEL 명령)이 필요합니다.
- 전송할 메시지가 있는 경우에만 채널을 활성화하려면 연결 끊기 간격을 아주 낮은 값으로 설정하십시오. 기본 설정은 높은 값이므로 이 제어 레벨이 필요한 채널에는 권장되지 않습니다. 수신 채널은 인터럽트하기 어려우므로 가장 경제적인 옵션은 채널 연결이 자동으로 끊어졌다가 워크로드 요구에 따라 다시 연결되도록 하는 것입니다. 대부분 채널의 경우, 연결 끊기 간격의 적절한 설정은 경험적으로 설정할 수 있습니다.
- 하트비트 간격 속성을 사용하면 송신할 메시지가 없는 기간 동안 송신 MCA가 수신 MCA로 하트비트 플로우를 송신할 수 있습니다. 이 조치는 수신 MCA를 대기 상태에서 해제하며, 연결 끊기 간격이 만기될 때까지 기다리지 않고 채널을 일시정지할 수 있습니다. 하트비트 간격에는 연결 끊기 간격 값보다 작은 값을 지정하십시오.

  참고:

  1. 서버 채널의 경우 연결 끊기 간격을 낮은 값으로 설정하거나 하트비트를 사용하는 것이 좋습니다. 이 낮은 값은 서버 채널이 보낼 메시지가 없을 때 요청자 채널이 비정상적으로 종료되는 경우(예를 들어 채널이 취소되어)를 허용합니다. 연결 끊기 간격이 높게 설정되고 하트비트를 사용하지 않으면, 요청자가 종료된 사실을 서버가 감지하지 못합니다. 다음에 요청자에게 메시지를 보내려고 시도할 때만 감지합니다. 서버가 아직 실행 중인 경우 큐에 도달하는 메시지를 더 가져오기 위해 독점적 입력에 대해 전송 큐를 열린 상태로 유지합니다. 요청자로부터의 채널을 재시작하려고 시도하면 서버가 독점 입력을 위해 전송 큐를 계속 열린 상태로 유지하므로 시작 요청자가 오류를 수신합니다. 서버 채널을 중지한 다음 요청자로부터 다시 채널을 재시작해야 합니다.

중지된 채널 재시작

채널이 STOPPED 상태가 되면 채널을 수동으로 재시작해야 합니다.

이 태스크 정보

송신자 또는 서버 채널의 경우 채널에 STOPPED 상태를 입력하면, 연관된 전송 큐가 GET(DISABLED)로 설정되었으며 트리거 설정이 해제되었습니다. 시작 요청이 수신되면 이러한 속성이 자동으로 재설정됩니다.

z/OS 채널이 RETRYING 또는 STOPPED 상태일 때 채널 시작기가 중지되면 채널 시작기가 재시작될 때 채널 상태가 기억됩니다. 그러나 채널이 STOPPED 상태일 때 채널 시작기가 중지되면 SVRCONN 채널 유형의 채널 상태가 재설정됩니다.

Multi 채널이 RETRYING 또는 STOPPED 상태일 때 큐 관리자가 중지되면 큐 관리자가 재시작될 때 채널 상태가 기억됩니다. IBM MQ 8.0 이상에서는 SVRCONN 채널에도 적용됩니다. 이전에는 채널이 STOPPED 상태일 때 채널 시작기가 중지되면 SVRCONN 채널 유형에 대한 채널 상태가 재설정되었습니다.

프로시저

- 다음 방법 중 하나로 채널을 재시작하십시오.
  - START CHANNEL MQSC 명령 사용
  - Start Channel PCF 명령 사용
  - IBM MQ Explorer 사용
  - z/OS z/OS에서 채널 시작 패널 사용
  - IBM i IBM i에서 STRMQMCHL CL 명령 또는 WRKMQMCHL 패널에서 START 옵션 사용

인다우트*(in-doubt)* 채널

인다우트(in-doubt) 채널은 원격 채널이 송수신된 메시지를 알지 못하는 상태의 채널입니다.

이 채널과 큐에 커미트되어야 하는 메시지를 알지 못하는 큐 관리자의 차이에 유의하십시오.

배치 하트비트 채널 매개변수(BATCHHB)를 사용하여 채널이 인다우트(in-doubt) 상태가 되는 기회를 줄일 수 있습니다. 이 매개변수의 값을 지정하면, 송신자 채널이 추가 조치를 수행하기 전에 원격 채널이 계속 활성 상태인지 검사합니다. 응답이 수신되지 않으면 수신자 채널이 더 이상 활성 상태가 아닌 것으로 간주됩니다. 메시지를 롤백하고 다시 라우트할 수 있으므로 송신자 채널은 인다우트(in-doubt) 상태가 되지 않습니다. 따라서 송신자 채널이 수신자 채널이 계속 활성 상태인지 확인하고 수신자 채널이 송신된 메시지를 수신했는지 확인하는 동안

채널이 인다우트(in-doubt) 상태가 되어야 하는 시간을 줄일 수 있습니다. 배치 하트비트 매개변수에 대한 자세한 정보는 채널 속성을 참조하십시오.

인다우트(in-doubt) 채널 문제점은 일반적으로 자동으로 해결됩니다. 통신이 끊어지고 채널이 송신측에서 수신 상태를 알 수 없는 메시지 배치로 인해 채널이 인다우트(in-doubt) 상태가 되더라도 통신이 다시 설정되면 상황이 해결됩니다. 순서 번호 및 LUWID 레코드는 이런 목적을 위해 유지됩니다. 채널은 LUWID 정보가 교환될 때까지 인다우트(in-doubt) 상태이고, 하나의 메시지 배치만 채널에 대해 인다우트(in-doubt) 상태가 될 수 있습니다.

필요한 경우 채널을 수동으로 재동기화할 수 있습니다. 수동이라는 용어에는 IBM MQ 시스템 관리 명령을 포함하는 프로그램 또는 연산자 사용이 포함됩니다. 수동 재동기화 프로세스는 다음과 같이 진행됩니다. 이 설명은 MQSC 명령을 사용하지만 PCF 해당 명령도 사용할 수 있습니다.

1. DISPLAY CHSTATUS 명령을 사용하여 채널의 **각** 측에 대해 마지막으로 커미트된 논리적 작업 단위 ID(LUWID)를 찾으십시오. 이 경우 다음 명령을 사용하십시오.

   - 채널의 인다우트(in-doubt)측의 경우:

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

     CONNAME 및 XMITQ 매개변수를 사용하여 채널을 더 식별할 수 있습니다.
   - 채널의 수신측의 경우:

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

     CONNAME 매개변수를 사용하여 채널을 더 식별할 수 있습니다.

   채널의 송신측만이 인다우트(in-doubt) 상태일 수 있으므로 명령은 서로 다릅니다. 수신측은 절대로 인다우트(in-doubt) 상태가 되지 않습니다.

   IBM MQ for IBM i의 경우, STRMQMMQSC 명령 또는 MQM 채널 상태 CL에 대한 작업 명령, WRKMQMCHST를 사용하여 파일에서 DISPLAY CHSTATUS 명령이 실행될 수 있습니다.
2. 두 LUWID가 같으면 수신측이 송신자가 인다우트(in-doubt)로 간주하는 작업 단위를 커미트한 것입니다. 이제 송신측은 전송 큐에서 인다우트(in-doubt) 메시지를 제거한 후 다시 사용할 수 있습니다. 이는 다음 채널 RESOLVE 명령으로 수행됩니다.

   ```
   RESOLVE CHANNEL( name ) ACTION(COMMIT)
   ```
3. 두 개 LUWID가 다르면 수신측이 송신자가 인다우트(in-doubt) 상태로 간주하는 작업 단위를 커미트하지 않은 것입니다. 송신측은 인다우트(in-doubt) 메시지를 전송 큐에 보유했다가 재전송해야 합니다. 이는 다음 채널 RESOLVE 명령으로 수행됩니다.

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

   IBM i IBM MQ for IBM i에서 MQM 채널 해결 명령, RSVMQMCHL을 사용할 수 있습니다.

이 프로세스가 완료되면 채널이 더 이상 인다우트(in-doubt) 상태가 아닙니다. 필요한 경우 이제 다른 채널이 전송 큐를 사용할 수 있습니다.

문제점 판별

문제점 판별에는 중요한 두 가지 구분되는 측면(명령이 제출될 때 발견되는 문제점인지 채널 조작 중에 발견되는 문제점인지)이 있습니다.

명령 유효성 검증

명령 및 패널 데이터가 처리 승인을 받으려면 먼저 오류가 없어야 합니다. 유효성 검사에 의해 발견된 모든 오류는 즉시 오류 메시지에 의해 사용자에게 통보됩니다.

문제점 진단 작업은 이러한 오류 메시지를 해석하고 정정 조치를 취하는 것에서 시작합니다.

문제점 처리

채널의 일반 조작중에 발견된 문제점은 시스템 콘솔 또는 시스템 로그로 통지됩니다. 문제점 진단은 로그에서 관련된 모든 정보를 수집하는 것으로 시작하여 문제점 식별을 위한 분석을 계속합니다.

확정 및 오류 메시지는 가능한 경우 명령을 초기화한 터미널로 리턴됩니다.

IBM MQ가 회계 및 통계 데이터를 생성하면 활용 및 성능 추세를 식별하는 데 사용할 수 있습니다. Multi 멀티플랫폼에서는 이 정보가 PCF 레코드로 생성됩니다. 구조 데이터 유형을 참조하십시오. z/OS z/OS에서 이 정보는 SMF 레코드로서 생성되고, 성능 및 자원 사용 모니터링을 참조하십시오.

메시지 및 코드

문제점의 기본 진단에 도움이 되는 메시지 및 코드는 메시지 및 이유 코드의 내용을 참조하십시오.

메시지의 안전성

IBM MQ의 일반 복구 기능 이외에 분산 큐잉 관리를 통해 메시지 채널의 양측 간에 조정된 동기점 프로시저를 사용함으로써 메시지를 올바르게 전달할 수 있습니다. 이 프로시저가 오류를 감지하면, 채널이 닫혀 문제를 조사하고 채널이 다시 시작될 때까지 메시지를 전송 큐에 안전하게 유지할 수 있습니다.

동기점 프로시저에는 채널이 시작될 때 *in-doubt* 상황을 복구하려고 시도한다는 점에서 추가된 이점이 있습니다. (*in-doubt*는 동기점이 요청된 복구 단위의 상태이지만 요청 결과는 아직 알려지지 않았습니다.) 이 기능은 또한 다음 두 기능과 연관되어 있습니다.

1. 커미트 또는 백아웃으로 해석
2. 순서 번호 재설정

채널은 대부분의 경우 자동으로 복구되므로 이러한 기능은 예외적인 상황에서만 사용됩니다.

빠른, 비지속 메시지

비지속 메시지 속도(NPMSPEED) 채널 속성을 사용하여 채널의 비지속 메시지가 보다 빠르게 전달되도록 지정할 수 있습니다. 이 속성에 대한 자세한 정보는 비지속 메시지 속도(NPMSPEED)를 참조하십시오.

빠른, 비지속 메시지가 전송되는 동안 채널이 종료되면 메시지가 손실될 수 있으며 필요한 경우 애플리케이션이 복구를 조율해야 합니다.

수신 채널이 메시지를 목적지 큐에 넣지 못하는 경우, 데드 레터 큐가 정의되어 있으면 메시지가 이 큐에 위치하게 됩니다. 그렇지 않으면 메시지가 제거됩니다.

참고: 채널의 다른 측이 이 옵션을 지원하지 않는 경우에는 채널이 보통 속도로 실행됩니다.

미전달 메시지

메시지가 전달될 수 없을 때 발생하는 정보에 대해서는 167 페이지의 『메시지를 전달할 수 없을 경우의 메시지 처리』의 내용을 참조하십시오.

메시지를 전달할 수 없을 경우의 메시지 처리

메시지를 전달할 수 없는 경우 MCA가 여러 가지 방법으로 메시지를 처리할 수 있습니다. 다시 시도하거나 송신자로 돌아가거나 데드-레터 큐에 넣을 수 있습니다.

168 페이지의 그림 24에서는 MCA가 메시지를 수신지 큐에 넣을 수 없을 때 발생하는 프로세싱을 보여줍니다. 표시된 옵션이 모든 플랫폼에 적용되는 것은 아닙니다.

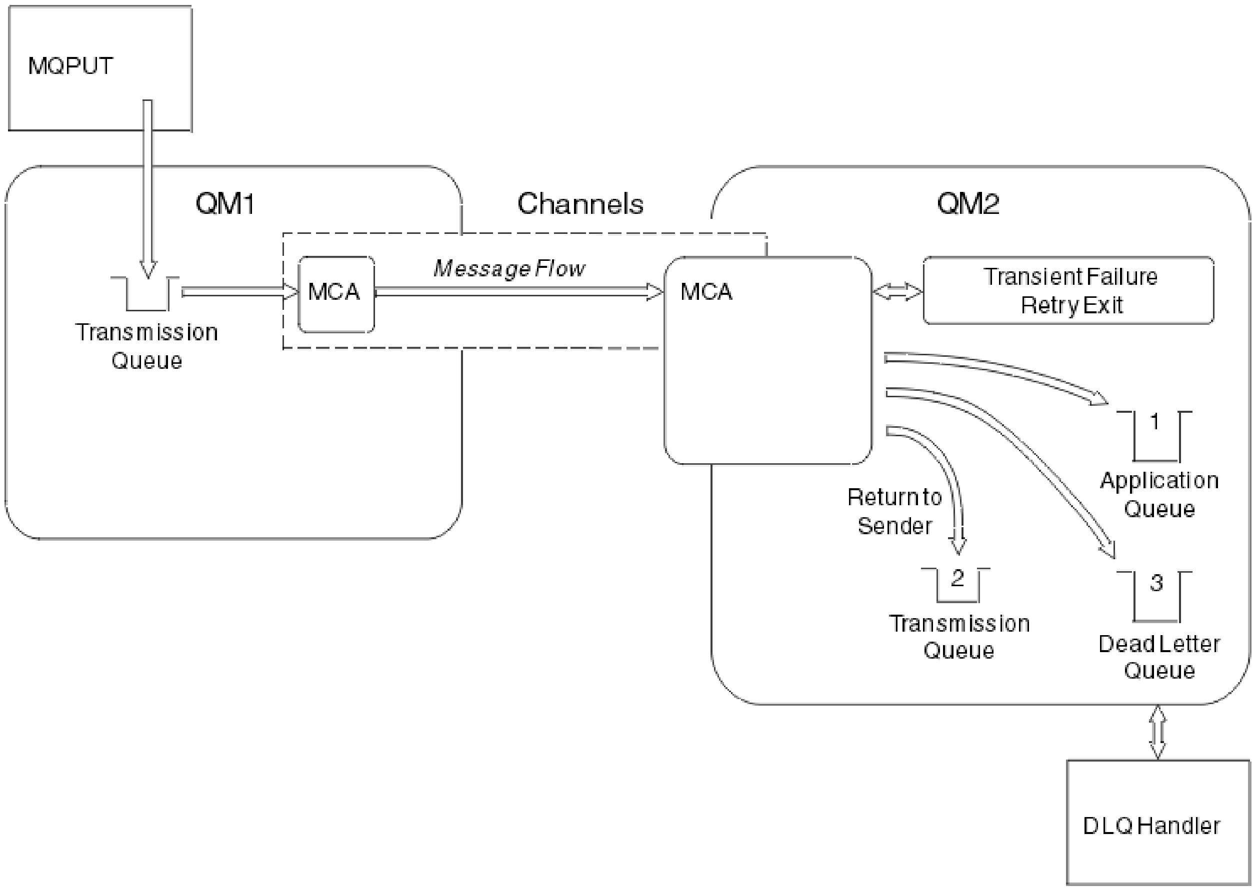


그림 *24.* 메시지를 전달할 수 없는 경우 발생하는 결과

그림과 같이, MCA는 전달할 수 없는 메시지에 대해 몇 가지 작업을 수행할 수 있습니다. 수행되는 조치는 채널이 정의될 때 지정된 옵션과 메시지에 대한 MQPUT 보고서 옵션으로 결정됩니다.

1. 메시지 재시도

   일시적인 어떤 이유로(예를 들어, 대상 큐가 가득 찬 경우) MCA가 대상 큐에 메시지를 넣을 수 없는 경우, MCA는 대기 후 나중에 조작을 다시 시도할 수 있습니다. MCA가 대기하는지 여부, 대기 시간, 시도 횟수를 판별할 수 있습니다.

   - 사용자는 채널을 정의할 때 MQPUT 오류에 대비하여 메시지 재시도 시간과 간격을 지정할 수 있습니다. 큐가 가득 찼거나 또는 넣기가 금지되어 메시지를 목적지 큐에 넣을 수 없는 경우, MCA는 지정된 시간 간격에서 지정된 횟수만큼 조작을 시도합니다.
   - 사용자 고유의 메시지 재시도 엑시트를 작성할 수 있습니다. 이 엑시트를 사용하여 MCA가 MQPUT 또는 MQOPEN 조작을 다시 시도하는 조건을 지정할 수 있습니다. 채널을 정의할 때 엑시트 이름을 지정하십시오.

2. 송신자로 돌아가기

   메시지 재시도가 성공하지 못했거나 다른 유형의 오류가 발생한 경우, MCA는 메시지를 작성자에게 다시 송신할 수 있습니다. 송신자로 돌아가기를 사용하려면 원래 큐에 메시지를 넣을 때 메시지 디스크립터에서 다음 옵션을 지정해야 합니다.

   - MQRO_EXCEPTION_WITH_FULL_DATA 보고 옵션
   - MQRO_DISCARD_MSG 보고 옵션
   - 응답 대상 큐 및 응답 대상 큐 관리자의 이름

   MCA가 메시지를 목적지 큐에 넣을 수 없을 경우, 원래 메시지를 수록한 예외 보고를 생성하고 이를 전송 큐에 넣어 원래 메시지에 지정된 응답 대상 큐로 송신되게 합니다 (응답 대상 큐가 MCA와 같은 큐 관리자에 있으면, 메시지를 전송 큐가 아닌 해당 큐에 바로 넣음).

3. 데드-레터 큐

   메시지를 전달하거나 리턴할 수 없는 경우, 이를 데드-레터 큐(DLQ)에 넣습니다. DLQ 핸들러를 사용하여 메시지를 처리할 수 있습니다. 이 처리는 데드-레터 큐의 메시지 처리(IBM MQ for UNIX, Linux, Windows 시스템의 경우) 및 데드-레터 큐 핸들러 유틸리티(CSQUDLQH)(z/OS 시스템의 경우)에 설명되어 있습니다. 데드-레터 큐를 사용할 수 없는 경우, 송신 MCA가 전송 큐에 메시지를 남겨 두고 채널은 중지됩니다. 고속 채널에서는 데드-레터 큐에 쓸 수 없는 비지속 메시지가 손실됩니다.

   IBM WebSphere MQ 7.0에서 로컬 데드-레터 큐가 정의되어 있지 않고 리모트 큐가 사용 불가능하거나 정의되어 있지 않으며 리모트 데드-레터 큐가 없으면, 송신자 채널은 재시도 상태가 되고 메시지는 전송 큐에 자동으로 롤백됩니다.

관련 정보

데드-레터 큐 사용(USEDLQ)

채널 트리거

IBM MQ에서는 큐에 대한 특정 조건이 충족될 때 자동으로 애플리케이션을 시작하기 위한 기능을 제공합니다. 이 기능을 트리거라고 합니다.

이 설명은 트리거 개념의 개요로 제공됩니다. 전체 설명은 트리거를 사용하여 IBM MQ 애플리케이션 시작을 참조하십시오.

플랫폼별 정보는 다음을 참조하십시오.

- Windows의 경우 UNIX and Linux 시스템, 170 페이지의 『UNIX, Linux, and Windows에서 채널 트리거링』의 내용을 참조하십시오.
- IBM i IBM i의 경우 171 페이지의 『IBM MQ for IBM i의 채널 트리거』의 내용을 참조하십시오.
- z/OS z/OS의 경우 659 페이지의 『전송 큐 및 트리거링 채널』의 내용을 참조하십시오.

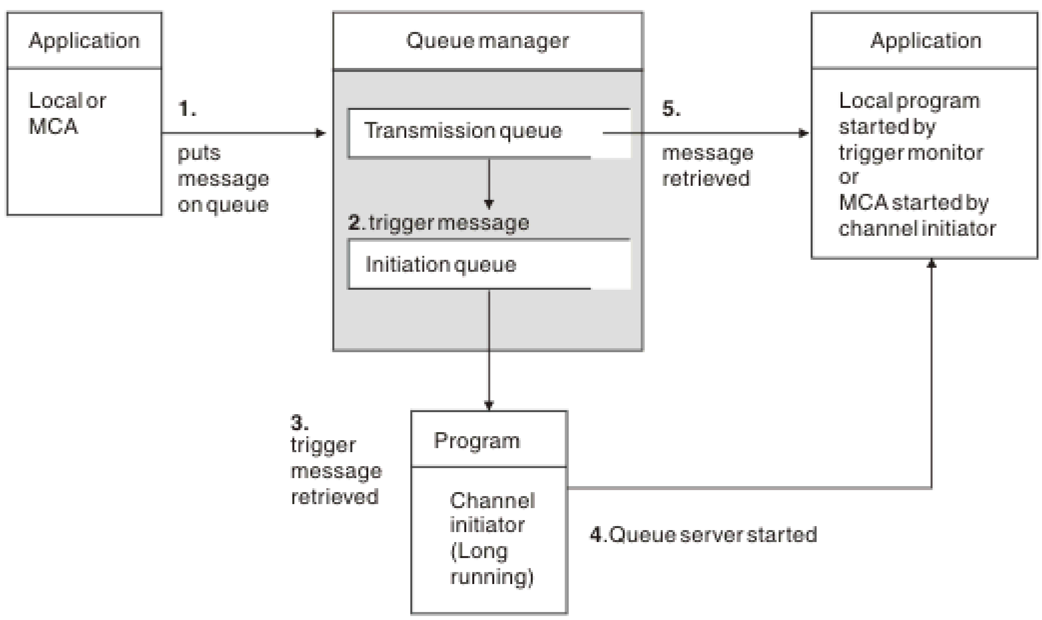


그림 *25.* 트리거의 개념

트리거에 필요한 오브젝트는 169 페이지의 그림 25에 나와 있습니다. 다음과 같은 순서의 이벤트를 보여줍니다.

1. 로컬 큐 관리자가 애플리케이션 또는 메시지 채널 에이전트(MCA)의 메시지를 전송 큐에 넣습니다.
2. 트리거 조건이 충족되면 로컬 큐 관리자는 트리거 메시지를 이니시에이션 큐에 넣습니다.
3. 장기 실행 채널 시작기 프로그램이 시작 큐를 모니터하고 도착한 메시지를 검색합니다.
4. 채널 시작기가 메시지에 포함된 정보에 따라 트리거 메시지를 처리합니다. 이 정보는 채널 이름을 포함할 수 있으며 이 경우 해당 MCA가 시작됩니다.

5. 트리거된 로컬 애플리케이션 또는 MCA가 전송 큐에서 메시지를 검색합니다.

이 시나리오를 설정하려면 다음을 수행해야 합니다.

- 해당 속성에 시작 큐(즉, SYSTEM.CHANNEL.INITQ)의 이름을 지정하여 전송 큐를 작성합니다.
- 시작 큐(SYSTEM.CHANNEL.INITQ)가 있는지 확인합니다.
- 채널 시작기 프로그램이 사용 가능하고 실행 중인지 확인합니다. 채널 시작기 프로그램에는 해당 시작 명령에 시작 큐의 이름이 제공되어야 합니다. z/OS z/OS에서는 이니시에이션 큐의 이름이 고정되므로 시작 명령에서 사용되지 않습니다.
- 선택적으로, 트리거를 위한 프로세스 정의를 작성하십시오. 존재하지 않는 경우, *UserData* 필드에 제공되는 채널의 이름이 포함되었는지 확인하십시오. 프로세스 정의를 작성하는 대신 전송 큐의 **TriggerData** 속성에 채널 이름을 지정할 수 있습니다. IBM i IBM i, UNIX, Linux, and Windows 시스템의 IBM MQ , 채널 이름을 공백으로 지정할 수 있습니다. 이 경우 이 전송 큐를 사용할 수 있는 첫 번째 채널 정의가 사용됩니다.
- 전송 큐 정의에 제공할 프로세스 정의의 이름(해당되는 경우), 이니시에이션 큐 이름, 가장 적합하다고 생각되는 트리거 특성이 포함되는지 확인합니다. 트리거 제어 속성은 필요에 따라 트리거 사용 여부를 허용합니다.

참고:

1. 채널 시작기 프로그램은 채널을 시작하는 데 사용되는 이니시에이션 큐를 모니터하는 '트리거 모니터' 역할을 합니다.
2. 이니시에이션 큐와 트리거 프로세스를 사용하여 원하는 수의 채널을 트리거할 수 있습니다.
3. 원하는 수의 이니시에이션 큐와 트리거 프로세스를 정의할 수 있습니다.
4. 시스템에 과도하게 채널이 시작되지 않게 하기 위해 FIRST 트리거 유형이 권장됩니다.

UNIX, Linux, and Windows에서 채널 트리거링

ULW

트리거될 프로세스를 정의하여 IBM MQ에 프로세스 정의를 작성할 수 있습니다. 메시지가 전송 큐에 도달할 때 트리거될 프로세스의 이름을 지정하는 프로세스 정의를 작성하려면 MQSC 명령 DEFINE PROCESS를 사용하십시오. 프로세스 정의의 USERDATA 속성은 전송 큐에서 제공되는 채널의 이름을 포함합니다.

채널(QM3.TO.QM4)을 시작하는 애플리케이션을 트리거하기 위한 트리거 메시지를 이니시에이션 큐(IQ)에 기록하도록 지정하여 로컬 큐(QM4)를 정의하십시오.

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

시작될 애플리케이션(프로세스 P1)을 정의하십시오.

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

또는 IBM MQ for UNIX, Linux, Windows 시스템의 경우 전송 큐의 TRIGDATA 속성에 채널 이름을 지정하여 프로세스 정의가 필요하지 않게 할 수 있습니다.

로컬 큐(QM4)를 정의하십시오. 트리거 메시지를 기본 이니시에이션 큐 SYSTEM.CHANNEL.INITQ에 작성하여 채널(QM3.TO.QM4)을 시작하는 애플리케이션(프로세스 P1)을 트리거하도록 지정하십시오.

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

채널 이름을 지정하지 않으면 채널 시작기가 이름 지정된 전송 큐와 관련된 채널을 찾을 때까지 채널 정의 파일을 검색합니다.

IBM MQ for IBM i의 채널 트리거

IBM i

IBM MQ for IBM i의 채널 트리거는 채널 시작기 프로세스로 구현됩니다. 큐 관리자 SCHINIT 속성을 대체하여 사용하지 않도록 설정하지 않는 한, 이니시에이션 큐 SYSTEM.CHANNEL.INITQ에 대한 채널 시작기 프로세스가 큐 관리자와 함께 자동으로 시작됩니다.

SYSTEM.CHANNEL.INITQ를 이니시에이션 큐로 지정하고 큐에 트리거를 사용하도록 설정하여 채널의 전송 큐를 설정하십시오. 채널 시작기는 이 전송 큐를 지정하는 사용 가능한 첫 번째 채널을 시작합니다.

```
CRTMQMQ QNAME(MYXMITQ1) QTYPE(*LCL) MQMNAME(MYQMGR)
TRGENBL(*YES) INITQNAME(SYSTEM.CHANNEL.INITQ)
USAGE(*TMQ)
```

STRMQMCHLI 명령으로 세 개 채널 시작기 프로세스를 수동으로 시작하고 다른 이니시에이션 큐를 지정할 수 있습니다. 전송 큐를 처리할 수 있는 채널을 여러 개 지정하고 시작할 채널을 선택할 수도 있습니다. 이 기능은 이전 릴리스와 호환 가능하도록 제공됩니다. 이 기능은 더 이상 사용되지 않습니다.

참고: 한 번에 하나의 채널만 전송 큐를 처리할 수 있습니다.

```
STRMQMCHLI QNAME(MYINITQ)
```

TRGENBL(*YES)를 지정하여 채널의 전송 큐를 설정하고, 시작하려는 채널을 선택하려면 TRIGDATA 필드에 채널 이름을 지정하십시오. 예를 들면, 다음과 같습니다.

```
CRTMQMQ QNAME(MYXMITQ2) QTYPE(*LCL) MQMNAME(MYQMGR)
TRGENBL(*YES) INITQNAME(MYINITQ)
USAGE(*TMQ) TRIGDATA(MYCHANNEL)
```

관련 개념

171 페이지의 『채널 시작기 시작 및 중지』
트리거는 채널 시작기 프로세스를 사용하여 구현됩니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

관련 정보

UNIX, Linux, and Windows에서 채널 프로그램

IBM i IBM i의 상호통신 작업

IBM i IBM i의 채널 상태

채널 시작기 시작 및 중지

트리거는 채널 시작기 프로세스를 사용하여 구현됩니다.

이 채널 이니시에이터 프로세스는 MQSC 명령 START CHINIT로 시작됩니다. 기본 이니시에이션 큐를 사용하지 않는 경우 명령에서 이니시에이션 큐의 이름을 지정하십시오. 예를 들어, 기본 큐 관리자에 대해 큐 IQ를 시작하기 위해 START CHINIT 명령을 사용하려면 다음을 입력하십시오.

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

기본적으로 기본 이니시에이션 큐, SYSTEM.CHANNEL.INITQ를 사용하여 자동으로 채널 이니시에이터가 시작됩니다. 모든 채널 시작기를 수동으로 시작하려면 다음 단계를 수행하십시오.

1. 큐 관리자를 작성한 후 시작하십시오.
2. 큐 관리자의 SCHINIT 등록 정보를 MANUAL로 변경하십시오.
3. 큐 관리자를 종료한 후 재시작하십시오.

IBM MQ for Multiplatforms 시스템에서 채널 시작기가 자동으로 시작됩니다. 시작할 수 있는 채널 시작기 수는 제한적입니다. 기본값 및 최대값은 3입니다. UNIX and Linux 시스템의 qm.ini 파일 및 Windows 시스템의 레지스트리에서 MAXINITIATORS를 사용하여 이를 변경할 수 있습니다.

채널 시작기 실행 명령 **runmqchi** 및 기타 제어 명령에 대한 자세한 내용은 IBM MQ 제어 명령을 참조하십시오.

채널 시작기 중지

큐 관리자를 시작하면 기본 채널 시작기가 자동으로 시작됩니다. 큐 관리자가 중지되면 모든 채널 시작기가 자동으로 중지됩니다.

초기화 및 구성 파일

채널 초기화 데이터 처리는 IBM MQ 플랫폼에 따라 다릅니다.

z/OS 시스템

z/OS

IBM MQ for z/OS의 경우, 초기화 및 구성 정보는 ALTER QMGR MQSC 명령을 사용하여 지정됩니다. CSQINP2 초기화 입력 데이터 세트에 ALTER QMGR 명령을 넣으면 큐 관리자가 시작될 때마다 이 명령이 처리됩니다.

채널 시작기를 시작할 때마다 MQSC 명령(예: START LISTENER)을 실행하려면 CSQINPX 초기화 입력 데이터 세트에 원하는 명령을 넣고 작업 절차를 시작한 채널 시작기에 선택적 DD문 CSQINPX를 지정하십시오.

CSQINP2 및 CSQINPX에 대한 추가 정보는 초기화 입력 데이터 세트 사용자 정의 및 ALTER QMGR을 참조하십시오.

Windows, IBM i IBM i, UNIX and Linux 시스템

IBM MQ for Windows, IBM i IBM i, UNIX and Linux 시스템에는 IBM MQ 설치에 대한 기본 구성 정보를 보유하는 구성 파일 이 있습니다.

두 개의 구성 파일이 있는데, 하나는 시스템에 적용되고 다른 하나는 개별 큐 관리자에 적용됩니다.

IBM MQ 구성 파일

이 파일에는 IBM MQ 시스템의 모든 큐 관리자와 관련된 정보가 저장됩니다. 이 파일을 `mqs.ini`라고 합니다. 이 파일에 대한 설명은 관리 for IBM MQ for Windows, , IBM i 관리, UNIX and Linux 시스템을 참조하십시오.

큐 관리자 구성 파일

이 파일에는 특정 단일 큐 관리자와 관련된 구성 정보가 포함됩니다. 이 파일의 이름은 `qm.ini`입니다.

이 파일은 큐 관리자 작성 중에 작성되며 큐 관리자의 모든 측면과 관련된 구성 정보를 저장할 수 있습니다. 파일에 저장되는 정보에는 로그 구성과 IBM MQ 구성 파일의 기본값 차이에 대한 자세한 정보가 포함됩니다.

큐 관리자 구성 파일은 큐 관리자가 있는 디렉토리 트리의 루트에 저장됩니다. 예를 들어, DefaultPath 속성의 경우 큐 관리자 QMNAME의 큐 관리자 구성 파일은 다음과 같습니다.

UNIX and Linux 시스템의 경우

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

`qm.ini` 파일의 일부는 다음과 같습니다. 여기서는 TCP/IP 리스너가 포트 2500에서 대기하고 최대 현재 채널 수가 200이며 최대 활성 채널 수는 100인 것으로 지정됩니다.

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

아웃바운드 채널이 사용할 TCP/IP 포트 범위를 지정할 수 있습니다. 한 가지 방법은 qm.ini 파일을 사용하여 포트 값 범위의 시작과 끝을 지정하는 것입니다. 다음 예는 채널의 범위를 지정하는 qm.ini 파일을 표시합니다.

```
TCP:
StrPort=2500
EndPort=3000
CHANNELS:
MaxChannels=200
MaxActiveChannels=100
```

StrPort 또는 EndPort의 값을 지정할 경우, 둘 다에 대해 값을 지정해야 합니다. EndPort의 값은 항상 StrPort의 값보다 커야 합니다.

채널은 지정된 범위의 각 포트 값을 사용하려고 합니다. 성공적으로 연결되면, 포트 값은 채널이 사용할 수 있는 포트가 됩니다.

IBM i IBM i의 경우:

```
/QIBM/UserData/mqm/qmgrs/QMNAME/qm.ini
```

Windows 시스템의 경우

```
C:\ProgramData\IBM\MQ\qmgrs\QMNAME\qm.ini
```

`qm.ini` 파일에 대한 자세한 정보는 분산 큐잉의 구성 파일 스탠자의 내용을 참조하십시오.

데이터 변환

IBM MQ 메시지를 다른 큐 관리자의 큐 사이에 보내려면 데이터 변환이 필요할 수 있습니다.

IBM MQ 메시지는 다음 두 파트로 구성됩니다.

- 메시지 디스크립터 내의 제어 정보
- 애플리케이션 데이터

다른 큐 관리자의 큐 사이에 보내려면 두 파트 중 하나에 데이터 변환이 필요할 수 있습니다. 애플리케이션 데이터 변환에 대한 정보는 애플리케이션 데이터 변환을 참조하십시오.

사용자 고유의 메시지 채널 에이전트 작성

IBM MQ를 사용하여 사용자 고유의 메시지 채널 에이전트(MCA) 프로그램을 작성하거나 독립 소프트웨어 벤더의 프로그램을 설치할 수 있습니다.

IBM MQ가 자체 독점 통신 프로토콜을 통해 상호운용되거나 IBM MQ가 지원하지 않는 프로토콜을 통해 메시지를 보내는 사용자 고유의 MCA 프로그램을 작성할 수 있습니다. IBM MQ 제공 MCA와 상호운용되는 사용자 고유의 MCA는 다른 측에 작성할 수 없습니다.

IBM MQ가 제공하지 않은 MCA를 사용하려는 경우 다음 사항을 고려해야 합니다.

메시지 송신 및 수신
: 애플리케이션이 메시지를 넣는 어느 위치에서나(예를 들어, 전송 큐에서) 메시지를 가져오고 통신하려는 프로토콜에서 전송하는 송신 애플리케이션을 작성해야 합니다. 이 프로토콜에서 메시지를 가져와 목적지 큐에 넣는 수신 애플리케이션도 작성해야 합니다. 송신 및 수신 애플리케이션은 특별한 인터페이스가 아닌 메시지 큐 인터페이스(MQI) 호출을 사용합니다.

  메시지는 한 번만 전달되어야 합니다. 동기점 조정을 사용하여 이 전달을 도울 수 있습니다.

채널 제어 기능
: 채널을 제어하는 고유 관리 기능을 제공해야 합니다. 채널 구성(예를 들어, DEFINE CHANNEL 명령) 또는 모니터링(예를 들어, DISPLAY CHSTATUS 명령)을 위해 IBM MQ 채널 관리 기능을 사용할 수 없습니다.

초기화 파일
: 필요한 경우 고유 초기화 파일을 제공해야 합니다.

애플리케이션 데이터 변환
: 다른 시스템으로 보내는 메시지의 데이터 변환을 허용할 수 있습니다. 이러한 경우 애플리케이션이 메시지를 넣는 위치에 관계없이(예를 들어, 전송 큐) 메시지를 검색할 때 MQGET 호출에서 MQGMO_CONVERT 옵션을 사용하십시오.

사용자 엑시트
: 사용자 엑시트가 필요한지 여부를 고려하십시오. 필요한 경우 IBM MQ가 사용하는 동일한 인터페이스 정의를 사용할 수 있습니다.

트리거링
: 애플리케이션이 메시지를 전송 큐에 넣으면 메시지가 큐에 도달할 때 송신 MCA가 트리거되도록 전송 큐 속성을 설정할 수 있습니다.

채널 시작기
: 사용자 고유의 채널 시작기를 제공해야 할 수도 있습니다.

분산 큐 관리를 위해 고려해야 할 기타 사항

분산 큐 관리를 위한 IBM MQ 준비 시 고려해야 할 기타 주제. 이 주제에서는 미전달 메시지 큐, 사용 중인 큐, 시스템 확장 기능, 사용자 엑시트 프로그램과 실행 중인 채널 및 리스너를 신뢰성 있는 애플리케이션으로 다룹니다.

미전달 메시지 큐

미전달 메시지 큐에 도착하는 메시지(데드-레터 큐 또는 DLQ라고도 함)가 처리되었는지 확인하려면, 이러한 메시지를 처리하기 위해 정기적으로 실행 또는 트리거될 수 있는 프로그램을 작성하십시오.

Linux UNIX DLQ 핸들러는 UNIX and Linux 시스템에서 IBM MQ 를 사용하여 제공됩니다. 자세한 정보는 샘플 DLQ 핸들러, amqsdlq를 참조하십시오.

IBM i IBM MQ for IBM i에 대한 자세한 정보는 IBM MQ for IBM i 데드-레터 큐 핸들러를 참조하십시오.

사용 중인 큐

메시지가 전송 중이 아니더라도 수신자 채널용 MCA는 목적지 큐를 계속 열어둘 수 있습니다. 이 경우 큐가 "사용 중"으로 표시됩니다.

최대 채널 수

IBM i IBM MQ for IBM i에서 시스템에 허용되는 최대 채널 수와 한 번에 활성화될 수 있는 최대 수를 지정할 수 있습니다. 이 숫자는 QIM/UserData/mqm/qmgrs/*queue_manager_name*디렉토리의 `qm.ini` 파일에 지정합니다. 분산 큐잉을 위한 파일 스탠자 구성을 참조하십시오.

시스템 확장 및 사용자 엑시트 프로그램

메시지 처리 중 정의된 시간에 기타 프로그램을 실행할 수 있는 기능이 채널 정의에 제공됩니다. 이러한 프로그램은 IBM MQ와 함께 제공되지 않지만 로컬 요구사항에 따라 각 설치에서 제공됩니다.

이러한 사용자 엑시트 프로그램을 실행하려면 사전정의된 이름이 있어야 하며 채널 프로그램에 대한 호출에서 사용 가능해야 합니다. 사용자 엑시트 프로그램의 이름은 메시지 채널 정의에 포함되어 있습니다.

이 프로그램에 대한 제어 핸들링 및 이 프로그램으로부터의 제어 리턴 핸들링을 위해 정의된 제어 블록 인터페이스가 있습니다.

이러한 프로그램이 호출되는 정확한 위치와 제어 블록 및 이름에 대한 자세한 내용은 메시징 채널에 대한 채널 엑시트 프로그램을 참조하십시오.

채널 및 리스너를 신뢰 애플리케이션으로 실행

환경에서 성능이 중요한 고려사항이고 환경이 안정적인 경우, FASTPATH 바인딩을 사용하여 채널과 리스너를 신뢰할 수 있는 것으로 실행할 수 있습니다. 채널 및 리스너를 신뢰 상태로 실행할지 여부에 영향을 주는 요인은 두 가지가 있습니다.

- 환경 변수 MQ_CONNECT_TYPE=FASTPATH 또는 MQ_CONNECT_TYPE=STANDARD. 대소문자를 구분합니다. 올바르지 않은 값을 지정하면 무시됩니다.
- `qm.ini` 또는 레지스트리 파일의 Channels 스탠자의 MQIBindType. 이를 FASTPATH 또는 STANDARD로 설정하고 대소문자를 구분하지 않습니다. 기본값은 STANDARD입니다.

MQIBindType을 환경 변수와 함께 사용하면 다음과 같이 필요한 결과를 얻을 수 있습니다.

MQIBindType	환경 변수	결과
STANDARD	UNDEFINED	STANDARD
FASTPATH	UNDEFINED	FASTPATH
STANDARD	STANDARD	STANDARD
FASTPATH	STANDARD	STANDARD
STANDARD	FASTPATH	STANDARD
FASTPATH	FASTPATH	FASTPATH
STANDARD	클라이언트	클라이언트
FASTPATH	클라이언트	STANDARD
STANDARD	LOCAL	STANDARD
FASTPATH	LOCAL	STANDARD

즉, 채널과 리스너를 실제로 신뢰 상태로 실행하는 방법은 다음과 같은 두 가지가 있습니다.

1. `qm.ini` 또는 레지스트리에 MQIBindType=FASTPATH를 지정하고 환경 변수는 지정하지 않습니다.
2. `qm.ini` 또는 레지스트리에 MQIBindType=FASTPATH를 지정하고 환경 변수를 FASTPATH로 설정합니다.

리스너는 안정적인 프로세스이므로 리스너를 신뢰 상태로 실행할 수 있습니다. 불안정한 채널 엑시트나 명령 STOP CHANNEL MODE(TERMINATE)를 사용하지 않는 한 채널을 신뢰 상태로 실행할 수 있습니다.

ULW UNIX, Linux, and Windows에서 채널 모니터링 및 제어

DQM의 경우 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어해야 합니다. 명령, 프로그램, IBM MQ Explorer, 채널 정의 파일, 동기화 정보 스토리지 영역을 사용하여 채널을 제어할 수 있습니다.

이 태스크 정보

채널을 제어하는 데 다음 유형의 명령을 사용할 수 있습니다.

IBM MQ 명령(MQSC)

UNIX, Linux, and Windows 시스템에서는 MQSC를 MQSC 세션에서 단일 명령으로 사용할 수 있습니다. 보다 복잡한 명령이나 복수 명령을 실행하려면 MQSC를 파일에 빌드하여 명령행에서 해당 파일을 실행할 수 있습니다. 자세한 내용은 MQSC 명령을 참조하십시오. 이 절에서는 분산 큐잉을 위해 MQSC를 사용하는 간단한 몇 가지 예를 제공합니다.

채널 명령은 IBM MQ 명령(MQSC)의 서브세트입니다. MQSC 및 제어 명령을 사용하여 다음을 수행할 수 있습니다.

- 채널 정의 작성, 복사, 표시, 변경, 삭제
- 채널 시작 및 중지, ping, 채널 순서 번호 재설정, 링크를 재설정할 수 없는 경우 인다우트(in-doubt) 메시지 해결
- 채널에 대한 상태 정보 표시

제어 명령

일부 기능의 경우 명령행으로 제어 명령을 실행할 수도 있습니다. 자세한 내용은 IBM MQ 제어 명령 참조를 참조하십시오.

프로그래밍 가능 명령 형식 명령
자세한 내용은 PCF 명령을 참조하십시오.

Windows Linux **IBM MQ Explorer**
Linux 및 Windows 시스템에서는 IBM MQ Explorer를 사용할 수 있습니다. 제어 명령이나 MQSC 명령을 사용하지 않고 관리 태스크를 수행하기 위해 그래픽 관리 인터페이스를 제공합니다. 채널 정의는 큐 관리자 오브젝트로 보유됩니다.

각 큐 관리자에는 호환 가능한 리모트 큐 관리자에 대한 상호 연결을 제어하기 위한 DQM 컴포넌트가 있습니다. 스토리지 영역에는 순서 번호와 논리적 작업 단위*(LUW)* ID가 저장됩니다. 이러한 영역은 채널 동기화 목적으로 사용됩니다.

서로 다른 유형의 명령을 사용하여 메시지 채널을 설정하고 제어할 때 사용하는 기능들의 목록에 대해서는 177 페이지의 표 19의 내용을 참조하십시오.

프로시저

- 176 페이지의 『채널 설정 및 제어에 필요한 기능』
- 178 페이지의 『오브젝트 시작하기』
- 184 페이지의 『Windows에서 통신 설정』
- 191 페이지의 『UNIX and Linux에서 통신 설정』

관련 태스크
196 페이지의 『IBM i에서 채널 모니터링 및 제어』
DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다. 각 큐 관리자에는 호환 가능한 리모트 큐 관리자에 대한 상호 연결을 제어하기 위한 DQM 프로그램이 있습니다.

관련 정보
ULW UNIX, Linux, and Windows에서 채널 프로그램
ULW UNIX, Linux, and Windows의 메시지 채널 계획 예
구성 정보 예
채널 속성

ULW 채널 설정 및 제어에 필요한 기능

채널을 설정하고 제어하려면 여러 가지 IBM MQ 기능이 필요합니다. 이 주제에서는 채널 기능을 설명합니다.

IBM MQ가 제공하는 기본값으로 채널 이름, 작성하는 채널 유형, 사용할 통신 방법, 전송 큐 이름과 연결 이름을 지정하여 채널 정의를 작성할 수 있습니다.

채널 이름은 채널 양쪽에서 동일하고 네트워크 내에서 고유해야 합니다. 그러나 사용되는 문자를 IBM MQ 오브젝트 이름에 유효한 문자로 제한해야 합니다.

다른 채널 관련 기능은 다음 주제를 참조하십시오.

- 178 페이지의 『오브젝트 시작하기』
- 179 페이지의 『연관된 오브젝트 작성』
- 179 페이지의 『기본 오브젝트 작성』
- 179 페이지의 『채널 작성』
- 180 페이지의 『채널 표시』
- 180 페이지의 『채널 상태 표시』
- 181 페이지의 『Ping을 사용하여 링크 검사』
- 181 페이지의 『채널 시작』
- 182 페이지의 『채널 중지』
- 183 페이지의 『채널 이름 바꾸기』
- 183 페이지의 『채널 재설정』

- 184 페이지의 『채널상의 인다우트(in-doubt) 메시지 해결』

177 페이지의 표 19은 필요한 IBM MQ 기능의 전체 목록을 보여줍니다.

표 *19. UNIX, Linux, and Windows* 시스템에 필요한 기능

Function	**제어 명령**	**MQSC**	**IBM MQ 탐색기 해당 여부**
큐 관리자 기능			
큐 관리자 변경		ALTER QMGR	예
큐 관리자 작성	crtmqm		예
큐 관리자 삭제	dltmqm		예
큐 관리자 표시		DISPLAY QMGR	예
큐 관리자 종료	endmqm		예
큐 관리자 ping		PING QMGR	아니오
큐 관리자 시작	strmqm		예
명령 서버 기능			
명령 서버 표시	dspmqcsv		아니오
명령 서버 종료	endmqcsv		아니오
명령 서버 시작	strmqcsv		아니오
큐 기능			
큐 변경		ALTER QALIAS ALTER QLOCAL ALTER QMODEL ALTER QREMOTE ALTER 큐를 참조하십시오.	예
큐 지우기		CLEAR QLOCAL	예
큐 작성		DEFINE QALIAS DEFINE QLOCAL DEFINE QMODEL DEFINE QREMOTE DEFINE 큐를 참조하십시오.	예
큐 삭제		DELETE QALIAS DELETE QLOCAL DELETE QMODEL DELETE QREMOTE DELETE 큐를 참조하십시오.	예
큐 표시		DISPLAY QUEUE	예
프로세스 기능			
프로세스 변경		ALTER PROCESS	예
프로세스 작성		DEFINE PROCESS	예

표 19. UNIX, Linux, and Windows 시스템에 필요한 기능 (계속)			
Function	**제어 명령**	**MQSC**	**IBM MQ 탐색기 해당 여부**
프로세스 삭제		프로세스 삭제	예
프로세스 표시		DISPLAY PROCESS	예
채널 기능			
채널 변경		ALTER CHANNEL	예
채널 작성		DEFINE CHANNEL	예
채널 삭제		채널 삭제	예
채널 표시		DISPLAY CHANNEL	예
채널 상태 표시		DISPLAY CHSTATUS	예
채널 종료		STOP CHANNEL	예
채널 ping		PING CHANNEL	예
채널 재설정		RESET CHANNEL	예
채널 해결		RESOLVE CHANNEL	예
채널 실행	runmqchl	START CHANNEL	예
채널 시작기 실행	runmqchi	START CHINIT	아니오
리스너 실행[1]	runmqlsr	START LISTENER	아니오
리스너 종료	endmqlsr(Windows 시스템, AIX, HP-UX, Solaris만 해당)		아니오

참고:

1. 큐 관리자가 시작될 때 리스너가 자동으로 시작될 수 있습니다.

ULW 오브젝트 시작하기

채널을 시작하려면 채널이 정의되고 연관된 오브젝트가 존재해야 하며 해당 채널을 사용할 수 있어야 합니다. 이 절은 그 방법을 보여줍니다.

IBM MQ 명령(MQSC) 또는 IBM MQ Explorer를 사용하여 다음을 수행할 수 있습니다.

1. 메시지 채널 및 연관된 오브젝트 정의
2. 메시지 채널 모니터 및 제어

정의해야 하는 연관된 오브젝트는 다음과 같습니다.

- 전송 큐
- 리모트 큐 정의
- 큐 관리자 알리어스 정의
- 응답 대상 큐 알리어스 정의
- 응답 대상 로컬 큐
- 트리거용 프로세스(MCA)
- 메시지 채널 정의

채널을 실행하려면 각 채널에 대한 특정 통신 링크를 정의하고 사용 가능해야 합니다. LU 6.2, TCP/IP, NetBIOS, SPX, DECnet 링크 정의 방법에 대한 설명은 사용자 설치에 맞는 특정 통신 안내서를 참조하십시오. 구성 정보 예도 참조하십시오.

오브젝트 작성 및 관련 작업에 대한 자세한 정보는 다음 하위 주제를 참조하십시오.

ULW 연관된 오브젝트 작성

MQSC를 사용하여 연관된 오브젝트를 작성할 수 있습니다.

MQSC를 사용하여 큐 및 알리어스 오브젝트(전송 큐, 리모트 큐 정의, 큐 관리자 알리어스 정의, 응답 대상 큐 알리어스 정의, 응답 대상 로컬 큐)를 작성하십시오.

또한 유사한 방법으로 트리거(MCA) 프로세스 정의를 작성하십시오.

모든 필수 오브젝트 작성 방법을 보여주는 예는 UNIX, Linux, and Windows용 메시지 채널 계획 예를 참조하십시오.

ULW 기본 오브젝트 작성

큐 관리자가 작성될 때 기본 오브젝트가 자동으로 작성됩니다. 해당 오브젝트는 큐, 채널, 프로세스 정의, 관리 큐입니다. 기본 오브젝트가 작성된 후에는 언제라도 -c 옵션과 strmqm 명령을 실행하여 오브젝트를 바꿀 수 있습니다.

crtmqm 명령을 사용하여 큐 관리자를 작성하는 경우 이 명령은 또한 기본 오브젝트 세트를 작성하는 프로그램을 시작합니다.

1. 각 기본 오브젝트가 차례로 작성됩니다. 프로그램은 성공적으로 정의되는 오브젝트 수, 바뀐 기존 오브젝트 수와 시도 실패 횟수를 보존합니다.
2. 프로그램은 그 결과를 사용자에게 표시하며 오류가 발생하면 세부사항을 나타내는 적절한 오류 로그를 제시합니다.

프로그램 실행이 완료되면 strmqm 명령을 사용하여 큐 관리자를 시작할 수 있습니다.

crtmqm 및 strmqm 명령에 대한 자세한 정보는 IBM MQ 제어 명령 참조를 참조하십시오.

기본 오브젝트 변경

-c 옵션을 지정하는 경우 큐 관리자는 오브젝트가 작성되는 동안 임시로 시작되고 다시 종료됩니다. -c 옵션을 사용하여 strmqm을 실행하면 기존 시스템 오브젝트가 기본값으로 새로 고쳐집니다(예를 들어, 채널 정의의 MCAUSER 속성이 공백으로 설정됨). 큐 관리자를 시작하려면 -c 옵션 없이 strmqm 명령을 다시 사용해야 합니다.

기본 오브젝트를 변경하려는 경우 이전 amqscoma.tst 파일의 사용자 고유 버전을 작성하여 편집하면 됩니다.

ULW 채널 작성

연결의 각 측에 하나씩 두 개의 채널 정의를 작성할 수 있습니다. 첫 번째 큐 관리자에서 첫 번째 채널 정의를 작성합니다. 그런 다음 링크의 다른 측에 있는 두 번째 큐 관리자에서 두 번째 채널 정의를 작성합니다.

양 측을 모두 동일한 채널 이름을 사용해서 정의해야 합니다. 양 측에서는 호환 가능한 채널 유형(예: 송신자 및 수신자)을 사용해야 합니다.

링크의 한쪽 끝에 대해 채널 정의를 작성하려면 MQSC 명령 DEFINE CHANNEL을 사용하십시오. 채널 이름, 연결의 이 측에 대한 채널 유형, 연결 이름, 설명(필요한 경우), 전송 큐의 이름(필요한 경우), 전송 프로토콜을 포함하십시오. 또한 이전에 수집한 정보를 사용하여 필요한 채널 유형에 대해 시스템 기본값과 다르게 설정하려는 모든 속성을 포함하십시오.

채널 속성 값을 결정하기 위한 도움말이 채널 속성에 제공됩니다.

참고: 네트워크에서 모든 채널의 이름을 고유하게 지정하는 것이 좋습니다. 채널 이름에 소스 및 대상 큐 관리자 이름을 포함하는 것이 효과적인 방법입니다.

채널 작성 예

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

모든 MQSC 예에서는 명령이 명령 파일에 나타난 대로 표시되고 UNIX, Linux, and Windows에 입력된 대로 표시됩니다. 명령을 대화식으로 실행하기 위해서는 먼저 MQSC 세션을 시작해야 한다는 것을 제외하고는 두 가지 방법이 동일하게 보입니다. 기본 대기열 관리자의 경우 `runmqsc`또는 `runmqsc` *qmname* 을 입력하십시오. 여기서 *qmname* 은 필수 대기열 관리자의 이름입니다. 그런 다음, 예에 나타난 것처럼 명령을 입력하십시오.

이식성을 위해 명령의 행 길이는 72자로 제한하십시오. 위에 표시된 것처럼 연결 문자 +를 사용하면 여러 행에서 계속될 수 있습니다.

- Windows Windows의 경우 명령행에서 항목을 종료하려면 Ctrl-z를 사용하십시오.
- Linux UNIX UNIX and Linux의 경우 Ctrl-d를 사용하십시오.
- 또는, UNIX, Linux, and Windows에서 **end** 명령을 사용하십시오.

ULW 채널 표시

MQSC 명령 DISPLAY CHANNEL을 사용하여 채널 속성을 표시할 수 있습니다.

특정 속성이 필요하지 않고 지정된 채널 이름이 일반적이지 않은 경우 기본적으로 DISPLAY CHANNEL 명령의 ALL 매개변수를 가정합니다.

속성은 채널 속성에 설명되어 있습니다.

채널 표시 예

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

DISPLAY CHANNEL(QM1.TO.*) TRPTYPE,CONVERT

DISPLAY CHANNEL(*) TRPTYPE,CONVERT

DISPLAY CHANNEL(QM1.TO.QMR34) ALL
```

ULW 채널 상태 표시

채널 이름과, 현재 채널 상태 또는 저장된 정보 상태가 필요한지 여부를 지정하여 MQSC 명령 DISPLAY CHSTATUS를 사용합니다.

DISPLAY CHSTATUS는 모든 메시지 채널에 적용됩니다. 서버-연결 채널 이외의 MQI 채널에는 적용되지 않습니다.

표시되는 정보는 다음과 같습니다.

- 채널 이름
- 통신 연결 이름
- 채널의 인다우트(In-doubt) 상태(해당되는 경우)
- 마지막 순서 번호
- 전송 큐 이름(해당되는 경우)
- 인다우트(in-doubt) ID(해당되는 경우)
- 마지막으로 커미트된 순서 번호
- 논리적 작업 단위 ID
- 프로세스 ID
- Windows 스레드 ID(Windows만 해당)

채널 상태 표시 예

```
DISPLAY CHSTATUS(*) CURRENT

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

저장된 상태는 최소 하나의 메시지 배치가 채널에서 전송될 때까지 적용되지 않습니다. 채널이 중지되고(STOP CHL 명령 사용) 큐 관리자가 종료될 때에도 상태가 저장됩니다.

ULW *Ping*을 사용하여 링크 검사

MQSC 명령 PING CHANNEL을 사용하여 고정 데이터 메시지를 원격 끝과 교환할 수 있습니다.

ping은 시스템 감독자에게 링크를 사용할 수 있고 작동한다는 확신을 줄 수 있습니다.

Ping은 전송 큐와 대상 큐를 사용하지 않습니다. 채널 정의, 관련 통신 링크, 네트워크 설정을 사용합니다. 채널이 현재 활성 상태가 아닌 경우에만 사용할 수 있습니다.

송신자, 서버, 클러스터-송신자 채널에서만 사용할 수 있습니다. 해당 채널은 링크의 먼 측에서 시작되며 시동 매개변수 조정을 수행합니다. 오류는 정상적으로 보고됩니다.

메시지 교환의 결과는 `Ping complete` 또는 오류 메시지로 표시됩니다.

LU 6.2에서의 Ping

ping이 호출될 때 기본적으로 사용자 ID 또는 비밀번호가 수신측으로 이동하지 않습니다. 사용자 ID와 비밀번호가 필요한 경우에는 채널 정의의 시작하는 한쪽 끝에서 작성될 수 있습니다. 채널 정의에 비밀번호가 입력되면 저장되기 전에 IBM MQ로 암호화됩니다. 그런 다음 대화에서 이동하기 전에 암호화를 해제합니다.

ULW 채널 시작

MQSC 명령 START CHANNEL은 송신자, 서버, 요청자 채널에 사용합니다. 애플리케이션이 메시지를 교환하려면 인바운드 연결을 위해 리스너 프로그램을 시작해야 합니다.

큐 관리자 트리거로 채널이 설정된 경우에는 START CHANNEL이 필요하지 않습니다.

시작 시 송신 MCA가 채널 정의를 읽고 전송 큐를 엽니다. 채널 시작 순서가 발행되어 수신자 또는 송신자 채널의 해당 MCA가 원격으로 시작됩니다. 시작된 후에는 송신자 및 서버 프로세스가 전송 큐에 도착하는 메시지를 대기하고 도착 시 메시지를 전송합니다.

트리거를 사용하거나 채널을 스레드로 실행하는 경우, 시작 큐를 모니터하기 위해 채널 시작기를 사용할 수 있는지 확인하십시오. 기본적으로 채널 시작기는 큐 관리자의 일부분으로 시작됩니다.

그러나 TCP와 LU 6.2는 다른 기능을 제공합니다.

- Linux UNIX UNIX and Linux의 TCP의 경우, 채널을 시작하도록 inetd를 구성할 수 있습니다. inetd는 별도 프로세스로 시작됩니다.
- Linux UNIX UNIX and Linux의 LU 6.2의 경우, LU 6.2 응답자 프로세스를 시작하도록 SNA 제품을 구성하십시오.
- Windows Windows의 LU 6.2의 경우, SNA 서버를 사용하면 TpStart(SNA 서버와 함께 제공되는 유틸리티)로 채널을 시작할 수 있습니다. TpStart는 별도 프로세스로 시작됩니다.

필요한 경우 시작 옵션을 사용하면 항상 채널이 재동기화됩니다.

정상적으로 시작하려면,

- 로컬 및 원격 채널 정의가 존재해야 합니다. 수신자 또는 서버 연결 채널에 대해 적합한 채널 정의가 없으면, 채널이 자동 정의될 때 기본 정의가 자동으로 작성됩니다. 채널 자동 정의 엑시트 프로그램을 참조하십시오.
- 전송 큐가 존재해야 하며 해당 큐를 사용하는 다른 채널이 없어야 합니다.
- 로컬 및 원격 MCA가 존재해야 합니다.
- 통신 링크가 사용 가능해야 합니다.
- 로컬 및 리모트 큐 관리자가 실행 중이어야 합니다.

• 메시지 채널이 이미 실행되고 있으면 안됩니다.

채널 시작 요청이 승인되었음을 확인하는 메시지가 화면에 리턴됩니다. 시작 명령이 성공했는지 확인하려면 오류 로그를 검사하거나 DISPLAY CHSTATUS를 사용하십시오. 오류 로그는 다음과 같습니다.

Windows **Windows**

MQ_DATA_PATH\qmgrs\qmname\errors\AMQERR01.LOG (qmname라는 각 큐 관리자에 대해)

MQ_DATA_PATH\qmgrs\@SYSTEM\errors\AMQERR01.LOG (일반 오류의 경우)

MQ_DATA_PATH 는 IBM MQ 가 설치된 상위 레벨 디렉토리를 나타냅니다.

참고: Windows의 경우, Windows 시스템 애플리케이션 이벤트 로그에서 계속 메시지를 가져올 수도 있습니다.

Linux UNIX **UNIX and Linux**

/var/mqm/qmgrs/qmname/errors/AMQERR01.LOG (qmname라는 각 큐 관리자에 대해)

/var/mqm/qmgrs/@SYSTEM/errors/AMQERR01.LOG (일반 오류의 경우)

UNIX, Linux, and Windows에서 **runmqlsr** 명령을 사용하여 IBM MQ 리스너 프로세스를 시작하십시오. 기본적으로 채널 연결에 대한 인바운드 요청으로 인해 리스너 프로세스가 MCA를 amqrmppa 프로세스의 스레드로 시작합니다.

```
runmqlsr -t tcp -m QM2
```

아웃바운드 연결의 경우, 다음 세 가지 방법 중 하나로 채널을 시작해야 합니다.

1. MQSC 명령 START CHANNEL로 채널 이름을 지정하여 MCATYPE 매개변수에 따라 채널을 프로세스 또는 스레드로 시작하십시오. 채널이 스레드로 시작되면 채널은 채널 시작기의 스레드가 됩니다.

   ```
   START CHANNEL(QM1.TO.QM2)
   ```

2. 제어 명령 runmqchl을 사용하여 채널을 프로세스로 시작하십시오.

   ```
   runmqchl -c QM1.TO.QM2 -m QM1
   ```

3. 채널 시작기를 사용하여 채널을 트리거하십시오.

ULW 채널 중지

MQSC 명령 STOP CHANNEL을 사용하여 채널이 활동을 중지하도록 요청할 수 있습니다. 운영자가 채널을 다시 시작할 때까지 채널이 새 메시지 배치를 시작하지 않습니다.

중지된 채널 재시작에 대한 정보는 165 페이지의 『중지된 채널 재시작』의 내용을 참조하십시오.

이 명령은 MQCHT_CLNTCONN을 제외한 모든 채널 유형에 실행될 수 있습니다.

필요한 중지 유형을 선택할 수 있습니다.

일시정지 예

```
STOP CHANNEL(QM1.TO.QM2) MODE(QUIESCE)
```

이 명령은 채널을 순서대로 닫도록 요청합니다. 현재 메시지 배치가 완료되고 채널의 다른 한쪽 끝과 함께 동기점 프로시저가 수행됩니다. 채널이 유휴 상태이면 이 명령이 수신 채널을 종료하지 않습니다.

강제 중지 예

```
STOP CHANNEL(QM1.TO.QM2) MODE(FORCE)
```

이 옵션은 채널을 즉시 중지하지만 채널의 스레드 또는 프로세스는 종료하지 않습니다. 채널이 현재 메시지 배치 처리를 완료하지 않으므로 채널을 인다우트(in-doubt) 상태로 남겨둘 수 있습니다. 일반적으로 일시정지 옵션 사용을 고려하십시오.

종료 중지 예

```
STOP CHANNEL(QM1.TO.QM2) MODE(TERMINATE)
```

이 옵션은 채널을 즉시 중지하며 채널의 스레드 또는 프로세스를 종료합니다.

중지됨 상태의 중지(일시정지) 예

```
STOP CHANNEL(QM1.TO.QM2) STATUS(STOPPED)
```

이 명령은 MODE를 지정하지 않으므로 기본값은 MODE(QUIESCE)입니다. 채널이 중지되어 자동으로 재시작할 수 없고 수동으로 시작해야 하는 것으로 요청합니다.

비활성 상태의 중지(일시정지) 예

```
STOP CHANNEL(QM1.TO.QM2) STATUS(INACTIVE)
```

이 명령은 MODE를 지정하지 않으므로 기본값은 MODE(QUIESCE)입니다. 채널이 비활성화되어 필요할 때 자동으로 재시작되는 것으로 요청합니다.

ULW 채널 이름 바꾸기

MQSC를 사용하여 메시지 채널의 이름을 바꿀 수 있습니다.

MQSC를 사용하여 다음 단계를 수행하십시오.

1. STOP CHANNEL을 사용하여 채널을 중지하십시오.
2. DEFINE CHANNEL을 사용하여 새 이름으로 중복 채널 정의를 작성하십시오.
3. DISPLAY CHANNEL을 사용하여 올바르게 작성되었는지 확인하십시오.
4. DELETE CHANNEL을 사용하여 원래 채널 정의를 삭제하십시오.

메시지 채널의 이름을 변경하려는 경우, 채널의 각 측에 하나씩 두 개의 채널 정의가 있다는 점을 유념하십시오. 양쪽에서 동시에 채널의 이름을 바꾸었는지 확인하십시오.

ULW 채널 재설정

MQSC 명령 RESET CHANNEL을 사용해서 메시지 순서 매기기 번호를 변경할 수 있습니다.

RESET CHANNEL 명령은 메시지 채널에는 사용할 수 있지만 MQI 채널(클라이언트-연결 또는 서버-연결)에는 사용할 수 없습니다. 채널이 시작된 후에 첫 메시지가 새로운 순서를 시작합니다.

이 명령이 송신자 또는 서버 채널에서 실행되면 채널이 재시작될 때 다른 측에 변경사항을 알려줍니다.

관련 개념

178 페이지의 『오브젝트 시작하기』
채널을 시작하려면 채널이 정의되고 연관된 오브젝트가 존재해야 하며 해당 채널을 사용할 수 있어야 합니다. 이 절은 그 방법을 보여줍니다.

154 페이지의 『채널 제어 기능』
채널 제어 기능은 채널을 정의, 모니터, 제어할 수 있는 기능을 제공합니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

관련 정보

RESET CHANNEL

ULW 채널상의 인다우트*(in-doubt)* 메시지 해결

송신자 또는 서버가 메시지를 인다우트(in-doubt) 상태로 유지할 때 MQSC 명령 RESOLVE CHANNEL을 사용합니다. 예를 들어, 링크의 한쪽이 종료되고 복구될 가능성이 없는 경우입니다.

RESOLVE CHANNEL 명령은 두 매개변수 중 하나(BACKOUT 또는 COMMIT)를 승인합니다. 커미트가 메시지를 제거하는 반면 백아웃은 메시지를 전송 큐로 복원합니다.

채널 프로그램은 파트너와의 세션 설정을 시도하지 않습니다. 대신 인다우트(in-doubt) 메시지를 표시하는 논리적 작업 단위 ID(LUWID)를 판별합니다. 그런 다음 필요할 경우 다음 중 하나를 내보냅니다.

- BACKOUT: 전송 큐로 메시지를 복원합니다.
- COMMIT: 전송 큐에서 메시지를 삭제합니다.

해석이 성공하려면 다음 조건이 충족되어야 합니다.

- 채널이 비활성화되어 있어야 합니다.
- 채널이 인다우트(in-doubt) 상태여야 합니다.
- 채널 유형이 송신자, 서버 또는 클러스터-송신자여야 합니다.
- 로컬 채널 정의가 존재해야 합니다.
- 로컬 큐 관리자가 실행 중이어야 합니다.

관련 개념

178 페이지의 『오브젝트 시작하기』
채널을 시작하려면 채널이 정의되고 연관된 오브젝트가 존재해야 하며 해당 채널을 사용할 수 있어야 합니다. 이 절은 그 방법을 보여줍니다.

154 페이지의 『채널 제어 기능』
채널 제어 기능은 채널을 정의, 모니터, 제어할 수 있는 기능을 제공합니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

관련 정보

RESOLVE CHANNEL

Windows Windows에서 통신 설정

분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 이를 위해서는 연결이 정의되고 사용 가능해야 합니다. 이 절에서는 IBM MQ for Windows 시스템에 사용할 수 있는 통신 양식을 사용하여 이 작업을 수행하는 방법에 대해 설명합니다.

시작하기 전에

구성 예 - IBM MQ for Windows를 참조하는 것도 좋습니다.

이 태스크 정보

Windows에서 IBM MQ에 대한 통신을 설정하는 경우 다음 통신 유형 중에서 선택할 수 있습니다.

- TCP/IP
- LU 6.2

• NetBIOS

프로시저

- Windows 시스템의 통신 설정에 대한 정보는 선택한 통신 유형에 해당하는 하위 주제를 참조하십시오.
  - 185 페이지의 『Windows에서 TCP 연결 정의』
  - 187 페이지의 『Windows에서 LU 6.2 연결 정의』
  - 188 페이지의 『Windows에서 NetBIOS 연결 정의』

관련 태스크

175 페이지의 『UNIX, Linux, and Windows에서 채널 모니터링 및 제어』
DQM의 경우 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어해야 합니다. 명령, 프로그램, IBM MQ Explorer, 채널 정의 파일, 동기화 정보 스토리지 영역을 사용하여 채널을 제어할 수 있습니다.

14 페이지의 『서버와 클라이언트 간의 연결 구성』
IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

191 페이지의 『UNIX and Linux에서 통신 설정』
DQM은 IBM MQ의 리모트 큐잉 기능입니다. 이 기능은 시스템 운영자가 제어할 수 있는 통신 링크에 대한 인터페이스를 형성하는 큐 관리자용 채널 제어 프로그램을 제공합니다. 분산 큐잉 관리로 유지되는 채널 정의에서 이러한 연결을 사용합니다.

관련 참조

15 페이지의 『사용할 통신 유형』
플랫폼에 따라 다른 통신 프로토콜을 지원합니다. 전송 프로토콜 선택은 IBM MQ MQI client 및 서버 플랫폼의 조합에 따라 다릅니다.

Windows *Windows*에서 *TCP* 연결 정의

대상의 주소를 지정하기 위해 송신측에서 채널을 구성하고 수신측에서 리스너 프로그램을 실행하여 TCP 연결을 정의합니다.

송신측

채널 정의의 연결 이름 필드에서 대상 시스템의 TCP 주소 또는 호스트 이름을 지정하십시오.

연결할 포트는 기본적으로 1414입니다. 1414는 인터넷 할당 번호 관리 기관이 IBM MQ에 지정한 포트 번호입니다.

기본값 이외의 포트 번호를 사용하려면 채널 오브젝트 정의의 연결 이름 필드에서 포트 번호를 지정하십시오.

```
DEFINE CHANNEL('channel name') CHLTYPE(SDR) +
          TRPTYPE(TCP) +
          CONNAME('OS2ROG3(1822)') +
          XMITQ('XMitQ name') +
          REPLACE
```

여기서 OS2ROG3은 리모트 큐 관리자의 DNS 이름이고 1822는 필요한 포트입니다. 이 포트는 수신측의 리스너가 청취하는 포트여야 합니다.

채널 오브젝트 정의 변경사항을 적용하려면 실행 채널을 중지했다가 재시작해야 합니다.

IBM MQ for Windows의 경우 `.ini` 파일에 포트 번호를 지정하여 기본 포트 번호를 변경할 수 있습니다.

```
TCP:
Port=1822
```

참고: 사용할 TCP/IP 포트 번호를 선택하기 위해 IBM MQ가 다음 순서에서 처음으로 발견한 포트 번호를 사용합니다.

1. 채널 정의나 명령행에 명시적으로 지정된 포트 번호. 이 번호를 사용하면 채널에 대해 기본 포트 번호를 대체할 수 있습니다.
2. `.ini` 파일의 TCP 스탠자에 지정된 포트 속성. 이 번호를 사용하면 큐 관리자에 대해 기본 포트 번호를 대체할 수 있습니다.
3. 기본값 1414. 인터넷 할당 번호 관리 기관(IANA)이 인바운드 및 아웃바운드 연결 모두에 대해 IBM MQ에 지정한 번호입니다.

qm.ini를 사용하여 설정하는 값에 대한 자세한 정보는 분산 큐잉에 대한 구성 파일 스탠자를 참조하십시오.

TCP에서 수신

수신 채널 프로그램을 시작하려면 리스너 프로그램이 시작되어 수신되는 네트워크 요청을 감지하고 관련 채널을 시작해야 합니다. IBM MQ 리스너를 사용할 수 있습니다.

수신 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다.

수신 채널 프로그램을 시작하려면 리스너 프로그램이 시작되어 수신되는 네트워크 요청을 감지하고 관련 채널을 시작해야 합니다. IBM MQ 리스너를 사용할 수 있습니다.

IBM MQ와 함께 제공되고 새 채널을 스레드로 시작하는 리스너를 실행하려면 runmqlsr 명령을 사용합니다.

runmqlsr 명령을 사용하는 기본 예:

```
runmqlsr -t tcp [-m QMNAME] [-p 1822]
```

대괄호는 선택적 매개변수를 나타냅니다. 즉 기본 큐 관리자의 경우에는 QMNAME이 필수가 아니고, 기본값 (1414)을 사용하면 포트 번호가 필수가 아닙니다. 포트 번호는 65535를 초과해서는 안 됩니다.

참고: 사용할 TCP/IP 포트 번호를 선택하기 위해 IBM MQ가 다음 순서에서 처음으로 발견한 포트 번호를 사용합니다.

1. 채널 정의나 명령행에 명시적으로 지정된 포트 번호. 이 번호를 사용하면 채널에 대해 기본 포트 번호를 대체할 수 있습니다.
2. `.ini` 파일의 TCP 스탠자에 지정된 포트 속성. 이 번호를 사용하면 큐 관리자에 대해 기본 포트 번호를 대체할 수 있습니다.
3. 기본값 1414. 인터넷 할당 번호 관리 기관(IANA)이 인바운드 및 아웃바운드 연결 모두에 대해 IBM MQ에 지정한 번호입니다.

성능을 극대화하려면 IBM MQ 리스너를 신뢰 애플리케이션으로 실행하십시오(174 페이지의 『채널 및 리스너를 신뢰 애플리케이션으로 실행』의 설명 참조). 신뢰 애플리케이션에 대한 정보는 신뢰 애플리케이션에 대한 제한사항을 참조하십시오.

TCP/IP SO_KEEPALIVE 옵션 사용

Windows SO_KEEPALIVE 옵션을 사용하려면 레지스트리에 다음 항목을 추가해야 합니다.

```
TCP:
KeepAlive=yes
```

SO_KEEPALIVE 옵션에 대한 자세한 정보는 162 페이지의 『채널의 다른 쪽이 여전히 사용 가능한지 검사』의 내용을 참조하십시오.

Windows의 경우 Windows KeepAliveTime 옵션에 대한 `HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` 레지스트리 값이 연결을 검사하기 전 경과되는 간격을 제어합니다. 기본값은 2시간입니다.

TCP 리스너 백로그 옵션 사용

TCP에서는 서버와 클라이언트 사이에 3방향 데이터 교환이 발생하지 않는 한 연결이 완료되지 않은 것으로 처리됩니다. 이러한 연결을 미해결 연결 요청이라고 합니다. 이러한 미해결 연결 요청에 대해 최대값이 설정되며 이는 리스너가 요청을 승인할 때까지 TCP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

자세한 정보 및 Windows의 특정 값은 194 페이지의 『UNIX and Linux에서 TCP 리스너 백로그 옵션 사용』 를 참조하십시오.

Windows *Windows에서 LU 6.2 연결 정의*

SNA는 두 시스템 간에 LU 6.2 대화가 설정될 수 있도록 구성되어야 합니다.

SNA가 구성되면 다음과 같이 진행하십시오.

관련 정보는 다음 표를 참조하십시오.

표 *20.* 리모트 큐 관리자 플랫폼에 대한 로컬 *Windows* 시스템의 설정

원격 플랫폼	TPNAME	TPPATH
CICS 없는 z/OS 또는 MVS™/ESA	리모트 큐 관리자에 관한 해당 부가 정보와 동일합니다.	-
z/OS or MVS/ESA using CICS	CKRC(송신자) CKSV(요청자) CKRC(서버)	-
IBM i	IBM i 시스템의 라우팅 입력 항목에 있는 비교 값과 동일합니다.	-
UNIX and Linux 시스템	리모트 큐 관리자에 관한 해당 부가 정보와 동일합니다.	*MQ_INSTALLATION_PATH*/bin/amqcrs6a
Windows	Windows 리스너 실행 명령 또는 Windows에서 TpSetup을 사용하여 정의된 호출 가능 트랜잭션 프로그램에 지정되었습니다.	*MQ_INSTALLATION_PATH*\bin\amqcrs6a

*MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

동일한 시스템에 둘 이상의 큐 관리자가 있다면, 반드시 채널 정의에서 TPname이 고유한지 확인하십시오.

TCP/IP를 통한 AnyNet® SNA 구성에 대한 최신 정보는 온라인 IBM 문서, AnyNet TCP/IP를 통한 SNA 및 SNA 노드 조작을 참조하십시오.

관련 개념

187 페이지의 『LU 6.2의 송신측(Windows)』
사용하는 LU 6.2 제품의 관리 애플리케이션에서 CPI-C측 오브젝트(기호 목적지)를 작성합니다. 채널 정의의 연결 이름 필드에 이 이름을 입력합니다. 또한 파트너에 대한 LU 6.2 링크도 작성합니다.

188 페이지의 『Windows에서 LU 6.2 수신』
수신 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다.

Windows *LU 6.2*의 송신측*(Windows)*

사용하는 LU 6.2 제품의 관리 애플리케이션에서 CPI-C측 오브젝트(기호 목적지)를 작성합니다. 채널 정의의 연결 이름 필드에 이 이름을 입력합니다. 또한 파트너에 대한 LU 6.2 링크도 작성합니다.

CPI-C측 오브젝트에서 수신 시스템의 파트너 LU 이름, TP 이름, 모드 이름을 입력하십시오. 예를 들면, 다음과 같습니다.

```
Partner LU Name             OS2ROG2
Partner TP Name             recv
Mode Name                   #INTER
```

Windows *Windows*에서 *LU 6.2* 수신

수신 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다.

수신 채널 프로그램을 시작하려면 리스너 프로그램이 수신 네트워크 요청을 감지하기 시작하고 연관된 채널을 시작해야 합니다. 대기할 TpName을 제공하여 RUNMQLSR 명령으로 이 리스너 프로그램을 시작합니다. 또는 SNA Server for Windows에서 TpStart를 사용할 수 있습니다.

RUNMQLSR 명령 사용

리스너를 시작하는 명령 예는 다음과 같습니다.

```
RUNMQLSR -t LU62 -n RECV [-m QMNAME]
```

여기서 RECV는 다른(송신) 측에서 "원격 측에서 시작할 TpName"으로 지정되는 TpName입니다. 대괄호로 묶은 마지막 부분은 선택사항이이며 기본 큐 관리자에 필요하지 않습니다.

한 시스템에서 큐 관리자가 두 개 이상 실행될 수 있습니다. 각 큐 관리자에게 다른 TpName을 할당한 후 각각에 대해 리스너 프로그램을 시작해야 합니다. 예를 들면, 다음과 같습니다.

```
RUNMQLSR -t LU62 -m QM1 -n TpName1
RUNMQLSR -t LU62 -m QM2 -n TpName2
```

성능을 극대화하려면 IBM MQ 리스너를 신뢰 애플리케이션으로 실행하십시오(신뢰 애플리케이션으로 채널 및 리스너 실행의 설명 참조). 신뢰 애플리케이션에 대한 정보는 신뢰 애플리케이션에 대한 제한사항을 참조하십시오.

다음 명령을 사용하여 비활성 큐 관리자에서 실행 중인 모든 IBM MQ 리스너를 중지할 수 있습니다.

```
ENDMQLSR [-m QMNAME]
```

Windows에서 Microsoft SNA Server 사용

TpSetup(SNA Server SDK에 있음)을 사용하여 amqcrs6a.exe를 실행하는 호출 가능한 TP를 정의하거나 다양한 레지스트리 값을 수동으로 설정할 수 있습니다. amqcrs6a.exe로 전달되어야 하는 매개변수는 다음과 같습니다.

```
-m QM -n TpName
```

여기서 *QM*은 큐 관리자 이름이고 *TpName*은 TP 이름입니다. 자세한 정보는 *Microsoft SNA Server APPC* 프로그래머 가이드 또는 *Microsoft SNA Server CPI-C* 프로그래머 가이드를 참조하십시오.

큐 관리자 이름을 지정하지 않으면 기본 큐 관리자로 가정됩니다.

Windows ***Windows*에서 *NetBIOS* 연결 정의**

NetBIOS 연결은 Windows를 실행하는 클라이언트 및 서버에만 적용됩니다. IBM MQ는 다른 IBM MQ 제품에 대한 NetBIOS 연결을 설정할 때 세 가지 유형의 NetBIOS 자원(세션, 명령 이름)을 사용합니다. 이러한 자원은 각각 NetBIOS 설치 중 선택적으로 또는 기본적으로 설정되는 한계가 있습니다.

각 실행 채널은 유형에 관계 없이 하나의 NetBIOS 세션과 하나의 NetBIOS 명령을 사용합니다. IBM NetBIOS 구현에서는 여러 프로세스가 동일한 로컬 NetBIOS 이름을 사용할 수 있습니다. 따라서 IBM MQ는 하나의 NetBIOS 이름만 사용할 수 있어야 합니다. 예를 들어, Novell의 NetBIOS 에뮬레이션과 같은 다른 벤더의 구현은 프로세스마다 다른 로컬 이름이 필요합니다. 사용 중인 NetBIOS 제품에 대한 문서에서 요구사항을 확인하십시오.

모든 경우, 각 유형의 충분한 자원이 이미 사용 가능한지 확인하거나 구성에 지정된 최대값을 늘리십시오. 값을 변경하려면 시스템을 재시작해야 합니다.

시스템 시동 중에, NetBIOS 디바이스 드라이버가 애플리케이션이 사용할 수 있는 세션, 명령, 이름의 수를 표시합니다. 이러한 자원은 동일한 시스템에서 실행 중인 NetBIOS 기반 애플리케이션이 사용할 수 있습니다. 따라서

IBM MQ가 자원을 획득하기 전에 다른 애플리케이션이 해당 자원을 이용할 수 있습니다. LAN 네트워크 관리자는 사용자를 위해 이를 명백하게 설명할 수 있어야 합니다.

관련 개념

189 페이지의 『IBM MQ 로컬 NetBIOS 이름 정의』
IBM MQ 프로세스가 사용하는 로컬 NetBIOS 이름은 세 가지 방법으로 지정될 수 있습니다.

189 페이지의 『큐 관리자 NetBIOS 세션, 명령 및 이름 한계 설정』
NetBIOS 세션, 명령 및 이름에 대한 큐 관리자 한계는 두 가지 방법으로 지정할 수 있습니다.

190 페이지의 『LAN 어댑터 번호 설정』
NetBIOS에서 채널이 작동하려면 각 측의 어댑터 지원이 호환 가능해야 합니다. IBM MQ를 사용하면 qm.ini 파일의 NETBIOS 스탠자에서 AdapterNum 값을 사용하거나 runmqlsr 명령에서 **-a** 매개변수를 지정하여 LAN 어댑터(LANA) 번호 선택을 제어할 수 있습니다.

190 페이지의 『NetBIOS 연결 시작』
연결을 시작하는 데 필요한 단계를 정의합니다.

190 페이지의 『NetBIOS 연결의 대상 리스너 정의』
NetBIOS 연결의 수신측에서 수행할 단계를 정의합니다.

Windows *IBM MQ* 로컬 *NetBIOS* 이름 정의

IBM MQ 프로세스가 사용하는 로컬 NetBIOS 이름은 세 가지 방법으로 지정될 수 있습니다.

세 가지 방법의 우선순위는 다음과 같습니다.

1. RUNMQLS 명령의 **-l** 매개변수에 지정되는 값, 예를 들어 다음과 같습니다.

   ```
   RUNMQLSR -t NETBIOS -l my_station
   ```

2. 명령으로 값이 설정되는 MQNAME 환경 변수:

   ```
   SET MQNAME= my_station
   ```

   MQNAME 값은 각 프로세스마다 설정할 수 있습니다. 또는 Windows 레지스트리에서 시스템 레벨에 설정할 수 있습니다.

   고유한 이름이 필요한 NetBIOS 구현을 사용하는 경우에는 IBM MQ 프로세스가 시작되는 각 창에서 SET MQNAME 명령을 실행해야 합니다. MQNAME 값은 임의로 정할 수 있지만 각 프로세스에 고유해야 합니다.

3. 큐 관리자 구성 파일 qm.ini의 NETBIOS 스탠자. 예를 들면, 다음과 같습니다.

   ```
   NETBIOS:
   LocalName= my_station
   ```

참고:

1. 지원되는 NetBIOS 제품 구현의 다양성으로 인해 네트워크에서 각 NetBIOS 이름을 고유하게 만드는 것이 좋습니다. 그렇지 않으면 예상치 못한 결과가 발생할 수 있습니다. NetBIOS 채널 설정에 문제점이 있고 큐 관리자 오류 로그에 NetBIOS 리턴 코드 X'15'를 표시하는 오류 메시지가 있는 경우 NetBIOS 이름 사용을 검토하십시오.
2. Windows에서는 시스템 이름을 NetBIOS 이름으로 사용할 수 없습니다. Windows에서 이미 사용하고 있기 때문입니다.
3. 송신자 채널을 시작하려면 qm.ini 파일에서 MQNAME 환경 변수 또는 LocalName을 사용하여 NetBIOS 이름이 지정되어야 합니다.

Windows 큐 관리자 *NetBIOS* 세션, 명령 및 이름 한계 설정

NetBIOS 세션, 명령 및 이름에 대한 큐 관리자 한계는 두 가지 방법으로 지정할 수 있습니다.

우선순위에 따른 방법은 다음과 같습니다.

1. RUNMQLSR 명령에 지정된 값:

   ```
   -s Sessions
   -e Names
   -o Commands
   ```

   명령에 -m 피연산자가 지정되지 않은 경우에는 기본 큐 관리자에만 값이 적용됩니다.
2. 큐 관리자 구성 파일 qm.ini의 NETBIOS 스탠자. 예를 들면, 다음과 같습니다.

   ```
   NETBIOS:

   NumSess= Qmgr_max_sess
   NumCmds= Qmgr_max_cmds
   NumNames= Qmgr_max_names
   ```

Windows *LAN* 어댑터 번호 설정

NetBIOS에서 채널이 작동하려면 각 측의 어댑터 지원이 호환 가능해야 합니다. IBM MQ를 사용하면 qm.ini 파일의 NETBIOS 스탠자에서 AdapterNum 값을 사용하거나 runmqlsr 명령에서 **-a** 매개변수를 지정하여 LAN 어댑터(LANA) 번호 선택을 제어할 수 있습니다.

NetBIOS 연결에 대해 IBM MQ에서 사용하는 기본 LAN 어댑터 번호는 0입니다. 다음과 같이 시스템에서 사용되는 번호를 확인하십시오.

Windows에서는 운영 체제를 통해 직접 LAN 어댑터 번호를 조회할 수 없습니다. 대신 Microsoft에서 사용 가능한 LANACFG.EXE 명령행 유틸리티를 사용합니다. 도구의 출력은 가상 LAN 어댑터 번호와 해당 적용 바인딩을 보여줍니다. LAN 어댑터 번호에 대한 추가 정보는 Microsoft 지식 베이스 문서 138037 *HOWTO: 32*비트 환경에서 *LANA* 번호 사용을 참조하십시오.

큐 관리자 구성 파일, qm.ini의 NETBIOS 스탠자에서 올바른 값을 지정하십시오.

```
NETBIOS:
AdapterNum= n
```

여기서 n은 이 시스템의 올바른 LAN 어댑터 번호입니다.

Windows *NetBIOS* 연결 시작

연결을 시작하는 데 필요한 단계를 정의합니다.

연결을 시작하려면 송신측에서 다음 단계를 수행하십시오.

1. MQNAME 또는 LocalName 값을 사용하여 NetBIOS 스테이션 이름을 정의하십시오.
2. 시스템에서 사용하는 LAN 어댑터 번호를 확인하고 AdapterNum을 사용하여 올바른 파일을 지정하십시오.
3. 채널 정의의 ConnectionName 필드에서 대상 리스너 프로그램이 사용하는 NetBIOS 이름을 지정하십시오. Windows에서 NetBIOS 채널은 스레드로 실행되어야 합니다. 이 작업을 수행하려면 채널 정의에서 MCATYPE(THREAD)를 지정하십시오.

   ```
   DEFINE CHANNEL (chname) CHLTYPE(SDR) +
   TRPTYPE(NETBIOS) +
   CONNAME(your_station) +
   XMITQ(xmitq) +
   MCATYPE(THREAD) +
   REPLACE
   ```

Windows *NetBIOS* 연결의 대상 리스너 정의

NetBIOS 연결의 수신측에서 수행할 단계를 정의합니다.

수신측에서 다음 단계를 수행하십시오.

1. MQNAME 또는 LocalName 값을 사용하여 NetBIOS 스테이션 이름을 정의하십시오.
2. 시스템에서 사용하는 LAN 어댑터 번호를 확인하고 AdapterNum을 사용하여 올바른 파일을 지정하십시오.

3. 수신자 채널을 정의하십시오.

```
DEFINE CHANNEL (chname) CHLTYPE(RCVR) +
TRPTYPE(NETBIOS) +
REPLACE
```

4. IBM MQ 리스너 프로그램을 시작하여 스테이션을 설정하고 접속 가능하게 하십시오. 예를 들면, 다음과 같습니다.

```
RUNMQLSR -t NETBIOS -l your_station [-m qmgr]
```

   이 명령은 `your_station`을 접속 대기 중인 NetBIOS 스테이션으로 설정합니다. NetBIOS 스테이션 이름은 NetBIOS 네트워크 전체에서 고유해야 합니다.

성능을 극대화하려면 IBM MQ 리스너를 신뢰 애플리케이션으로 실행하십시오(174 페이지의 『채널 및 리스너를 신뢰 애플리케이션으로 실행』의 설명 참조). 신뢰 애플리케이션에 대한 정보는 신뢰 애플리케이션에 대한 제한사항을 참조하십시오.

다음 명령을 사용하여 비활성 큐 관리자에서 실행 중인 모든 IBM MQ 리스너를 중지할 수 있습니다.

```
ENDMQLSR [-m QMNAME]
```

큐 관리자 이름을 지정하지 않으면 기본 큐 관리자로 가정됩니다.

Linux UNIX UNIX and Linux에서 통신 설정

DQM은 IBM MQ의 리모트 큐잉 기능입니다. 이 기능은 시스템 운영자가 제어할 수 있는 통신 링크에 대한 인터페이스를 형성하는 큐 관리자용 채널 제어 프로그램을 제공합니다. 분산 큐잉 관리로 유지되는 채널 정의에서 이러한 연결을 사용합니다.

시작하기 전에

다음 절의 내용을 참조하면 도움이 될 수 있습니다.

- AIX 구성 예 - IBM MQ for AIX
- HP-UX 구성 예 - IBM MQ for HP-UX
- Solaris 구성 예 - IBM MQ for Solaris
- Linux 구성 예 - IBM MQ for Linux

이 태스크 정보

분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 성공하려면 연결이 정의되고 사용 가능해야 합니다. 이 절에서는 이를 실행하는 방법을 설명합니다.

UNIX and Linux에서 IBM MQ 에 대한 통신을 설정할 때 다음 유형의 통신 중에서 선택할 수 있습니다.

- TCP/IP
- LU 6.2

각 채널 정의마다 전송 프로토콜(전송 유형) 속성으로만 지정해야 합니다. 큐 관리자는 하나 이상의 프로토콜을 사용할 수 있습니다.

IBM MQ MQI clients의 경우에는 다른 전송 프로토콜을 사용하는 대체 채널을 보유하는 것이 유용할 수 있습니다. IBM MQ MQI clients에 대한 자세한 정보는 IBM MQ MQI clients 개요를 참조하십시오.

프로시저

UNIX and Linux 시스템의 통신 설정에 대한 정보는 선택한 통신 유형에 해당하는 하위 주제를 참조하십시오.

- 192 페이지의 『UNIX and Linux에서 TCP 연결 정의』
- 195 페이지의 『UNIX and Linux에서 LU 6.2 연결 정의』

관련 태스크

175 페이지의 『UNIX, Linux, and Windows에서 채널 모니터링 및 제어』
DQM의 경우 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어해야 합니다. 명령, 프로그램, IBM MQ Explorer, 채널 정의 파일, 동기화 정보 스토리지 영역을 사용하여 채널을 제어할 수 있습니다.

14 페이지의 『서버와 클라이언트 간의 연결 구성』
IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

184 페이지의 『Windows에서 통신 설정』
분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 이를 위해서는 연결이 정의되고 사용 가능해야 합니다. 이 절에서는 IBM MQ for Windows 시스템에 사용할 수 있는 통신 양식을 사용하여 이 작업을 수행하는 방법에 대해 설명합니다.

관련 참조

15 페이지의 『사용할 통신 유형』
플랫폼에 따라 다른 통신 프로토콜을 지원합니다. 전송 프로토콜 선택은 IBM MQ MQI client 및 서버 플랫폼의 조합에 따라 다릅니다.

Linux UNIX *UNIX and Linux*에서 *TCP* 연결 정의

송신측의 채널 정의는 대상의 주소를 지정합니다. 리스너 또는 inet 디먼은 수신측에서의 연결용으로 구성됩니다.

송신측

채널 정의의 연결 이름 필드에서 대상 시스템의 TCP 주소 또는 호스트 이름을 지정하십시오. 연결할 포트는 기본적으로 1414입니다. 1414는 인터넷 할당 번호 관리 기관이 IBM MQ에 지정한 포트 번호입니다.

기본값 이외의 포트 번호를 사용하려면 연결 이름 필드를 변경하십시오.

```
Connection Name REMHOST(1822)
```

여기서 REMHOST는 원격 시스템의 호스트 이름이고 1822는 필요한 포트 번호입니다. 이 포트는 수신측의 리스너가 청취하는 포트여야 합니다.

또는 큐 관리자 구성 파일(qm.ini)에서 포트 번호를 지정하여 변경할 수 있습니다.

```
TCP:
Port=1822
```

qm.ini를 사용하여 설정하는 값에 대한 자세한 정보는 분산 큐잉에 대한 구성 파일 스탠자를 참조하십시오.

TCP에서 수신

TCP/IP 리스너(inet 디먼(inetd)) 또는 IBM MQ 리스너를 사용할 수 있습니다.

일부 Linux 배포는 이제 inet 디먼 대신 확장 inet 디먼(xinetd)을 사용합니다. Linux 시스템에서 확장된 inet 디먼 사용법에 대한 정보는 Linux에서 TCP 연결 설정을 참조하십시오.

관련 개념

193 페이지의 『UNIX and Linux에서 TCP/IP 리스너 사용』
UNIX and Linux에서 채널을 시작하려면 `/etc/services` 파일과 `inetd.conf` 파일을 편집해야 합니다.

194 페이지의 『UNIX and Linux에서 TCP 리스너 백로그 옵션 사용』
TCP에서는 서버와 클라이언트 사이에 3방향 데이터 교환이 발생하지 않는 한 연결이 완료되지 않은 것으로 처리됩니다. 이러한 연결을 미해결 연결 요청이라고 합니다. 이러한 미해결 연결 요청에 대해 최대값이 설정되며 이는 리스너가 요청을 승인할 때까지 TCP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

194 페이지의 『IBM MQ 리스너 사용』

IBM MQ와 함께 제공되고 새 채널을 새 스레드로 시작하는 리스너를 실행하려면 `runmqlsr` 명령을 사용합니다.

195 페이지의 『TCP/IP SO_KEEPALIVE 옵션 사용』
일부 UNIX and Linux 시스템에서는 연결을 계속 사용할 수 있는지 확인하기 전에 TCP가 대기하는 시간과 첫 번째 확인에 실패하는 경우 연결을 다시 시도하는 빈도를 정의할 수 있습니다. 이는 커널 조정 가능 매개변수이거나 명령행에서 입력할 수 있습니다.

Linux UNIX *UNIX and Linux*에서 *TCP/IP* 리스너 사용
UNIX and Linux에서 채널을 시작하려면 `/etc/services` 파일과 `inetd.conf` 파일을 편집해야 합니다.

다음 지시사항을 따르십시오.

1. `/etc/services` 파일을 편집하십시오.

   참고: `/etc/services` 파일을 편집하려면 superuser 또는 루트로 로그인해야 합니다. 이를 변경할 수는 있지만 송신측에 지정된 포트 번호와 일치해야 합니다.

   파일에 다음 행을 추가하십시오.

   ```
   MQSeries 1414/tcp
   ```

   여기서 1414는 IBM MQ에 필요한 포트 번호입니다. 포트 번호는 65535를 초과해서는 안 됩니다.
2. `inetd.conf` 파일에 행을 추가하여 amqcrsta 프로그램을 호출하십시오. 여기서 *`MQ_INSTALLATION_PATH`*은(는) IBM MQ이(가) 설치된 상위 레벨 디렉토리를 나타냅니다.

   ```
   MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta
   [-m Queue_Man_Name]
   ```

inetd가 구성 파일을 다시 읽은 후 업데이트가 활성화됩니다. 이를 수행하려면 루트 사용자 ID에서 다음 명령을 실행하십시오.

- AIX에서:

  ```
  refresh -s inetd
  ```
- HP-UX의 경우 mqm 사용자 ID에서:

  ```
  inetd -c
  ```
- Solaris 10 이상의 경우:

  ```
  inetconv
  ```
- 다른 UNIX and Linux 시스템(Solaris 9 포함):

  ```
  kill -1 process_number
  ```

inetd로 시작된 리스너 프로그램이 inetd에서 로케일을 상속하는 경우 MQMDE를 지원하지 않아(병합) 큐에 메시지 데이터로 배치됩니다. MQMDE를 유지하려면 로케일을 올바르게 설정해야 합니다. inetd로 설정된 로케일은 IBM MQ 프로세스에서 사용하는 다른 로케일에 대해 선택된 로케일과 일치하지 않을 수 있습니다. 로케일을 설정하려면 다음을 수행하십시오.

1. 로케일 환경 변수 LANG, LC_COLLATE, LC_CTYPE, LC_MONETARY, LC_NUMERIC, LC_TIME, LC_MESSAGES를 다른 IBM MQ 프로세스에 사용되는 로케일로 설정하는 쉘 스크립트를 작성하십시오.
2. 동일한 쉘 스크립트에서 리스너 프로그램을 호출하십시오.
3. 리스너 프로그램 대신 사용자의 쉘 스크립트를 호출하도록 `inetd.conf` 파일을 수정하십시오.

서버에서 큐 관리자가 두 개 이상 실행될 수 있습니다. 각 큐 관리자마다 두 개 파일 각각에 행을 추가해야 합니다. 예를 들면, 다음과 같습니다.

```
MQSeries1   1414/tcp
MQSeries2   1822/tcp
```

```
MQSeries2 stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta -m QM2
```

여기서 *MQ_INSTALLATION_PATH*는 IBM MQ가 설치되어 있는 상위 레벨 디렉토리를 나타냅니다.

이를 통해 단일 TCP 포트에서 큐잉되는 미해결 연결 요청 수에 제한이 있는 경우 오류 메시지가 생성되지 않도록 할 수 있습니다. 미해결 연결 요청 수에 대한 정보는 194 페이지의 『UNIX and Linux에서 TCP 리스너 백로그 옵션 사용』의 내용을 참조하십시오.

Linux UNIX *UNIX and Linux*에서 *TCP* 리스너 백로그 옵션 사용

TCP에서는 서버와 클라이언트 사이에 3방향 데이터 교환이 발생하지 않는 한 연결이 완료되지 않은 것으로 처리됩니다. 이러한 연결을 미해결 연결 요청이라고 합니다. 이러한 미해결 연결 요청에 대해 최대값이 설정되며 이는 리스너가 요청을 승인할 때까지 TCP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

기본 리스너 백로그 값은 194 페이지의 표 21에 표시됩니다.

표 *21. TCP/IP* 포트에서 큐잉되는 최대 미해결 연결 요청 수

서버 플랫폼	**최대 연결 요청 수**
AIX AIX	100
HP-UX HP-UX	20
Linux Linux	100
IBM i IBM i	255임)
Solaris Solaris	100
Windows Windows 서버	100
Windows Windows 워크스테이션	100

백로그가 194 페이지의 표 21에 표시된 값에 도달하면 TCP/IP 연결이 거부되고 채널을 시작할 수 없습니다.

MCA 채널의 경우에는 채널이 RETRY 상태로 진행되고 나중에 연결을 다시 시도합니다.

그러나 이 오류를 방지하기 위해 `qm.ini` 파일에 다음 항목을 추가할 수 있습니다.

```
TCP:
ListenerBacklog = n
```

이는 TCP/IP 요청에 대한 기본 최대 미해결 요청 수(194 페이지의 표 21 참조)를 대체합니다.

참고: 일부 운영 체제에서는 기본값보다 더 큰 값을 지원합니다. 필요한 경우 이 값은 연결 제한에 도달하는 것을 방지하기 위해 사용할 수 있습니다.

백로그 옵션을 활성화한 상태로 리스너를 실행하려면 다음을 수행하십시오.

- `runmqlsr -b` 명령을 사용하십시오. 또는
- BACKLOG 속성을 필수 값으로 설정하여 MQSC 명령 **DEFINE LISTENER**를 사용하십시오.

`runmqlsr` 명령에 대한 정보는 runmqlsr을 참조하십시오. DEFINE LISTENER 명령에 대한 정보는 DEFINE LISTENER를 참조하십시오.

Linux UNIX *IBM MQ* 리스너 사용

IBM MQ와 함께 제공되고 새 채널을 새 스레드로 시작하는 리스너를 실행하려면 `runmqlsr` 명령을 사용합니다.

예를 들면, 다음과 같습니다.

```
runmqlsr -t tcp [-m QMNAME] [-p 1822]
```

대괄호는 선택적 매개변수를 나타냅니다. 즉 기본 큐 관리자의 경우에는 QMNAME이 필수가 아니고, 기본값 (1414)을 사용하면 포트 번호가 필수가 아닙니다. 포트 번호는 65535를 초과해서는 안 됩니다.

성능을 극대화하려면 IBM MQ 리스너를 신뢰 애플리케이션으로 실행하십시오(174 페이지의 『채널 및 리스너를 신뢰 애플리케이션으로 실행』의 설명 참조). 신뢰 애플리케이션에 대한 정보는 신뢰 애플리케이션에 대한 제한사항을 참조하십시오.

다음 명령을 사용하여 비활성 큐 관리자에서 실행 중인 모든 IBM MQ 리스너를 중지할 수 있습니다.

```
endmqlsr [-m QMNAME]
```

큐 관리자 이름을 지정하지 않으면 기본 큐 관리자로 가정됩니다.

Linux UNIX *TCP/IP SO_KEEPALIVE* 옵션 사용

일부 UNIX and Linux 시스템에서는 연결을 계속 사용할 수 있는지 확인하기 전에 TCP가 대기하는 시간과 첫 번째 확인에 실패하는 경우 연결을 다시 시도하는 빈도를 정의할 수 있습니다. 이는 커널 조정 가능 매개변수이거나 명령행에서 입력할 수 있습니다.

SO_KEEPALIVE 옵션을 사용하려면(자세한 정보는 162 페이지의 『채널의 다른 쪽이 여전히 사용 가능한지 검사』 참조) 큐 관리자 구성 파일(qm.ini)에 다음 입력 항목을 추가해야 합니다.

```
TCP:
KeepAlive=yes
```

자세한 정보는 UNIX and Linux 시스템 문서를 참조하십시오.

Linux UNIX *UNIX and Linux*에서 *LU 6.2* 연결 정의

SNA는 두 시스템 간에 LU 6.2 대화가 설정될 수 있도록 구성되어야 합니다.

TCP/IP를 통한 SNA 구성에 대한 최신 정보는 온라인 IBM 문서, Communications Server를 참조하십시오.

SNA는 두 시스템 간에 LU 6.2 대화가 설정될 수 있도록 구성되어야 합니다.

해당 정보는 다중 플랫폼 *APPC* 구성 안내서와 다음 표를 참조하십시오.

표 22. 리모트 큐 관리자 플랫폼에 대한 로컬 *UNIX and Linux* 시스템의 설정

원격 플랫폼	TPNAME	TPPATH
CICS를 사용하지 않는 z/OS	리모트 큐 관리자에 대한 부가 정보에 있는 해당 TPName과 동일합니다.	-
CICS를 사용한 z/OS	CKRC(송신자) CKSV(요청자) CKRC(서버)	-
IBM i	IBM i 시스템의 라우팅 입력 항목에 있는 비교 값과 동일합니다.	-
UNIX and Linux 시스템	리모트 큐 관리자에 대한 부가 정보에 있는 해당 TPName과 동일합니다.	*MQ_INSTALLATION_PATH*/bin/amqcrs6a
Windows	Windows 리스너 실행 명령 또는 Windows에서 TpSetup을 사용하여 정의된 호출 가능 트랜잭션 프로그램에 지정되었습니다.	*MQ_INSTALLATION_PATH*\bin\amqcrs6a

*MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

동일한 시스템에 둘 이상의 큐 관리자가 있다면, 반드시 채널 정의에서 TPname이 고유한지 확인하십시오.

관련 개념

196 페이지의 『LU 6.2의 송신측(UNIX and Linux)』
UNIX and Linux 시스템에서 CPI-C측 오브젝트(기호 목적지)를 작성하고 채널 정의의 연결 이름 필드에 이 이름을 입력합니다. 또한 파트너에 대한 LU 6.2 링크도 작성합니다.

196 페이지의 『UNIX and Linux에서 LU 6.2 수신』
UNIX and Linux 시스템에서 수신측의 대기 첨부, LU 6.2 논리 연결 프로파일, TPN 프로파일을 작성합니다.

Linux UNIX *LU 6.2*의 송신측*(UNIX and Linux)*

UNIX and Linux 시스템에서 CPI-C측 오브젝트(기호 목적지)를 작성하고 채널 정의의 연결 이름 필드에 이 이름을 입력합니다. 또한 파트너에 대한 LU 6.2 링크도 작성합니다.

CPI-C측 오브젝트에서 수신 시스템의 파트너 LU 이름, 트랜잭션 프로그램 이름, 모드 이름을 입력하십시오. 예를 들면, 다음과 같습니다.

```
Partner LU Name                REMHOST
Remote TP Name                 recv
Service Transaction Program    no
Mode Name                      #INTER
```

HP-UX에서 APPCLLU 환경 변수를 사용하여 송신자가 사용해야 하는 로컬 LU의 이름을 지정하십시오. Solaris에서 APPC_LOCAL_LU 환경 변수를 로컬 LU 이름으로 설정하십시오.

IBM MQ가 SNA 세션 설정을 시도할 때 CPI-C가 지원하는 SECURITY PROGRAM을 사용합니다.

Linux UNIX *UNIX and Linux*에서 *LU 6.2* 수신

UNIX and Linux 시스템에서 수신측의 대기 첨부, LU 6.2 논리 연결 프로파일, TPN 프로파일을 작성합니다.

TPN 프로파일에서 실행 파일의 전체 경로와 트랜잭션 프로그램 이름을 입력하십시오.

```
Full path to TPN executable     MQ_INSTALLATION_PATH/bin/amqcrs6a
Transaction Program name       recv
User ID                        0
```

*MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다.

사용자 ID를 설정할 수 있는 시스템의 경우 mqm 그룹의 구성원인 사용자를 지정하십시오. AIX, Solaris, HP-UX의 경우 APPCTPN(트랜잭션 이름) 및 APPCLLU(로컬 LU 이름) 환경 변수를 설정하십시오(호출된 트랜잭션 프로그램에 구성 패널 사용 가능).

기본 큐 관리자 이외의 큐 관리자를 사용해야 할 수 있습니다. 이러한 경우 다음 명령을 호출하는 명령 파일을 정의하십시오.

```
amqcrs6a -m Queue_Man_Name
```

그런 다음 명령 파일을 호출하십시오.

IBM i IBM i에서 채널 모니터링 및 제어

DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다. 각 큐 관리자에는 호환 가능한 리모트 큐 관리자에 대한 상호 연결을 제어하기 위한 DQM 프로그램이 있습니다.

이 태스크 정보

다음 목록은 채널 제어 기능의 컴포넌트에 대한 간단한 설명입니다.

- 채널 정의는 큐 관리자 오브젝트로 보유됩니다.
- 채널 명령은 IBM MQ for IBM i 명령 세트의 서브세트입니다.

  GO CMDMQM 명령을 사용하여 IBM MQ for IBM i 명령의 전체 세트를 표시할 수 있습니다.
- 채널 정의 패널 또는 명령을 사용하여 다음을 수행할 수 있습니다.

- 채널 정의 작성, 복사, 표시, 변경, 삭제
- 채널 시작 및 중지, ping, 채널 순서 번호 재설정, 링크를 재설정할 수 없는 경우 인다우트(in-doubt) 메시지 해결
- 채널에 대한 상태 정보 표시

- 채널은 MQSC를 사용하여 관리할 수도 있습니다.
- 채널은 IBM MQ 탐색기를 사용하여 관리할 수도 있습니다.
- 순서 번호와 논리적 작업 단위*(LUW)* ID는 동기화 파일에 저장되며 채널 동기화 용도로 사용됩니다.

명령 및 패널을 사용하여 메시지 채널 및 연관된 오브젝트를 정의하고 메시지 채널을 모니터하고 제어할 수 있습니다. F4=프롬프트 키를 사용하여 관련 큐 관리자를 지정할 수 있습니다. 프롬프트를 사용하지 않는 경우 기본 큐 관리자가 가정됩니다. F4=프롬프트를 사용하면 관련 큐 관리자 이름 및 때때로 기타 데이터를 입력할 수 있는 추가 패널이 표시됩니다.

패널로 정의해야 하는 오브젝트는 다음과 같습니다.

- 전송 큐
- 리모트 큐 정의
- 큐 관리자 알리어스 정의
- 응답 대상 큐 알리어스 정의
- 응답 대상 로컬 큐
- 메시지 채널 정의

이 오브젝트 사용과 관련된 개념에 관한 자세한 정보는 128 페이지의 『분산 큐잉 구성』의 내용을 참조하십시오.

채널을 시작하려면 채널이 완벽하게 정의되고 연관된 오브젝트가 존재해야 하며 해당 채널을 사용할 수 있어야 합니다.

또한 채널을 실행하려면 각 채널에 대한 특정 통신 링크가 정의되고 사용 가능해야 합니다. LU 6.2 및 TCP/IP 링크가 정의되는 방법에 대한 설명은 설치에 맞는 특정 통신 안내서를 참조하십시오.

프로시저

- 오브젝트 작성 및 관련 작업에 대한 자세한 정보는 다음을 참조하십시오.

  - 198 페이지의 『IBM i에서 오브젝트 작성』
  - 198 페이지의 『IBM i에서 채널 작성』
  - 200 페이지의 『IBM i에서 채널 시작』
  - 200 페이지의 『IBM i에서 채널 선택』
  - 201 페이지의 『IBM i에서 채널 찾아보기』
  - 202 페이지의 『IBM i에서 채널 이름 바꾸기』
  - 203 페이지의 『IBM i에서 채널 상태에 대한 작업』
  - 203 페이지의 『IBM i에서 채널에 대한 작업 선택사항』

관련 개념

209 페이지의 『IBM i에 대한 통신 설정』
분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 이러한 시도가 성공하려면 연결이 정의되어 있고 사용 가능해야 합니다.

관련 태스크

14 페이지의 『서버와 클라이언트 간의 연결 구성』
IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

관련 정보

구성 예 - IBM MQ for IBM i

IBM MQ for IBM i의 메시지 채널 계획 예
IBM MQ for IBM i CL 명령

IBM i IBM i에서 오브젝트 작성

CRTMQMQ 명령을 사용하여 큐 및 알리어스 오브젝트를 작성할 수 있습니다.

큐 및 알리어스 오브젝트(예를 들어, 전송 큐, 리모트 큐 정의, 큐 관리자 알리어스 정의, 응답 대상 큐 알리어스 정의, 응답 대상 로컬 큐)를 작성할 수 있습니다.

기본 오브젝트 목록은 IBM MQ for IBM i 시스템 및 기본 오브젝트를 참조하십시오.

IBM i IBM i에서 채널 작성

채널 작성 패널에서 또는 명령행에서 CRTMQMCHL 명령을 사용하여 채널을 작성할 수 있습니다.

채널을 작성하려면 다음을 수행하십시오.

1. MQM 채널에 대한 작업 패널(WRKMQMCHL)에서 F6을 사용하십시오.

   또는 명령행에서 CRTMQMCHL 명령을 사용하십시오.

   두 가지 경우 모두 채널 작성 패널이 표시됩니다. 유형:

   - 제공된 필드에 채널 이름
   - 링크의 한쪽 측에 대한 채널 유형
2. Enter를 누르십시오.

참고: 네트워크에서 모든 채널의 이름을 고유하게 지정해야 합니다. 모든 채널을 보여주는 네트워크 다이어그램에 나타난 것처럼 채널 이름에 소스 및 대상 큐 관리자 이름을 포함하는 것이 좋습니다.

입력 항목의 유효성을 검증하고 오류가 즉시 보고됩니다. 오류를 정정하고 계속 진행하십시오.

선택한 채널 유형에 맞는 적절한 채널 설정 패널이 표시됩니다. 이전에 수집한 정보로 필드를 완료하십시오. Enter를 눌러 채널을 작성하십시오.

이 도움말 패널의 채널 정의 패널 설명과 채널 속성에서 다양한 필드의 컨텐츠를 결정할 수 있는 도움말이 제공됩니다.

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

채널 이름.......... > CHANNAME________________
채널 유형......... > * SDR__ * RCVR, * SDR, * SVR, * RQSTR ...
Message Queue Manager name     *DFT__________________________________
_____
대체............  * NO_ * NO, * 예
전송 유형........  * TCP____ *LU62, *TCP, *SYSDFTCHL
텍스트 '설명'........ > '예제 채널 정의' _______________
___________________________________
연결 이름.........  * SYSDFTCHL_________________________________
______________________________________________________________________________
______________________________________________________________________________
______________________________________________________________________________
______________________________________________________________________________
______________________________________________________________________________
________________________________________________
More...
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *26.* 채널 작성*(1)*

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

전송 큐.......TRANSMISSION_QUEUE_NAME'__________________
__________________
메시지 채널 에이전트.....  *NONE______ 이름, *SYSDFTCHL, *NONE
라이브러리.........  __________ 이름
메시지 채널 에이전트 사용자 ID.  *SYSDFTCHL__ 문자 값 ...
Coded Character Set Identifier   *SYSDFTCHL__ 0-9999, *SYSDFTCHL
일괄처리 크기...........  50 __ __ __ __ 1-9999, * SYSDFTCH1
연결 끊기 간격.......  6000 __ __ __ 1-999999, * SYSDFTCH1
짧은 재시도 간격.......  60 __ __ __ _ 0-999999999, * SYSDFTCH1
짧은 재시도 수........  10 __ __ __ _ 0-999999999, * SYSDFTCH1
긴 재시도 간격.......  1200 __ __ __ _ 0-999999999, * SYSDFTCH1
긴 재시도 횟수........  999999999 __ 0-999999999, * SYSDFTCH1
보안 종료.........  *NONE_____ 이름, *SYSDFTCHL, *NONE
라이브러리.........  __________ 이름
보안 엑시트 사용자 데이터....  * SYSDFTCHL______________________

More...
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *27.* 채널 작성*(2)*

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

송신 종료...........  *NONE______ 이름, *SYSDFTCHL, *NONE
라이브러리.........  ___________ 이름
+ for more values   __________
나감 사용자 데이터 송신......  ________________________________
+ for more values  ________________________________
수신 종료.........  *NONE_____ 이름, *SYSDFTCHL, *NONE
라이브러리.........  __________ 이름
+ for more values   __________
__________
나감 사용자 데이터 수신......  ________________________________
+ for more values  ________________________________
메세지 종료..........  *NONE_____ 이름, *SYSDFTCHL, *NONE
라이브러리.........  __________ 이름
+ for more values   __________
__________
More...
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *28.* 채널 작성*(3)*

```
Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

메세지 나감 사용자 데이터......  ________________________________
+ for more values  ____________________________
메세지 변환..........  * SYSDFTCHL_ * YES, * NO, * SYSDFTCH1
순번 랩........  999999999 __ 100-999999999, * SYSDFTCH1
최대 메시지 길이........  4194304 __ __ 0-4194304, * SYSDFTCH1
하트비트 간격........  300 __ __ __ __ - 999999999, * SYSDFTCH1
비지속 메시지 속도...  * FAST_____ * FAST, * NORMAL, * SYSDFTCH1
암호.............  *SYSDFTCHL_ 문자 값, *BLANK ...
태스크 사용자 프로파일......  *SYSDFTCHL_ 문자 값, *BLANK ...
트랜잭션 프로그램 이름.....  * SYSDFTCH1




Bottom
F3=끝내기  F4=프롬프트  F5=새로 고치기  F12=취소  F13=이 화면 사용 방법
F24=More keys
```

그림 *29.* 채널 작성*(4)*

IBM i IBM i에서 채널 시작

채널은 채널에 대한 작업 패널에서 또는 명령행에서 STRMQMCHL 명령을 사용하여 시작할 수 있습니다.

리스너는 TCP에만 유효합니다. SNA 리스너의 경우 통신 서브시스템을 구성해야 합니다.

애플리케이션이 메시지를 교환하려면 STRMQMLSR 명령을 사용하여 인바운드 연결을 위해 리스너 프로그램을 시작해야 합니다.

아웃바운드 연결의 경우, 다음 중 한 가지 방법으로 채널을 시작해야 합니다.

1. 채널 이름을 지정하여 CL 명령 STRMQMCHL로 MCATYPE 매개변수에 따라 채널을 프로세스 또는 스레드로 시작합니다. 채널이 스레드로 시작되면 채널은 채널 시작기의 스레드가 됩니다.

   ```
   STRMQMCHL CHLNAME(QM1.TO.QM2) MQNAME(MYQMGR)
   ```

2. 채널 시작기를 사용하여 채널을 트리거합니다. 큐 관리자가 시작되면 하나의 채널 시작기가 자동으로 시작됩니다. 이 자동 시작은 해당 큐 관리자의 qm.ini 파일에서 chinit 스탠자를 변경하여 제거할 수 있습니다.
3. WRKMQMCHL 명령을 사용하여 채널에 대한 작업 패널을 시작하고 옵션 14를 선택하여 채널을 시작합니다.

IBM i IBM i에서 채널 선택

채널에 대한 작업 패널에서 채널을 선택할 수 있습니다.

채널을 선택하려면 WRKMQMCHL 명령을 사용하여 채널에 대한 작업 패널에서 시작하십시오.

1. 필수 채널 이름과 연관된 옵션 필드로 커서를 움직이십시오.
2. 옵션 번호를 입력하십시오
3. Enter를 눌러 선택사항을 활성화하십시오.

여러 채널을 선택하면 옵션이 순서대로 활성화됩니다.

```
Work with MQM Channels

큐 관리자 이름...:  스라소니

Type options, press Enter.
2=Change  3=Copy  4=Delete  5=Display  8=Work with Status  13=Ping
14=Start  15=End  16=Reset  17=Resolve

Opt   Name           Type    Transport   Status
CHLNIC          *RCVR    *TCP      INACTIVE
CORSAIR.TO.MUSTANG    *SDR    *LU62    INACTIVE
FV.CHANNEL.MC.DJE1    *RCVR    *TCP     INACTIVE
FV.CHANNEL.MC.DJE2    *SDR    *TCP     INACTIVE
FV.CHANNEL.MC.DJE3    *RQSTR   *TCP     INACTIVE
FV.CHANNEL.MC.DJE4    *SVR    *TCP     INACTIVE
FV.CHANNEL.PETER     *RCVR    *TCP     INACTIVE
FV.CHANNEL.PETER.LU   *RCVR    *LU62    INACTIVE
FV.CHANNEL.PETER.LU1   *RCVR    *LU62    INACTIVE
More...
Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Create  F9=Retrieve  F12=Cancel
F21=Print
```

그림 *30.* 채널에 대한 작업

IBM i IBM i에서 채널 찾아보기

채널 표시 패널에서 또는 명령행에서 DSPMQMCHL 명령을 사용하여 채널을 찾아볼 수 있습니다.

채널 설정을 찾아보려면 WRKMQMCHL 명령을 사용하여 채널 표시 패널에서 시작하십시오.

1. 필수 채널 이름에 옵션 5(표시)를 입력하십시오.
2. Enter를 눌러 선택사항을 활성화하십시오.

여러 채널을 선택하면 순서대로 채널이 나타납니다.

또는 명령행에서 DSPMQMCHL 명령을 사용할 수 있습니다.

채널에 대한 현재 설정의 세부사항과 함께 해당 채널 표시 패널이 표시됩니다. 필드는 채널 속성에서 설명됩니다.

```
Display MQM Channel

채널 이름......... : ST.JST.2TO1
큐 관리자 이름...... : QMREL
채널 유형........ : * 스닥터
전송 유형....... : * TCP
텍스트 '설명'....... : John의 발신인이 WINSDOA1

연결 이름........ : 무스탕


전송 큐....... : WINSDOA1

메시지 채널 에이전트....:
라이브러리........:
Message channel agent user ID :  *NONE
일괄처리 간격........ : 0
일괄처리 크기.......... : 50
연결 끊기 간격...... : 6000



F3=Exit  F12=Cancel  F21=Print
```

그림 *31. TCP/IP* 채널 표시*(1)*

```
Display MQM Channel

짧은 재시도 간격....... :
짧은 재시도 수....... : 10
긴 재시도 간격....... : 6000
긴 재시도 횟수........ : 10
보안 종료........:
라이브러리........:
보안 엑시트 사용자 데이터...:
송신 나감...........:
라이브러리........:
나감 사용자 데이터 송신.....:
수신 종료.........:
라이브러리........:
나감 사용자 데이터 수신.....:
메세지 종료.........:
라이브러리........:
메시지 엑시트 사용자 데이터....:
More...


F3=Exit  F12=Cancel  F21=Print
```

그림 *32. TCP/IP* 채널 표시*(2)*

```
Display MQM Channel

순번 랩....... : 999999999
최대 메시지 길이....... : 10000
메세지 변환........ : * 아니오
하트비트 간격........  300
비지속적 메시지 속도...  * 빠르게








Bottom





F3=Exit  F12=Cancel  F21=Print
```

그림 *33. TCP/IP* 채널 표시*(3)*

IBM i IBM i에서 채널 이름 바꾸기

채널에 대한 작업 패널에서 채널 이름을 바꿀 수 있습니다.

메시지 채널의 이름을 바꾸려면 채널에 대한 작업 패널부터 시작하십시오.

1. 채널을 종료하십시오.
2. 옵션 3(복사)을 사용하여 새 이름의 복제본을 작성하십시오.
3. 옵션 5(표시)를 사용하여 이름이 올바르게 작성되었는지 확인하십시오.
4. 옵션 4(삭제)를 사용하여 원래 채널을 삭제하십시오.

메시지 채널의 이름을 바꾸려면 채널 양쪽 끝의 이름을 둘 다 동시에 변경해야 합니다.

IBM i IBM i에서 채널 상태에 대한 작업

채널 상태에 대한 작업 패널에서 채널 상태에 대한 작업을 수행할 수 있습니다.

채널 상태를 보여주는 패널 세트를 중 첫 번째 패널을 표시하려면 WRKMQMCHST 명령을 사용하십시오. 변경-보기(F11)를 선택하면 상태 패널을 순서대로 볼 수 있습니다.

또는 MQM 채널에 대한 작업 패널에서 옵션 8(상태에 대한 작업)을 선택하면 첫 번째 상태 패널도 표시됩니다.

```
MQSeries Work with Channel Status

Type options, press Enter.
5=Display  13=Ping  14=Start  15=End  16=Reset  17=Resolve


Opt Name         Connection      Indoubt   Last Seq
CARTS_CORSAIR_CHAN  GBIBMIYA.WINSDOA1    NO        1
CHLNIC        9.20.2.213       NO        3
FV.CHANNEL.PETER2   9.20.2.213       NO       6225
JST.1.2         9.20.2.201       NO       28
MP_MUST_TO_CORS    9.20.2.213       NO       100
MUSTANG.TO.CORSAIR  GBIBMIYA.WINSDOA1    NO        10
MP_CORS_TO_MUST    9.20.2.213       NO       101
JST.2.3         9.5.7.126        NO       32
PF_WINSDOA1_LU62   GBIBMIYA.IYA80020    NO       54
PF_WINSDOA1_LU62   GBIBMIYA.WINSDOA1    NO       500
ST.JCW.EXIT.2T01.CHL 9.20.2.213       NO       216

Bottom
Parameters or command
===>
F3=Exit  F4=Prompt  F5=Refresh  F6=Create  F9=Retrieve  F11=Change view
F12=Cancel  F21=Print
```

그림 *34.* 채널 상태 패널 세트의 첫 번째 패널

채널 상태에 대한 작업 패널에서 사용 가능한 옵션은 다음과 같습니다.

메뉴 옵션	설명
5=Display	채널 설정을 표시합니다.
13=Ping	적절한 경우 ping 조치를 시작합니다.
14=Start	채널을 시작합니다.
15=End	채널을 중지합니다.
16=Reset	채널 순서 번호를 재설정합니다.
17=Resolve	인다운트(in-doubt) 채널 상황을 수동으로 해결합니다.

IBM i IBM i에서 채널에 대한 작업 선택사항

WRKMQMCHL 명령으로 채널에 대한 작업 패널에 도달할 수 있으며, 이 패널에서 나열된 모든 채널의 상태를 모니터하고 선택된 패널에 대해 명령을 실행할 수 있습니다.

채널에 대한 작업 패널에서 사용 가능한 옵션은 다음과 같습니다.

메뉴 옵션	설명
204 페이지의 『2=Change』	채널 속성을 변경합니다.
204 페이지의 『3=Copy』	채널의 속성을 새 채널로 복사합니다.
204 페이지의 『4=Delete』	채널을 삭제합니다.

메뉴 옵션	설명
204 페이지의 『5=Display』	채널의 현재 설정을 표시합니다.
205 페이지의 『6=Create』	채널 작성 패널을 표시합니다.
205 페이지의 『8=상태에 대한 작업』	채널 상태 패널을 표시합니다.
206 페이지의 『13=Ping』	ping 기능을 실행하고 고정된 데이터 메시지를 원격 끝과 교환하여 인접 시스템으로의 연결을 테스트합니다.
206 페이지의 『14=Start』	선택된 채널을 시작하거나 사용 불가능한 수신자 채널을 재설정합니다.
207 페이지의 『15=End』	채널을 종료하도록 요청합니다.
207 페이지의 『16=Reset』	채널이 이 링크측에 순서 번호를 재설정하도록 요구합니다. 이 번호는 채널이 시작할 양측에서 동일해야 합니다.
208 페이지의 『17=Resolve』	채널이 다른 측과의 연결을 설정하지 않은 채 인다우트(in-doubt) 메시지를 해결하도록 요구합니다.
208 페이지의 『18=권한 표시』	IBM MQ 오브젝트 권한을 표시합니다.
208 페이지의 『19=권한 부여』	IBM MQ 오브젝트 권한을 부여합니다.
208 페이지의 『20=권한 취소』	IBM MQ 오브젝트 권한을 취소합니다.
208 페이지의 『21=오브젝트 복구』	IBM MQ 오브젝트를 복구합니다.
208 페이지의 『22=이미지 레코드』	IBM MQ 오브젝트 이미지를 기록합니다.

IBM i 2=Change

변경 옵션을 사용하여 기존 채널 정의를 변경할 수 있습니다.

변경 옵션 또는 CHGMQMCHL 명령은 채널 이름을 제외한 기존 채널 정의를 변경합니다. 채널 정의 패널에서 변경될 필드를 입력한 다음 Enter를 눌러 업데이트된 정의를 저장하십시오.

IBM i 3=Copy

복사 옵션을 사용하여 기존 채널을 복사할 수 있습니다.

복사 옵션과 CPYMQMCHL 명령을 사용하여 기존 채널을 복사합니다. 복사 패널을 통해 새 채널 이름을 정의할 수 있습니다. 그러나 사용되는 문자를 IBM i 오브젝트 이름에 유효한 문자로 제한해야 합니다. IBM MQ for IBM i 관리를 참조하십시오.

현재 설정의 세부사항을 표시하려면 복사 패널에서 Enter를 누르십시오. 새 채널 설정을 변경할 수 있습니다. Enter를 눌러 새 채널 정의를 저장하십시오.

IBM i 4=Delete

삭제 옵션을 사용하여 선택한 채널을 삭제할 수 있습니다.

요청을 확인 또는 취소할 수 있는 패널이 표시됩니다.

IBM i 5=Display

표시 옵션을 사용하여 채널에 대한 현재 정의를 표시할 수 있습니다.

이 옵션은 매개변수의 현재 값을 표시하는 필드와 사용자 입력에 대해 보호된 필드를 포함한 패널을 표시합니다.

IBM i *6=Create*

작성 옵션을 사용하여 작성 채널 패널을 표시할 수 있습니다.

작성 옵션을 사용하거나 명령행에서 CRTMQMCHL 명령을 입력하여 채널 작성 패널을 가져오십시오. 198 페이지의 그림 26부터 채널 작성 패널에 대한 예가 나와 있습니다.

이 패널에서는 IBM MQ for IBM i가 제공하는 기본값으로 채워진 필드 화면에서 채널 정의를 작성합니다. 채널 이름을 입력하고 작성하는 채널 유형과 사용될 통신 방법을 선택하십시오.

Enter를 누르면 패널이 표시됩니다. 이 패널과 나머지 패널의 모든 필수 필드에 정보를 입력한 다음 Enter를 눌러 정의를 저장하십시오.

채널 이름은 채널 양쪽에서 동일하고 네트워크 내에서 고유해야 합니다. 그러나 사용되는 문자를 IBM MQ for IBM i 오브젝트 이름에 유효한 문자로 제한해야 합니다.

모든 패널은 일부 필드에 대해 IBM MQ for IBM i가 제공하는 기본값을 갖습니다. 이러한 값은 사용자 정의하거나, 채널을 작성 또는 복사할 때 변경할 수 있습니다. 값을 사용자 정의하려면 *IBM MQ for IBM i* 시스템 관리를 참조하십시오.

각 채널 유형에 필요한 기본값으로 더미 채널을 설정하고 새 채널 정의를 작성하려고 할 때마다 해당 채널을 복사하여 사용자 고유의 채널 기본값 세트를 작성할 수 있습니다.

관련 정보
채널 속성

IBM i *8=상태에 대한 작업*

상태에 대한 작업을 사용하여 세부적인 채널 상태 정보를 볼 수 있습니다.

상태 열에는 채널이 활성 또는 비활성 상태인지 여부가 나타나며 MQM 채널에 대한 작업 패널에도 계속 표시됩니다. 옵션 8(상태에 대한 작업)을 사용하면 표시된 상태 정보를 보다 자세히 알 수 있습니다. 이 정보는 명령행에서 WRKMQMCHST 명령을 사용하여 표시될 수도 있습니다. 203 페이지의 『IBM i에서 채널 상태에 대한 작업』를 참조하십시오.

- 채널 이름
- 채널 유형
- 채널 상태
- 채널 인스턴스
- 리모트 큐 관리자
- 전송 큐 이름
- 통신 연결 이름
- 채널의 인다우트(in-doubt) 상태
- 마지막 순서 번호
- 인다우트(in-doubt) 메시지 수
- 인다우트(in-doubt) 순서 번호
- 전송 큐의 메시지 수
- 논리적 작업 단위 ID
- 인다우트(in-doubt) 논리적 작업 단위 ID
- 채널 하위 상태
- 채널 모니터링
- 헤더 압축
- 메시지 압축
- 압축 시간 표시기
- 압축 비율 표시기

- 전송 큐 시간 표시기
- 네트워크 시간 표시기
- 엑시트 시간 표시기
- 배치 크기 표시기
- 현재 공유 대화
- 최대 공유 대화

IBM i 13=Ping

ping 옵션을 사용하여 고정 데이터 메시지와 원격 끝을 교환할 수 있습니다.

IBM MQ Ping이 성공하면 시스템 감독자에게 채널을 사용할 수 있고 작동한다는 확신을 줄 수 있습니다.

Ping은 전송 큐와 대상 큐를 사용하지 않습니다. 채널 정의, 관련 통신 링크, 네트워크 설정을 사용합니다.

송신자 및 서버 채널에서만 사용할 수 있습니다. 해당 채널은 링크의 먼 측에서 시작되며 시동 매개변수 조정을 수행합니다. 오류는 정상적으로 보고됩니다.

메시지 교환의 결과가 ping 패널에 표시되는데 그 결과는 메시지를 송신한 시간, 응답을 수신한 시간 및 리턴되는 메시지 텍스트입니다.

LU 6.2에서의 Ping

ping이 IBM MQ for IBM i에서 호출되는 경우, 함수를 요청하는 사용자의 사용자 ID로 실행되는 반면 채널 프로그램이 실행되는 일반적인 방식은 QMQM 사용자 ID를 채널 프로그램에 사용하는 것입니다. 사용자 ID는 수신측으로 이동하며 수신측에서 올바른 경우에만 LU 6.2 대화가 할당될 수 있습니다.

IBM i 14=Start

시작 옵션을 사용하여 채널을 수동으로 시작할 수 있습니다.

시작 옵션은 송신자, 서버, 요청자 채널에 사용할 수 있습니다. 채널에 큐 관리자 트리거가 설정된 경우에는 이 옵션이 필요하지 않습니다.

시작 옵션은 수신자, 서버 연결, 클러스터 송신자, 클러스터 수신자 채널에도 사용됩니다. STOPPED 상태의 수신자 채널을 시작하는 것은 원격 채널에서 시작할 수 있음을 의미합니다.

시작 시 송신 MCA가 채널 정의 파일을 읽고 전송 큐를 엽니다. 채널 시작 순서가 발행되어 수신자 또는 송신자 채널의 해당 MCA가 원격으로 시작됩니다. 시작된 후에는 송신자 및 서버 프로세스가 전송 큐에 도착하는 메시지를 대기하고 도착 시 메시지를 전송합니다.

트리거를 사용하는 경우, 연속 실행 트리거 프로세스를 시작하여 이니시에이션 큐를 모니터해야 합니다. STRMQMCHLI 명령은 프로세스를 시작하는 데 사용할 수 있습니다.

채널 맨 끝에서 송신측의 채널 시작에 대한 응답으로 수신 프로세스가 시작될 수 있습니다. 이를 수행하는 방법은 LU 6.2 및 TCP/IP 연결 채널에 서로 다릅니다.

- LU 6.2 연결 채널은 채널 수신측에서 명시적인 조치가 필요하지 않습니다.
- TCP 연결 채널은 리스너 프로세스가 연속적으로 실행되어야 합니다. 이 프로세스는 링크의 원격 끝으로부터 채널 시작 요청을 기다리며 그 연결에 대한 채널 정의에 정의되어 있는 프로세스를 시작합니다.

  원격 시스템이 IBM i인 경우 STRMQMLSR 명령을 사용할 수 있습니다.

필요한 경우 시작 옵션을 사용하면 항상 채널이 재동기화됩니다.

정상적으로 시작하려면,

- 로컬 및 원격의 채널 정의가 존재해야 합니다. 수신자 또는 서버 연결 채널에 대해 적합한 채널 정의가 없으면, 채널이 자동 정의될 때 기본 정의가 자동으로 작성됩니다. 채널 자동 정의 엑시트 프로그램을 참조하십시오.
- 전송 큐가 존재하고 GET에 사용 가능해야 하며 다른 채널에서 해당 큐를 사용하지 않아야 합니다.
- 로컬 및 원격 MCA가 존재해야 합니다.
- 통신 링크가 사용 가능해야 합니다.

- 로컬 및 리모트 큐 관리자가 실행 중이어야 합니다.
- 메시지 채널이 비활성화되어 있어야 합니다.

메시지를 전송하려면 리모트 큐와 리모트 큐 정의가 존재해야 합니다.

채널 시작 요청이 승인되었음을 확인하는 메시지가 패널에 리턴됩니다. 시작 프로세스가 성공했는지 확인하려면 시스템 로그를 검사하거나 F5(화면 새로 고치기)를 누르십시오.

IBM i ***15=End***

종료를 사용하여 채널 활동을 중지할 수 있습니다.

종료 옵션을 사용하여 채널 활동을 중지하도록 요청할 수 있습니다. 채널은 더 이상 메시지를 보내지 않습니다.

Enter를 눌러 채널이 STOPPED 또는 INACTIVE가 되는지 여부와 CONTROLLED 또는 IMMEDIATE 중지를 사용하여 채널을 중지할지 여부를 선택하려면 F4를 선택하십시오. 중지된 채널이 다시 활성화되려면 운영자가 재시작해야 합니다. 비활성 채널은 트리거할 수 있습니다.

즉시 중지

즉시 중지를 사용하여 작업 단위를 완료하지 않고도 채널을 중지할 수 있습니다.

이 옵션은 채널 프로세스를 종료합니다. 결과적으로 채널이 현재 메시지 배치 처리를 완료하지 않으므로 채널을 인다우트(in-doubt) 상태로 둘 수 없습니다. 일반적으로 운영자가 제어된 중지 옵션을 사용하는 것이 좋습니다.

제어 중지

제어 중지를 사용하여 현재 작업 단위 끝에서 채널을 중지할 수 있습니다.

이 선택사항은 채널이 순서대로 종료되도록 요청합니다. 즉, 현재 메시지 배치가 완료되고 채널의 다른 한쪽 끝과 함께 동기점 프로시저가 수행됩니다.

중지된 채널 재시작

채널이 STOPPED 상태가 되면 채널을 수동으로 재시작해야 합니다. 다음 방법으로 채널을 재시작할 수 있습니다.

- **START CHANNEL** MQSC 명령 사용
- **Start Channel** PCF 명령 사용
- IBM MQ Explorer 사용
- z/OS z/OS에서 채널 시작 패널 사용
- IBM i IBM i에서 **STRMQMCHL CL** 명령 또는 WRKMQMCHL 패널에서 **START** 옵션 사용

송신자 또는 서버 채널의 경우 채널에 STOPPED 상태를 입력하면, 연관된 전송 큐가 GET(DISABLED)로 설정되었으며 트리거 설정이 해제되었습니다. 시작 요청이 수신되면 이러한 속성이 자동으로 재설정됩니다.

z/OS 채널이 RETRYING 또는 STOPPED 상태일 때 채널 시작기가 중지되면 채널 시작기가 재시작될 때 채널 상태가 기억됩니다. 그러나 채널이 STOPPED 상태일 때 채널 시작기가 중지되면 SVRCONN 채널 유형의 채널 상태가 재설정됩니다.

Multi 채널이 RETRYING 또는 STOPPED 상태일 때 큐 관리자가 중지되면 큐 관리자가 재시작될 때 채널 상태가 기억됩니다. IBM MQ 8.0 이상에서는 SVRCONN 채널에도 적용됩니다. 이전에는 채널이 STOPPED 상태일 때 채널 시작기가 중지되면 SVRCONN 채널 유형에 대한 채널 상태가 재설정되었습니다.

IBM i ***16=Reset***

재설정 옵션을 사용하여 새 메시지 순서를 강제 실행할 수 있습니다.

재설정 옵션은 메시지 순서 번호를 변경합니다. 이 옵션은 인다우트(in-doubt) 상황을 해결하기 위해 해결 옵션을 사용한 후에만 신중하게 사용하십시오. 이 옵션은 송신자 또는 서버 채널에서만 사용할 수 있습니다. 채널이 시작된 후에 첫 메시지가 새로운 순서를 시작합니다.

IBM i 17=Resolve

해결 옵션을 사용하여 전송 큐에 있는 인다우트(in-doubt) 메시지의 로컬 커미트 또는 백아웃을 강제 실행할 수 있습니다.

예를 들어, 링크의 한쪽이 종료되었고 복구한 가능성이 없으므로 송신자 또는 서버가 메시지를 인다우트(in-doubt) 상태로 유지할 때 해석 옵션을 사용합니다. 해석 옵션은 BACKOUT 또는 COMMIT 매개변수 중 하나를 수용합니다. 커미트가 메시지를 제거하는 반면 백아웃은 메시지를 전송 큐로 복원합니다.

채널 프로그램은 파트너와의 세션 설정을 시도하지 않습니다. 대신 인다우트(in-doubt) 메시지를 표시하는 논리적 작업 단위 ID(LUWID)를 판별합니다. 그런 다음 필요할 경우 다음 중 하나를 내보냅니다.

- BACKOUT: 전송 큐로 메시지를 복원합니다.
- COMMIT: 전송 큐에서 메시지를 삭제합니다.

해석이 성공하려면 다음 조건이 충족되어야 합니다.

- 채널이 비활성화되어 있어야 합니다.
- 채널이 인다우트(in-doubt) 상태여야 합니다.
- 채널 유형이 송신자 또는 서버여야 합니다.
- 로컬 채널 정의가 존재해야 합니다.
- 로컬 큐 관리자가 실행 중이어야 합니다.

IBM i 18=권한 표시

권한 표시 옵션을 사용하여 사용자가 특정 IBM MQ 오브젝트에 대해 수행하도록 권한이 부여된 조치를 표시할 수 있습니다.

선택한 오브젝트와 사용자에 대해, DSPMQAUT 명령은 IBM MQ 오브젝트에 대한 조치를 수행하기 위해 사용자에게 필요한 권한을 표시합니다. 사용자가 여러 그룹의 구성원인 경우 이 명령은 오브젝트에 대한 모든 그룹의 결합된 권한을 표시합니다.

IBM i 19=권한 부여

권한 부여 옵션을 사용하여 IBM MQ 오브젝트에 대한 조치를 수행할 수 있는 권한을 다른 사용자 또는 사용자 그룹에 부여할 수 있습니다.

GRTMQMAUT 명령은 QMQMADM 그룹의 사용자만 사용할 수 있습니다. QMQMADM의 사용자가 사용자를 이름으로 식별하거나 *PUBLIC의 모든 사용자에게 권한을 부여하여 명령에 이름이 지정된 IBM MQ 오브젝트에 대한 조치를 수행할 수 있는 권한을 다른 사용자에게 부여합니다.

IBM i 20=권한 취소

권한 취소를 사용하여 오브젝트에서 조치를 수행할 수 있는 사용자 권한을 제거할 수 있습니다.

RVKMQMAUT 명령은 QMQMADM 그룹의 사용자만 사용할 수 있습니다. QMQMADM 그룹의 사용자가 사용자를 이름으로 식별하거나 *PUBLIC의 모든 사용자 권한을 취소하여 명령에서 이름이 지정된 IBM MQ 오브젝트에 대한 조치를 수행할 수 있는 다른 사용자의 권한을 제거할 수 있습니다.

IBM i 21=오브젝트 복구

오브젝트 복구를 사용하여 IBM MQ 저널에 저장된 정보에서 손상된 오브젝트를 복원할 수 있습니다.

오브젝트 복구에서는 RCRMQMOBJ(MQ 오브젝트 다시 작성) 명령을 사용하여 명령에서 손상된 것으로 이름 지정되는 모든 오브젝트를 복구하십시오. 오브젝트가 손상되지 않은 경우 해당 오브젝트에 대해서는 조치가 수행되지 않습니다.

IBM i 22=이미지 레코드

이미지 레코드를 사용하여 일련의 오브젝트 복구에 필요한 저널 수신자의 수를 줄이고 복구 시간을 최소화할 수 있습니다.

RCDMQMIMG 명령은 명령에서 선택되는 모든 오브젝트에 대한 체크포인트를 가져옵니다. 이 명령은 통합 파일 시스템(IFS)의 현재 오브젝트 값을 오브젝트에 대한 최근 정보(예: 저널 수신자에 기록된 MQPUT 및 MQGET)와 동기화합니다.

명령이 완료되면 IFS의 오브젝트가 최신이 되고 해당 저널 수신자는 오브젝트를 복구하기 위해 표시되지 않아도 됩니다. 연결이 끊어진 저널 수신자는 다른 오브젝트를 복구하기 위해 표시되지 않아도 되는 한 분리할 수 있습니다.

IBM i IBM i에 대한 통신 설정

분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 이러한 시도가 성공하려면 연결이 정의되어 있고 사용 가능해야 합니다.

DQM은 IBM MQ for IBM i의 리모트 큐잉 기능입니다. 이는 통신 링크에 대한 인터페이스를 구성하는 IBM MQ for IBM i 큐 관리자에 대한 채널 제어 프로그램을 제공하여 이 프로그램은 시스템 운영자에 의해 제어 가능합니다.

분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 이러한 시도가 성공하려면 연결이 정의되어 있고 사용 가능해야 합니다. 이 절에서는 연결이 정의되고 사용 가능한지 확인하는 방법을 설명합니다.

채널을 시작하려면 먼저 이 절에 설명된 대로 전송 큐가 정의되어야 하고 메시지 채널 정의에 포함되어야 합니다.

다음 두 개의 IBM MQ for IBM i 시스템 간 통신 양식 중 선택할 수 있습니다.

- 209 페이지의 『IBM i에서 TCP 연결 정의』

  TCP의 경우 호스트 주소를 사용할 수 있으며 이러한 연결은 *IBM i* 통신 구성 참조에서 설명된 대로 설정됩니다.

  TCP 환경에서 각 분배 서비스에는 서비스에 액세스하기 위해 원격 시스템이 사용할 수 있는 고유 TCP 주소가 할당됩니다. TCP 주소는 호스트 이름/번호 및 포트 번호로 구성됩니다. 모든 큐 관리자는 그러한 번호를 사용하여 TCP를 통해 서로 통신합니다.
- 210 페이지의 『TCP에서 수신』

  이 양식의 통신에서는 로컬 큐 관리자를 지원하는 IBM i 시스템과 리모트 큐 관리자를 지원하는 시스템 간 물리적 링크를 제공하는 IBM i SNA 논리 장치 유형 6.2(LU 6.2)의 정의가 필요합니다. *IBM i*에서의 통신 구성에 대한 자세한 내용은 IBM i 통신 구성 참조를 참조하십시오.

또한 필요한 경우 필요한 프로세스 및 큐의 정의와 함께 트리거 지정을 준비해야 합니다.

관련 태스크

196 페이지의 『IBM i에서 채널 모니터링 및 제어』
DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다. 각 큐 관리자에는 호환 가능한 리모트 큐 관리자에 대한 상호 연결을 제어하기 위한 DQM 프로그램이 있습니다.

관련 정보

구성 예 - IBM MQ for IBM i
IBM MQ for IBM i의 메시지 채널 계획 예
IBM i의 상호통신 작업
IBM i의 채널 상태

IBM i *IBM i*에서 *TCP* 연결 정의

연결 이름 필드를 사용하여 채널 정의에서 TCP 연결을 정의할 수 있습니다.

채널 정의에는 대상이나 호스트 이름의 TCP 네트워크 주소를 포함하는 필드인 CONNECTION NAME이 있습니다(예: ABCHOST). TCP 네트워크 주소는 IPv4 점분리 십진수 양식(예: 127.0.0.1)이거나 IPv6 16진수 양식(예: 2001:DB8:0:0:0:0:0:0)일 수 있습니다. CONNECTION NAME이 호스트 이름이거나 이름 서버인 경우 IBM i 호스트 테이블은 호스트 이름을 TCP 호스트 주소로 변화하는 데 사용됩니다.

포트 번호는 전체 TCP 주소를 위해 필요합니다. 이 번호가 제공되지 않으면 기본 포트 번호 1414가 사용됩니다. 연결을 시작하는 한쪽 끝(송신자, 요청자 및 서버 채널 유형)에서 연결에 대한 선택적 포트 번호를 제공할 수 있습니다. 예를 들어, 다음과 같습니다.

```
Connection name 127.0.0.1 (1555)
```

이 경우 시작하는 한쪽 끝은 포트 1555에서 수신 프로그램에 대한 연결을 시도합니다.

TCP 리스너 백로그 옵션 사용

TCP에서는 서버와 클라이언트 사이에 3방향 데이터 교환이 발생하지 않는 한 연결이 완료되지 않은 것으로 처리됩니다. 이러한 연결을 미해결 연결 요청이라고 합니다. 이러한 미해결 연결 요청에 대해 최대값이 설정되며 이는 리스너가 요청을 승인할 때까지 TCP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

자세한 정보 및 IBM i의 특정 값은 194 페이지의 『UNIX and Linux에서 TCP 리스너 백로그 옵션 사용』 를 참조하십시오.

관련 개념

210 페이지의 『TCP에서 수신』
수신 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다. 시동 요청에 대해 응답하려면 리스너 프로그램이 시작되어 수신되는 네트워크 요청을 감지하고 연관된 채널을 시작해야 합니다. 이 리스너 프로그램은 STRMQMLSR 명령을 사용하여 시작합니다.

IBM i *TCP*에서 수신

수신 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다. 시동 요청에 대해 응답하려면 리스너 프로그램이 시작되어 수신되는 네트워크 요청을 감지하고 연관된 채널을 시작해야 합니다. 이 리스너 프로그램은 STRMQMLSR 명령을 사용하여 시작합니다.

각 큐 관리자에 대해 둘 이상의 리스너를 시작할 수 있습니다. 기본적으로 STRMQMLSR 명령은 포트 1414를 사용하지만 이 값을 대체할 수 있습니다. 기본 설정을 대체하려면 다음 명령문을 선택한 큐 관리자의 qm.ini 파일에 추가하십시오. 이 예에서 리스너는 포트 2500을 사용해야 합니다.

```
TCP:
Port=2500
```

qm.ini 파일은 이 IFS 디렉토리에 있습니다. /QIBM/UserData/mqm/qmgrs/ 큐 관리자 이름.

이 새 값은 TCP 리스너가 시작될 때에만 읽을 수 있습니다. 리스너가 이미 실행 중인 경우에는 해당 프로그램이 이 변경을 인식하지 못합니다. 새 값을 사용하려면 리스너를 중지하고 STRMQMLSR 명령을 다시 발행하십시오. 이제 STRMQMLSR 명령을 사용할 때마다 리스너가 새 포트로 기본값 설정됩니다.

또는 STRMQMLSR 명령에서 다른 포트 번호를 지정할 수 있습니다. 예를 들면 다음과 같습니다.

```
STRMQMLSR MQMNAME( queue manager name ) PORT(2500)
```

이렇게 변경하면 리스너 작업 중에 리스너가 새 포트로 기본값 설정됩니다.

TCP SO_KEEPALIVE 옵션 사용

SO_KEEPALIVE 옵션을 사용하려는 경우(자세한 정보는 162 페이지의 『채널의 다른 쪽이 여전히 사용 가능한지 검사』 참조) 다음 입력 항목을 큐 관리자 구성 파일(IFS 디렉토리 /QIBM/UserData/mqm/qmgrs/ 큐 관리자 이름에 있는 qm.ini)에 추가해야 합니다.

```
TCP:
KeepAlive=yes
```

그런 다음, 다음 명령을 발행해야 합니다.

```
CFGTCP
```

옵션 3(TCP 속성 변경)을 선택하십시오. 이제 시간 간격을 분 단위로 지정할 수 있습니다. 1 - 40320분 범위의 값을 지정할 수 있으며 기본값은 120입니다.

TCP 리스너 백로그 옵션 사용

TCP에서 수신할 때 최대 미해결 연결 요청 수가 설정됩니다. 이 수는 리스너가 요청을 승인할 때까지 TCP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

IBM i에서 기본 리스너 백로그 값은 255입니다. 백로그가 이 값에 도달하면 TCP 연결이 거부되고 채널을 시작할 수 없습니다.

MCA 채널의 경우에는, 채널이 RETRY 상태가 되고 나중에 연결을 재시도하게 됩니다.

클라이언트 연결의 경우, 클라이언트가 MQCONN에서 MQRC_Q_MGR_NOT_AVAILABLE 이유 코드를 수신하며 나중에 연결을 재시도할 수 있습니다.

그러나 이 오류를 방지하기 위해 qm.ini 파일에 다음 입력 항목을 추가할 수 있습니다.

```
ListenerBacklog = n
```

이는 TCP 리스너에 대한 최대 미해결 요청 수(255)를 대체합니다.

참고: 일부 운영 체제에서는 기본값보다 더 큰 값을 지원합니다. 필요한 경우 이 값은 연결 제한에 도달하는 것을 방지하기 위해 사용할 수 있습니다.

IBM i *IBM i*에서 *LU 6.2* 연결 정의

완전한 LU 6.2 연결의 모드 이름, TP 이름 및 연결 이름을 사용하여 LU 6.2 통신 세부사항을 정의합니다.

링크를 시작된 한쪽 끝에는 이 CSI 오브젝트를 보완하는 라우팅 입력 항목 정의가 있어야 합니다. 원격 LU 6.2 시스템에서 작업 요청 관리에 대한 추가 정보는 *IBM i* 프로그래밍*:* 작업 관리 안내서에서 확인할 수 있습니다.

해당 정보는 다중 플랫폼 *APPC* 구성 안내서와 다음 표를 참조하십시오.

표 *23.* 리모트 큐 관리자 플랫폼에 대한 로컬 *IBM i* 시스템의 설정

원격 플랫폼	TPNAME
z/OS 또는 MVS	리모트 큐 관리자에 관한 해당 부가 정보와 동일합니다.
IBM i	IBM i 시스템의 라우팅 입력 항목에 있는 비교 값과 동일합니다.
UNIX and Linux 시스템	원격 LU 6.2 구성에서 정의된 호출 가능한 트랜잭션 프로그램입니다.
Windows	Windows 리스너 실행 명령 또는 Windows에서 TpSetup을 사용하여 정의된 호출 가능 트랜잭션 프로그램에 지정되었습니다.

같은 컴퓨터에 둘 이상의 큐 관리자가 있는 경우 채널 정의의 TPname이 고유한지 확인하십시오.

관련 개념

211 페이지의 『시작하는 한쪽 끝(송신자)』
CRTMQMCHL 명령을 사용하여 전송 유형 *LU62의 채널을 정의할 수 있습니다.

214 페이지의 『시작된 한쪽 끝(수신자)』
CRTMQMCHL 명령을 사용하여 전송 유형 *LU62로 메시지 채널 링크의 수신 측을 정의할 수 있습니다.

IBM i 시작하는 한쪽 끝*(*송신자*)*

CRTMQMCHL 명령을 사용하여 전송 유형 *LU62의 채널을 정의할 수 있습니다.

CSI 오브젝트의 사용은 IBM MQ for IBM i 5.3 이상에서 선택적입니다.

시작하는 한쪽 끝 패널은 그림 LU 6.2 통신 설정 패널 - 시작하는 한쪽 끝에 표시되어 있습니다. 표시된 대로 전체 패널을 확보하려면 첫 번째 패널에서 F10을 누르십시오.

```
Create Comm Side Information (CRTCSI)

Type choices, press Enter.

Side information . . . . . . . . > WINSDOA1   Name
Library . . . . . . . . . . . >  QSYS    Name, *CURLIB
Remote location . . . . . . . . > WINSDOA1   Name
Transaction program . . . . . . > MQSERIES

Text 'description' . . . . . . .  *BLANK


Additional Parameters

Device . . . . . . . . . . . . .  *LOC     Name, *LOC
Local location . . . . . . . . .  *LOC     Name, *LOC, *NETATR
Mode . . . . . . . . . . . . . .  JSTMOD92   Name, *NETATR
Remote network identifier . . .  *LOC     Name, *LOC, *NETATR, *NONE
Authority . . . . . . . . . . .  *LIBCRTAUT  Name, *LIBCRTAUT, *CHANGE...

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

그림 *35. LU 6.2* 통신 설정 패널 - 시작하는 한쪽 끝

시작하는 한쪽 끝 필드를 다음과 같이 완료하십시오.

부가 정보

이 정의에 작성할 부가 정보 오브젝트를 저장하는 데 사용되는 이름(예: WINSDOA1)을 제공합니다.

참고: LU 6.2의 경우 메시지 채널 정의와 통신 연결 간 링크는 송신 측 메시지 채널 정의의 **연결 이름** 필드입니다. 이 필드는 CSI 오브젝트의 이름을 포함합니다.

라이브러리

이 정의가 저장되는 라이브러리의 이름입니다.

CSI 오브젝트는 메시지 채널을 지원하는 프로그램에 액세스 가능한 라이브러리(예: QSYS, QMQM, QGPL)에서 사용 가능해야 합니다.

이름이 부정확하거나 누락되었거나 찾을 수 없는 경우 채널 시작 시 오류가 발생합니다.

원격 위치

프로그램과 통신할 원격 위치 이름을 지정합니다.

즉, 두 개의 시스템 간 통신 링크에 사용되는 디바이스 설명에 정의된 대로 이 필수 매개변수에는 원격 시스템에 있는 파트너의 논리 장치 이름이 들어 있습니다.

원격 위치 이름은 원격 시스템에서 DSPNETA 명령을 발행하고 기본 로컬 위치 이름을 확인함으로써 찾을 수 있습니다.

트랜잭션 프로그램

원격 시스템에서 시작할 트랜잭션 프로그램의 이름(최대 64자)을 지정합니다. 이 이름은 트랜잭션 프로그램 이름, 프로그램 이름, 채널 이름 또는 라우팅 입력 항목의 **비교 값**과 일치하는 문자열일 수 있습니다.

필수 매개변수입니다.

참고: SNA 서비스 트랜잭션 프로그램 이름을 지정하려면 서비스 트랜잭션 프로그램 이름의 16진 표현을 입력하십시오. 예를 들어, 21F0F0F1의 16진 표현으로 서비스 트랜잭션 프로그램 이름을 지정하려면 X'21F0F0F1'를 입력합니다.

SNA 서비스 트랜잭션 이름에 대한 자세한 정보는 LU 유형 6.2용 *SNA* 트랜잭션 프로그래머 참조 매뉴얼에 있습니다.

수신 측이 다른 IBM i 시스템인 경우 **트랜잭션 프로그램** 이름은 송신 측 CSI 오브젝트를 수신 측 라우팅 입력 항목과 일치시키는 데 사용됩니다. 이 이름은 대상 IBM i 시스템에서 각 큐 관리자에 대해 고유해야 합니다. 시작된 한쪽 끝(수신자) 아래에 있는 **호출할 프로그램** 매개변수를 참조하십시오. 라우팅 입력 항목 추가 패널에 있는 **비교 데이터: 비교 값** 매개변수도 참조하십시오.

텍스트 설명
이 연결의 사용 목적을 상기시키는 설명(최대 50자)입니다.

디바이스
원격 시스템에 사용되는 디바이스 설명의 이름을 지정합니다. 가능한 값은 다음과 같습니다.

***LOC**
디바이스가 시스템에 의해 판별됩니다.

Device-name
원격 위치와 연관된 디바이스의 이름을 지정합니다.

로컬 위치
로컬 위치 이름을 지정합니다. 가능한 값은 다음과 같습니다.

***LOC**
로컬 위치 이름이 시스템에 의해 판별됩니다.

***NETATR**
시스템 네트워크 속성에서 지정되는 LCLLOCNAME 값이 사용됩니다.

Local-location-name
위치 이름을 지정합니다. 원격 위치에 특정 위치 이름을 표시하려는 경우 로컬 위치를 지정하십시오. 위치 이름은 DSPNETA 명령을 사용해서 찾을 수 있습니다.

모드
세션을 제어하는 데 사용되는 모드를 지정합니다. 이 이름은 CPI(Common Programming Interface)- 통신 Mode_Name과 같습니다. 가능한 값은 다음과 같습니다.

***NETATR**
네트워크 속성에 있는 모드가 사용됩니다.

공백
8개의 공백 문자가 사용됩니다.

모드 이름
원격 위치의 모드 이름을 지정합니다.

참고: 이 모드는 통신 세션의 전송 우선순위를 판별하기 때문에 송신되는 메시지의 우선순위에 따라 여러 모드(예: MQMODE_HI, MQMODE_MED, MQMODE_LOW)를 정의하는 것이 유용할 수 있습니다. (같은 위치를 가리키는 CSI를 둘 이상 가질 수 있습니다.)

원격 네트워크 ID
원격 위치에 사용되는 원격 네트워크 ID를 지정합니다. 가능한 값은 다음과 같습니다.

***LOC**
원격 위치에 대한 원격 네트워크 ID가 사용됩니다.

***NETATR**
네트워크 속성에서 지정되는 원격 네트워크 ID가 사용됩니다.

***NONE**
원격 네트워크에 이름이 없습니다.

Remote-network-id
원격 네트워크 ID를 지정합니다. 원격 위치에서 DSPNETA 명령을 사용하여 이 네트워크 ID의 이름을 찾으십시오. 원격 위치에 있는 '로컬 네트워크 ID'입니다.

권한
오브젝트에 대한 특정 권한이 없는 사용자, 권한 목록에 없는 사용자 및 오브젝트에 대한 특정 권한이 없는 그룹 프로파일을 사용하는 사용자에게 제공하는 권한을 지정합니다. 가능한 값은 다음과 같습니다.

***LIBCRTAUT**
오브젝트에 대한 공용 권한을 지정된 라이브러리의 CRTAUT 매개변수에서 가져옵니다. 이 값은 작성 시 판별됩니다. 오브젝트가 작성된 후에 라이브러리의 CRTAUT 값이 변경되면 새 값이 기존 오브젝트에 적용되지 않습니다.

***CHANGE**
변경 권한을 통해 사용자가 오브젝트에 대한 기본 기능을 수행할 수 있지만 사용자가 오브젝트를 변경할 수는 없습니다. 변경 권한은 오브젝트 조작 권한 및 모든 데이터 권한을 제공합니다.

*** 모든**
: 사용자는 소유자로 제한된 조작 또는 권한 목록 관리 권한으로 제어되는 조작을 제외한 모든 조작을 수행할 수 있습니다. 사용자는 오브젝트의 존재를 제어하고 오브젝트에 대한 보안을 지정하며 오브젝트를 변경하고 오브젝트에서 기본 기능을 수행할 수 있습니다. 사용자는 오브젝트의 소유권을 변경할 수 있습니다.

***USE**
: 사용 권한은 오브젝트 조작 권한 및 읽기 권한을 제공합니다.

***EXCLUDE**
: 제외 권한은 사용자가 오브젝트에 액세스하지 못하게 합니다.

Authorization-list
: 부가 정보에 사용되는 권한이 포함된 권한 목록의 이름을 지정합니다.

IBM i 시작된 한쪽 끝(수신자)

CRTMQMCHL 명령을 사용하여 전송 유형 *LU62로 메시지 채널 링크의 수신 측을 정의할 수 있습니다.

CONNECTION NAME 필드는 공백으로 두고 해당 세부사항이 채널의 송신 측과 일치하는지 확인하십시오. 자세한 내용은 채널 작성의 내용을 참조하십시오.

시작하는 한쪽 끝에서 수신 채널을 시작할 수 있게 하려면 시작된 한쪽 끝의 서브시스템에 라우팅 입력 항목을 추가하십시오. 서브시스템은 LU 6.2 세션에서 사용되는 APPC 디바이스를 할당하는 서브시스템이어야 합니다. 따라서 해당 디바이스에 대해 유효한 통신이 있어야 합니다. 라우팅 입력 항목은 메시지 채널의 수신 측을 시작하는 프로그램을 호출합니다.

IBM i 명령(예: ADDRTGE)을 사용하여 통신 세션에 의해 시작되는 링크 측을 정의하십시오.

시작된 한쪽 끝 패널이 LU 6.2 통신 설정 패널 - 라우팅 입력 항목 추가에 표시됩니다.

```
Add Routing Entry (ADDRTGE)

Type choices, press Enter.

Subsystem description . . . . .  QCMN      Name
Library . . . . . . . . . . .   *LIBL     Name, *LIBL, *CURLIB
Routing entry sequence number .  1        1-9999
Comparison data:
Compare value . . . . . . . .  MQSERIES

Starting position . . . . . .  37      1-80
Program to call . . . . . . . .  AMQCRC6B    Name, *RTGDTA
Library . . . . . . . . . . .  QMAS400    Name, *LIBL, *CURLIB
Class . . . . . . . . . . . . .  *SBSD     Name, *SBSD
Library . . . . . . . . . . .  *LIBL     Name, *LIBL, *CURLIB
Maximum active routing steps . .  *NOMAX    0-1000, *NOMAX
Storage pool identifier . . . .  1        1-10



Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
```

그림 *36. LU 6.2* 통신 설정 패널 - 시작된 한쪽 끝

서브시스템 설명
: 이 정의가 상주하는 서브시스템의 이름입니다. IBM i WRKSBSD 명령을 사용하여 라우팅 입력 항목에 대해 적절한 서브시스템 설명을 보고 업데이트하십시오.

라우팅 입력 항목 순서 번호
: 이 통신 정의를 식별하기 위한 서브시스템의 고유 번호입니다. 1 - 9999 범위의 값을 사용할 수 있습니다.

비교 데이터: 비교 값
: 그림 1에 표시된 대로 **트랜잭션 프로그램** 매개변수에 의해 세션이 시작될 때 수신되는 문자열과 비교할 텍스트 문자열입니다. 문자열은 송신자 CSI의 트랜잭션 프로그램 필드에서 파생됩니다.

비교 데이터: 시작 위치
문자열 중 비교를 시작할 문자 위치입니다.

참고: 시작 위치 필드는 문자열 중 비교를 위한 문자 위치이고 이 위치는 항상 37입니다.

호출할 프로그램
세션을 시작하기 위해 호출할 인바운드 메시지 프로그램을 실행하는 프로그램의 이름입니다.

프로그램 AMQCRC6A가 기본 큐 관리자에 대해 호출됩니다. 이 프로그램은 IBM MQ for IBM i에 제공되어 환경을 설정한 다음 AMQCRS6A를 호출합니다.

추가 큐 관리자의 경우:

- 각 큐 관리자에는 해당 라이브러리에 특정 LU 6.2 호출 가능 프로그램이 있습니다. 이 프로그램을 AMQCRC6B라고 하며 큐 관리자가 작성될 때 자동으로 생성됩니다.
- 각 큐 관리자에서는 고유 라우팅 데이터가 있는 특정 라우팅 입력 항목이 추가되어야 합니다. 이 라우팅 데이터는 요청하는 시스템(시작하는 한쪽 끝(송신자))에서 제공되는 **트랜잭션 프로그램** 이름과 일치해야 합니다.

예는 LU 6.2 통신 설정 패널 - 라우팅 입력 항목 표시에 표시되어 있습니다.

```
Display Routing Entries
System:  MY400
Subsystem description:  QCMN      Status:  ACTIVE

Type options, press Enter.
5=Display details

Start
Opt  Seq Nbr  Program    Library    Compare Value   Pos
10   *RTGDTA           'QZSCSRVR'     37
20   *RTGDTA           'QZRCSRVR'     37
30   *RTGDTA           'QZHQTRG'     37
50   *RTGDTA           'QVPPRINT'     37
60   *RTGDTA           'QNPSERVR'     37
70   *RTGDTA           'QNMAPINGD'    37
80   QNMAREXECD  QSYS      'AREXECD'     37
90   AMQCRC6A   QMQMBW    'MQSERIES'     37
100   *RTGDTA           'QTFDWNLD'     37
150   *RTGDTA           'QMFRCVR'     37



F3=Exit  F9=Display all detailed descriptions  F12=Cancel
```

그림 *37. LU 6.2* 통신 설정 패널 - 시작된 한쪽 끝

LU 6.2 통신 설정 패널 - 라우팅 입력 항목 표시에서 순서 번호 90은 기본 큐 관리자를 나타내고 IBM MQ for IBM i의 이전 릴리스(즉, V3R2, V3R6, V3R7 및 V4R2) 구성과의 호환성을 제공합니다. 이러한 릴리스는 한 큐 관리자만 허용합니다. 순서 번호 92 및 94는 라이브러리 QMALPHA 및 QMBETA에서 작성되는 ALPHA 및 BETA라고 하는 두 큐 관리자를 나타냅니다.

참고: 여러 라우팅 데이터를 사용하여 각 큐 관리자에 대해 둘 이상의 라우팅 입력 항목을 가질 수 있습니다. 이러한 입력 항목은 사용되는 클래스에 따라 여러 우선순위 옵션을 제공합니다.

클래스
이 라우팅 입력 항목을 통해 시작된 단계에 사용되는 클래스의 이름 및 라이브러리입니다. 이 클래스는 라우팅 단계 실행 환경의 속성을 정의하고 작업 우선순위를 지정합니다. 적절한 클래스 입력 항목을 지정해야 합니다. 예를 들어, WRKCLS 명령을 사용하여 기존 클래스를 표시하거나 클래스를 작성하십시오. 원격 LU 6.2 시스템에서 작업 요청 관리에 대한 추가 정보는 *IBM i* 프로그래밍*:* 작업 관리 안내서에서 확인할 수 있습니다.

작업 관리에 대한 참고

AMQCRS6A 작업은 다른 IBM MQ 작업과 같은 방식으로 시작되지 않기 때문에 작업 관리에 문서화되어 있는 일반적인 IBM i 작업 관리 기능을 이용할 수 없습니다. LU62 수신자 작업의 런타임 특성을 변경하려면 다음 변경사항 중 하나를 작성할 수 있습니다.

- AMQCRS6A 작업에 대해 라우팅 입력 항목에서 지정되는 클래스 설명을 대체
- 통신 입력 항목에서 작업 설명 변경

통신 구성 작업에 대한 자세한 정보는 *IBM i* 프로그래밍*:* 작업 관리 안내서를 참조하십시오.

큐 관리자 클러스터 구성

클러스터는 초기 구성 및 진행 중인 관리를 모두 단순화하는 방식으로 큐 관리자를 상호 연결하기 위한 메커니즘을 제공합니다. 클러스터 컴포넌트를 정의하고 클러스터를 작성 및 관리할 수 있습니다.

시작하기 전에

클러스터링 개념에 대한 소개는 클러스터의 내용을 참조하십시오.

큐 관리자 클러스터를 설계할 때 몇 가지를 결정해야 합니다. 클러스터 예 및 클러스터 설계를 참조하십시오.

관련 태스크

333 페이지의 『다른 큐 관리자로 클러스터 토픽 정의 이동』
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

관련 정보

DELETE TOPIC

클러스터의 컴포넌트 정의

클러스터는 큐 관리자, 클러스터 채널 및 클러스터 큐로 구성됩니다. 클러스터 큐를 정의하고 기본 클러스터 오브젝트의 몇 가지 측면을 수정할 수 있습니다. 개별 클러스터 송신자 채널과 전송 큐 사이의 관계 및 자동 정의된 채널에 대한 구성 및 상태 정보를 가져올 수 있습니다.

각 클러스터 컴포넌트 정의에 대한 정보는 다음 하위 주제를 참조하십시오.

관련 태스크

226 페이지의 『새 클러스터 설정』
다음 지시사항에 따라 클러스터 예를 설정합니다. 개별 지시사항에서는 TCP/IP, LU 6.2 및 단일 전송 큐 또는 다중 전송 큐에서의 클러스터 설정을 설명합니다. 한 큐 관리자에서 다른 큐 관리자로 메시지를 송신하여 클러스터가 작동하는지 테스트합니다.

237 페이지의 『클러스터에 큐 관리자 추가』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 단일 클러스터 전송 큐 `SYSTEM.CLUSTER.TRANSMIT.QUEUE`를 사용하여 전송됩니다.

관련 정보

클러스터의 컴포넌트
클러스터 채널
클러스터 토픽 정의

클러스터 큐 정의

클러스터 큐는 클러스터 큐 관리자에 의해 호스팅되며 클러스터의 다른 큐 관리자가 사용할 수 있는 큐입니다. 큐가 호스팅되는 클러스터 큐 관리자의 로컬 큐로 클러스터 큐를 정의합니다. 큐가 속한 클러스터의 이름을 지정하십시오.

다음 예는 CLUSTER 옵션과 함께 클러스터 큐를 정의하는 **runmqsc** 명령을 보여줍니다.

```
DEFINE QLOCAL(Q1) CLUSTER(SALES)
```

클러스터 큐 정의는 클러스터의 다른 큐 관리자에 통지됩니다. 다른 큐 관리자는 해당하는 리모트 큐 정의 없이도 클러스터 큐에 메시지를 넣을 수 있습니다. 클러스터 큐는 클러스터 이름 목록을 사용하여 둘 이상의 클러스터에 통지될 수 있습니다.

큐가 통지되면 클러스터의 큐 관리자가 해당 큐에 메시지를 넣을 수 있습니다. 메시지를 넣으려면 큐 관리자가 전체 저장소에서 큐가 호스팅되고 있는 위치를 찾아야 합니다. 그런 다음 메시지에 몇 가지 라우팅 정보를 추가하고 클러스터 전송 큐에 메시지를 넣습니다.

z/OS 클러스터 큐는 IBM MQ for z/OS에서 큐 공유 그룹의 멤버가 공유하는 큐일 수 있습니다.

바인딩

둘 이상의 큐 관리자가 같은 클러스터 큐의 인스턴스를 호스팅하는 클러스터를 작성할 수 있습니다. 순서에 있는 모든 메시지가 큐의 동일한 인스턴스로 송신되는지 확인하십시오. MQOPEN 호출 시 MQOO_BIND_ON_OPEN 옵션을 사용하여 일련의 메시지를 특정 큐에 바인딩할 수 있습니다.

클러스터 전송 큐

큐 관리자는 클러스터의 다른 큐 관리자에 대한 메시지를 다중 전송 큐에 저장할 수 있습니다. 서로 다른 두 가지 방법으로 다중 클러스터 전송 큐에서 메시지를 저장하도록 큐 관리자를 구성할 수 있습니다. 큐 관리자 속성 **DEFCLXQ**를 CHANNEL로 설정하면 클러스터 송신자 채널마다 다른 클러스터 전송 큐가 SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE에서 자동으로 작성됩니다. 하나 이상의 클러스터 송신자 채널과 일치하도록 CLCHNAME 전송 큐 옵션을 설정할 경우 큐 관리자가 일치하는 채널에 대한 메시지를 해당 전송 큐에 저장할 수 있습니다.

주의: IBM WebSphere MQ 7.5보다 이전 버전의 제품에서 업그레이드된 큐 관리자와 함께 전용 SYSTEM.CLUSTER.TRANSMIT.QUEUES을(를) 사용하는 경우, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE에 SHARE/NOSHARE 옵션이 **SHARE**로 설정되어 있는지 확인하십시오.

다른 큐 관리자에 있는 클러스터 큐에 대한 메시지는 송신되기 전에 클러스터 전송 큐에 배치됩니다. 클러스터 송신자 채널은 클러스터 전송 큐에서 다른 큐 관리자의 클러스터 수신자 채널로 메시지를 전송합니다. 기본적으로 한 시스템에서 정의한 클러스터 전송 큐는 다른 클러스터 큐 관리자로 전송할 모든 메시지를 보유합니다. 이 큐를 SYSTEM.CLUSTER.TRANSMIT.QUEUE라고 합니다. 클러스터에 속한 큐 관리자는 이 클러스터 전송 큐에 있는 메시지를 같은 클러스터에 있는 다른 큐 관리자에게 송신할 수 있습니다.

단일 SYSTEM.CLUSTER.TRANSMIT.QUEUE 큐에 대한 정의가 z/OS를 제외한 모든 큐 관리자에서 기본적으로 작성됩니다. z/OS z/OS에서 정의는 제공되는 샘플 **CSQ4INSX**로 정의될 수 있습니다.

다중 전송 큐를 사용하여 메시지를 다른 클러스터된 큐 관리자에게로 전송하도록 큐 관리자를 구성할 수 있습니다. 수동으로 추가 클러스터 전송 큐를 정의하거나 큐 관리자가 이 큐를 자동으로 작성하게 할 수 있습니다.

큐 관리자가 자동으로 큐를 작성하게 하려면 큐 관리자 속성 DEFCLXQ를 SCTQ에서 CHANNEL로 변경하십시오. 결과적으로 이 큐 관리자는 작성되는 각 클러스터 송신자 채널에 대해 개별 클러스터 전송 큐를 작성합니다. 전송 큐는 모델 큐 SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE로부터 영구적 동적 큐로 작성됩니다. 각 영구 동적 큐의 이름은 SYSTEM.CLUSTER.TRANSMIT. *ChannelName*입니다. 각 영구적 동적 클러스터 전송 큐와 연관된 클러스터 송신자 채널의 이름은 로컬 전송 큐 속성 CLCHNAME에서 설정됩니다. 리모트 클러스터된 큐 관리자에 대한 메시지는 SYSTEM.CLUSTER.TRANSMIT.QUEUE가 아니라 연관된 클러스터 송신자 채널의 영구적 동적 클러스터 전송 큐에 배치됩니다.

클러스터 전송 큐를 수동으로 작성하려면 USAGE 속성이 XMITQ로 설정되고 CLCHNAME 속성이 하나 이상의 클러스터 송신자 채널로 해석되는 일반 채널 이름으로 설정된 로컬 큐를 작성하십시오. ClusterChannelName의 내용을 참조하십시오. 클러스터 전송 큐를 수동으로 작성하는 경우 이 전송 큐를 단일 클러스터 송신자 채널과 연관시킬지 또는 다중 클러스터 송신자 채널과 연관시킬지 선택할 수 있습니다. CLCHNAME 속성은 일반 이름이며 이는 이름에 여러 개의 와일드카드 문자 "*"를 배치할 수 있음을 의미합니다.

큐 관리자를 전체 저장소에 연결하기 위해 수동으로 작성하는 초기 클러스터 송신자 채널을 제외한 클러스터 송신자 채널은 자동으로 작성됩니다. 이러한 클러스터 송신자 채널은 클러스터 큐 관리자에 전송할 메시지가 있는 경우에 자동으로 작성됩니다. 목적지 큐 관리자의 해당 특정 클러스터에 대한 클러스터 메시지를 수신하는 클러스터 수신자 채널의 이름과 같은 이름으로 작성됩니다.

클러스터 수신자 채널에 대한 이름 지정 규칙을 따르는 경우 여러 종류의 클러스터 메시지를 서로 다른 전송 큐로 필터링하는 CLCHNAME에 일반 값을 정의할 수 있습니다. 예를 들어, *ClusterName. QmgrName*의 클러스터 수신자 채널에 대한 이름 지정 규칙을 따르는 경우, 일반 이름 *ClusterName.** 은 다른 전송 큐에 다른 클러스터에 대한 메시지를 필터링합니다. 전송 큐를 수동으로 정의하고 각 전송 큐에서 CLCHNAME을 *ClusterName.**로 설정해야 합니다.

클러스터 송신자 채널에 대한 클러스터 전송 큐의 연관을 변경하면 즉시 적용되지 않습니다. 클러스터 송신자 채널이 서비스를 제공하고 있는 현재 연관된 전송 큐가 클러스터 송신자 채널에서 전송 중인 메시지를 포함할 수 있습니다. 현재 연관된 전송 큐에 클러스터 송신자 채널이 처리 중인 메시지가 없는 경우에만 큐 관리자가 다른 전송 큐에 대한 클러스터 송신자 채널의 연관을 변경할 수 있습니다. 이러한 상황은 클러스터 송신자 채널이 처리할 메시지가 전송 큐에 남아 있지 않은 경우 또는 메시지 처리가 일시중단되어 클러스터 송신자 채널에 "인플라이트" 메시지가 없는 경우에 발생할 수 있습니다. 이러한 상황이 발생하면 클러스터 송신자 채널의 미처리된 메시지가 새로 연관된 전송 큐에 전송되고 클러스터 송신자 채널의 연관이 변경됩니다.

클러스터 전송 큐로 해석되는 리모트 큐 정의를 작성할 수 있습니다. 이 정의에서 큐 관리자 QMX는 로컬 큐 관리자와 같은 클러스터에 있으며 전송 큐 QMX가 없습니다.

```
DEFINE QREMOTE(A) RNAME(B) RQMNAME(QMX)
```

큐 이름 해석 중에 클러스터 전송 큐는 기본 전송 큐보다 우선합니다. A에 배치되는 메시지는 클러스터 전송 큐에 저장된 후 QMX의 리모트 큐 B로 송신됩니다.

큐 관리자는 클러스터에 속하지 않는 다른 큐 관리자와 통신할 수도 있습니다. 분산 큐잉 환경에서와 동일한 방식으로 다른 큐 관리자에 대한 채널 및 전송 큐를 정의해야 합니다.

참고: 애플리케이션은 클러스터 전송 큐로 해석되는 큐에 기록되어야 하고 클러스터 전송 큐에 직접적으로 기록되어서는 안됩니다.

리모트 큐의 자동 정의

클러스터에 있는 큐 관리자는 클러스터의 리모트 큐에 대한 리모트 큐 정의가 필요하지 않습니다. 클러스터 큐 관리자는 전체 저장소에서 리모트 큐의 위치를 찾습니다. 메시지에 라우팅 정보를 추가하고 이 메시지를 클러스터 전송 큐에 넣습니다. IBM MQ는 메시지가 송신될 수 있도록 리모트 큐 정의에 해당하는 정의를 자동으로 작성합니다.

자동으로 작성된 리모트 큐 정의는 대체하거나 삭제할 수 없습니다. 그러나 DISPLAY QUEUE **runmqsc** 명령을 CLUSINFO 속성과 함께 사용하여 리모트 큐 관리자의 클러스터 큐를 포함한 모든 클러스터 큐 및 큐 관리자의 모든 로컬 큐를 볼 수 있습니다. 예를 들면, 다음과 같습니다.

```
DISPLAY QUEUE(*) CLUSINFO
```

관련 정보
클러스터 큐
ClusterChannelName(MQCHAR20)

자동 정의된 클러스터 송신자 채널에 대한 작업

초기 CLUSSDR 및 CLUSRCVR 정의를 작성하여 클러스터에 큐 관리자를 도입한 후, IBM MQ은(는) 클러스터의 다른 큐 관리자로 메시지를 이동하는 데 필요한 경우 다른 클러스터 송신자 채널 정의를 자동으로 작성합니다. 자동 정의된 클러스터 송신자 채널에 대한 정보를 볼 수 있지만 수정할 수는 없습니다. 그 작동을 수정하기 위해 채널 자동 정의 엑시트를 사용할 수 있습니다.

시작하기 전에

자동 정의 채널에 대한 소개는 자동 정의된 클러스터 송신자 채널의 내용을 참조하십시오.

이 태스크 정보

자동 정의된 클러스터 송신자 채널이 필요한 경우 클러스터에 의해 생성되고 정상 연결 해제 간격 규칙을 사용하여 종료될 때까지 활성으로 유지됩니다.

클러스터 송신자 채널(CLUSSDR)은 애플리케이션 메시지와 내부 클러스터 관리 메시지를 이동하기 위해 자동으로 정의될 수 있습니다. 예를 들어, 발행/구독 클러스터(클러스터된 토픽이 정의된 항목)에서 부분 저장소 사이에 채널을 정의하여 '프록시 구독' 상태의 교환을 허용할 수 있습니다. 확장된 기간의 자동 정의된 CLUSSDR에 필요하지 않은 경우(비활성) 클러스터 정보의 부분 저장소의 캐시에서 제거되며 더 이상 해당 큐 관리자에 표시되지 않습니다

Multi 멀티플랫폼에서 OAM(Object Authority Manager)은 자동 정의된 클러스터 송신자 채널의 존재를 인식하지 못합니다. 자동 정의된 클러스터 송신자 채널에서 **start**, **stop**, **ping**, **reset** 또는 **resolve** 명령을 발행하면 OAM은 일치하는 클러스터 수신자 채널에서 사용자에게 같은 조치를 수행하기 위한 권한이 부여되는지 여부를 확인하기 위해 검사합니다.

z/OS z/OS에서 다른 채널과 같은 방식으로 자동 정의된 클러스터 송신자 채널을 보안 설정할 수 있습니다.

프로시저

- 지정된 클러스터 큐 관리자의 자동 정의된 채널에 대한 정보를 표시하십시오.

  DISPLAY CHANNEL **runmqsc** 명령을 사용해서는 자동으로 정의된 채널을 볼 수 없습니다. 자동 정의된 채널을 보려면 다음 명령을 사용하십시오.

  ```
  DISPLAY CLUSQMGR(qMgrName)
  ```

- 지정된 CLUSRCVR에 대한 자동 정의된 채널의 상태를 표시하십시오.

  사용자가 작성한 CLUSRCVR 채널 정의에 해당하는 자동 정의된 CLUSSDR 채널의 상태를 표시하려면 다음 명령을 사용하십시오.

  ```
  DISPLAY CHSTATUS(channelname)
  ```

- 자동 정의된 채널의 작동을 수정하려면 채널 자동 정의된 엑시트를 사용하십시오.

  사용자 엑시트 프로그램을 작성하여 클러스터 송신자 채널 또는 클러스터 수신자 채널을 사용자 정의하려는 경우 IBM MQ 채널 자동 정의 엑시트를 사용할 수 있습니다. 예를 들어, 클러스터 환경에서 채널 자동 정의 엑시트를 사용하여 다음과 같이 수정할 수 있습니다.

  - 통신 정의 즉, SNA LU6.2 이름을 맞춤화하십시오.
  - 다른 엑시트(예: 보안 엑시트)를 추가하거나 제거하십시오.
  - 채널 엑시트의 이름을 변경하십시오.

    CLUSSDR 채널 엑시트의 이름은 CLUSRCVR 채널 정의에서 자동 생성되므로 특히 채널의 양 끝이 다른 플랫폼에 있는 경우 사용자의 필요에 적합하지 않을 수 있습니다.

    엑시트 이름의 형식은 플랫폼에 따라 다릅니다. 예를 들면, 다음과 같습니다.

    - z/OS z/OS 플랫폼에서 SCYEXIT(*security exit name*) 매개변수의 형식은 `SCYEXIT('SECEXIT')`입니다.
    - Windows Windows 플랫폼에서 SCYEXIT(*security exit name*) 매개변수의 형식은 `SCYEXIT('`*drive*`:\`*path*`\`*library* `(secexit)')`입니다.

    참고: z/OS 채널 자동 정의 엑시트가 없는 경우 z/OS 큐 관리자는 채널의 다른 측에 있는 CLUSRCVR 채널 정의에서 CLUSSDR 채널 엑시트 이름을 도출합니다. 비z/OS 이름에서 z/OS 엑시트 이름을 도출하기 위해 다음 알고리즘이 사용됩니다.

    - 멀티플랫폼에서 엑시트 이름의 일반적인 양식은 *path/library(function)*입니다.

- *function*이 있는 경우 그 중 최대 8자가 사용됩니다.
- 그렇지 않으면 *library*의 최대 8자가 사용됩니다.

예를 들면, 다음과 같습니다.

- /var/mqm/exits/myExit.so(MsgExit)는 MSGEXIT로 변환됩니다.
- /var/mqm/exits/myExit는 MYEXIT로 변환됩니다.
- /var/mqm/exits/myExit.so(ExitLongName)는 EXITLONG로 변환됩니다.

• IBM WebSphere MQ 7 이전 큐 관리자의 경우 **PROPCTL** 속성의 값을 NONE으로 설정하십시오.

각 자동 정의된 클러스터 송신자 채널은 해당하는 클러스터 수신자 채널을 기반으로 합니다. IBM MQ 버전 7 이전 클러스터 수신자 채널에는 **PROPCTL** 속성이 없기 때문에 자동 정의된 클러스터 송신자 채널에서 이 속성은 COMPAT로 설정됩니다.

클러스터가 IBM WebSphere MQ 7 이상 큐 관리자에서 이전 버전의 IBM MQ로 이동하는 메시지에서 RFH2와 같은 애플리케이션 헤더를 제거하기 위해 **PROPCTL**을 사용해야 하는 경우 **PROPCTL**의 값을 NONE으로 설정하는 채널 자동 정의 엑시트를 작성해야 합니다.

• 채널 속성 LOCLADDR을 사용하여 주소 지정 측면을 제어하십시오.

  • 아웃바운드(TCP) 채널이 특정 IP 주소, 포트 또는 포트 범위를 사용할 수 있도록 하려면 채널 속성 LOCLADDR을 사용하십시오. 이는 둘 이상의 네트워크 카드가 있고 아웃바운드 통신을 위해 채널이 특정 네트워크 카드를 사용하게 하려는 경우에 유용합니다.
  • CLUSSDR 채널에서 가상 IP 주소를 지정하려면 수동으로 정의된 CLUSSDR에서 LOCLADDR의 IP 주소를 사용하십시오. 포트 범위를 지정하려면 CLUSRCVR의 포트 범위를 사용하십시오.
  • 클러스터가 특정 IP 주소를 바인딩할 아웃바운드 통신 채널을 가져오기 위해 LOCLADDR을 사용해야 하는 경우 자동으로 정의된 CLUSSDR 채널 중 어느 하나로 LOCLADDR 값을 강제 실행하는 채널 자동 정의 엑시트를 작성할 수 있습니다. 이를 수동으로 정의된 CLUSSDR 채널에서도 지정해야 합니다.
  • 클러스터 내 모든 큐 관리자가 해당 모든 아웃바운드 통신을 위해 특정 포트 또는 포트 범위를 사용하게 하려는 경우 CLUSRCVR채널의 LOCLADDR에 포트 번호 또는 포트 범위를 넣으십시오.

참고: 모든 큐 관리자가 같은 서버에 있지 않은 경우 CLUSRCVR 채널의 LOCLADDR 필드에 IP 주소를 넣지 마십시오. LOCLADDR IP 주소는 CLUSRCVR 채널을 사용하여 연결된 모든 큐 관리자의 자동 정의된 CLUSSDR 채널로 전파됩니다.

Multi 멀티플랫폼에서는 로컬 주소가 정의되지 않은 모든 송신자 채널에 사용되는 기본 로컬 주소 값을 설정할 수 있습니다. 큐 관리자를 시작하기 전에 MQ_LCLADDR 환경 변수를 설정하여 기본값이 정의됩니다. 값의 형식은 MQSC 속성 LOCLADDR의 형식과 일치합니다.

관련 정보
로컬 주소(LOCLADDR)

기본 클러스터 오브젝트에 대한 작업

다른 채널 정의와 동일한 방법으로, MQSC 또는 PCF 명령을 실행하여 기본 채널 정의를 변경할 수 있습니다. SYSTEM.CLUSTER.HISTORY.QUEUE를 제외하고, 기본 큐 정의는 변경하지 마십시오.

이러한 오브젝트의 전체 목록은 기본 클러스터 오브젝트의 내용을 참조하십시오. 다음 목록은 변경할 수 있는 오브젝트만 포함합니다.

SYSTEM.CLUSTER.HISTORY.QUEUE

클러스터의 각 큐 관리자에는 SYSTEM.CLUSTER.HISTORY.QUEUE라고 하는 로컬 큐가 있습니다. SYSTEM.CLUSTER.HISTORY.QUEUE 는 서비스 목적으로 클러스터 상태 정보의 히스토리를 저장하는 데 사용됩니다.

기본 오브젝트 설정에서 SYSTEM.CLUSTER.HISTORY.QUEUE은(는) PUT(ENABLED)으로 설정됩니다. 히스토리 콜렉션을 억제하려면 설정을 PUT(DISABLED)으로 변경하십시오.

SYSTEM.CLUSTER.TRANSMIT.QUEUE

각 큐 관리자에는 SYSTEM.CLUSTER.TRANSMIT.QUEUE라고 하는 로컬 큐에 대한 정의가 있습니다. SYSTEM.CLUSTER.TRANSMIT.QUEUE는 클러스터 내에 있는 모든 큐 및 큐 관리자에 대한 모든 메시지의 기본 전송 큐입니다. 큐 관리자 속성 SYSTEM.CLUSTER.TRANSMIT.*ChannelName*을(를) 제외하고 DEFXMITQ z/OS 을(를) 변경하여 각 클러스터 송신자 채널에 대한 기본 전송 큐를 z/OS(으)로 변경할 수 있습니다. SYSTEM.CLUSTER.TRANSMIT.QUEUE를 삭제할 수 없습니다. 사용되는 기본 전송 큐가 SYSTEM.CLUSTER.TRANSMIT.QUEUE 또는 SYSTEM.CLUSTER.TRANSMIT.*ChannelName*인지에 대한 권한 검사를 정의하는 데도 사용됩니다.

관련 정보

기본 클러스터 오브젝트

클러스터 전송 큐 및 클러스터 송신자 채널에 대한 작업

클러스터된 큐 관리자 간 메시지는 클러스터 전송 큐에 저장되고 클러스터 송신자 채널에서 전달됩니다. 임의의 시점에서 클러스터 송신자 채널은 한 개의 전송 큐와 연관됩니다. 채널의 구성을 변경하는 경우 다음에 시작할 때 다른 전송 큐로 전환할 수도 있습니다. 이 전환의 처리는 자동화되고 트랜잭션 가능합니다.

다음 MQSC 명령을 실행하여 클러스터 송신자 채널과 연관되는 전송 큐를 표시하십시오.

```
DISPLAY CHSTATUS(*) WHERE(CHLTYPE EQ CLUSSDR)
```

```
AMQ8417: Display Channel Status details.
CHANNEL(TO.QM2)           CHLTYPE(CLUSSDR)
CONNAME(9.146.163.190(1416))    CURRENT
RQMNAME(QM2)             STATUS(STOPPED)
SUBSTATE( )              XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

중지된 클러스터 송신자 채널의 저장된 채널 상태에 표시되는 전송 큐는 채널이 다시 시작되면 변경될 수 있습니다. 222 페이지의 『클러스터 송신자 채널에 따라 기본 전송 큐 선택』에서는 기본 전송 큐를 선택하는 프로세스를 설명하고 222 페이지의 『클러스터 송신자 채널에 따라 수동으로 정의된 전송 큐 선택』에서는 수동으로 정의된 전송 큐를 선택하는 프로세스를 설명합니다.

클러스터 송신자 채널이 시작되면 전송 큐와의 연관을 다시 검사합니다. 전송 큐의 구성 또는 큐 관리자 기본값이 변경되면 채널을 다른 전송 큐와 다시 연관시킬 수 있습니다. 구성을 변경함에 따라 채널이 다른 전송 큐로 재시작되면 새로 연관된 전송 큐에 대한 메시지 전송 프로세스가 발생합니다. 223 페이지의 『클러스터 송신자 채널을 다른 전송 큐로 전환하는 프로세스의 작동 방식』에서는 한 전송 큐에서 다른 전송 큐로 클러스터 송신자 채널을 전송하는 프로세스를 설명합니다.

클러스터 송신자 채널의 작동은 송신자 채널 및 서버 채널에 대해 다릅니다. 채널 속성 **XMITQ**이 대체될 때까지 같은 전송 큐와 연관 상태로 남습니다. 송신자 또는 서버 채널에서 전송 큐 속성을 대체하고 재시작하는 경우 이전 전송 큐에서 새 전송 큐로 메시지가 전송되지 않습니다.

클러스터 송신자 채널과 송신자 또는 서버 채널 간 또 다른 차이점은 다중 클러스터 송신자 채널은 클러스터 전송 큐를 열 수 있지만 한 송신자 또는 서버 채널만 정상 전송 큐를 열 수 있다는 것입니다. IBM WebSphere MQ 7.5까지는 클러스터 연결이 단일 클러스터 전송 큐 SYSTEM.CLUSTER.TRANSMIT.QUEUE를 공유했습니다. IBM WebSphere MQ 7.5 이상부터는 전송 큐를 공유하지 않는 클러스터 송신자 채널 옵션이 있습니다. 독점성은 강제 실행되지 않습니다. 이는 구성의 결과입니다. 다른 애플리케이션 사이를 플로우하는 메시지와 전송 큐나 채널을 공유하지 않도록 클러스터에서 메시지가 이동하는 경로를 구성할 수 있습니다. 클러스터링: 클러스터 전송 큐를 구성하는 방법 계획 및 271 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가』의 내용을 참조하십시오.

z/OS에서 클러스터 송신자 채널이 SYSTEM.CLUSTER.TRANSMIT.QUEUE 외의 전송 큐를 사용하도록 구성하려면 CSQ6SYSP 매크로에서 조작 모드(OPMODE) 시스템 매개변수를 사용하여 버전 8 새로운 가능을 사용으로 설정해야 합니다.

클러스터 송신자 채널에 따라 기본 전송 큐 선택

클러스터 전송 큐는 SYSTEM.CLUSTER.TRANSMIT으로 시작되는 이름의 시스템 기본 큐이거나 수동으로 정의된 큐입니다. 클러스터 송신자 채널은 기본 클러스터 전송 큐 메커니즘 또는 수동 구성 방식 중 하나로 클러스터 전송 큐와 연관됩니다.

기본 클러스터 전송 큐는 큐 관리자 속성 **DEFCLXQ**로 설정됩니다. 그 값은 SCTQ 또는 CHANNEL입니다. 새로운 큐 관리자 및 마이그레이션된 큐 관리자는 SCTQ로 설정됩니다. 이 값을 CHANNEL로 대체할 수 있습니다.

SCTQ가 설정되는 경우 기본 클러스터 전송 큐는 SYSTEM.CLUSTER.TRANSMIT.QUEUE입니다. 모든 클러스터 송신자 채널은 이 큐를 열 수 있습니다. 큐를 여는 클러스터 송신자 채널은 수동으로 정의된 클러스터 전송 큐와 연관되지 않은 채널입니다.

CHANNEL이 설정되면 큐 관리자가 모든 클러스터 송신자 채널에 대해 개별적으로 영구적 동적 전송 큐를 작성할 수 있습니다. 각 큐의 이름은 SYSTEM.CLUSTER.TRANSMIT. *ChannelName* 이고 모델 큐 SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE에서 작성됩니다. 수동으로 정의된 클러스터 전송 큐와 연관되지 않은 각 클러스터 송신자 채널은 영구적 동적 클러스터 전송 큐와 연관됩니다. 이 큐는 이 클러스터 송신자 채널이 서비스를 제공하는 클러스터 목적지에 대한 별도의 클러스터 전송 큐가 필요하고 큐가 존재하지 않는 경우에 큐 관리자가 작성합니다.

일부 클러스터 목적지는 수동으로 정의된 전송 큐와 연관된 클러스터 송신자 채널이 서비스를 제공할 수 있고 다른 목적지는 기본 큐 또는 여러 큐가 서비스를 제공할 수 있습니다. 클러스터 송신자 채널과 전송 큐와의 연관에서 수동으로 정의된 전송 큐는 항상 기본 전송 큐보다 우선합니다.

클러스터 전송 큐의 우선순위는 222 페이지의 그림 38에 설명됩니다. 수동으로 정의된 클러스터 전송 큐와 연관되지 않은 유일한 클러스터 송신자 채널은 CS.QM1입니다. 전송 큐의 **CLCHNAME** 속성에 있는 어떤 채널 이름도 CS.QM1과 일치하지 않기 때문에 이 채널은 수동으로 정의된 전송 큐와 연관되지 않습니다.

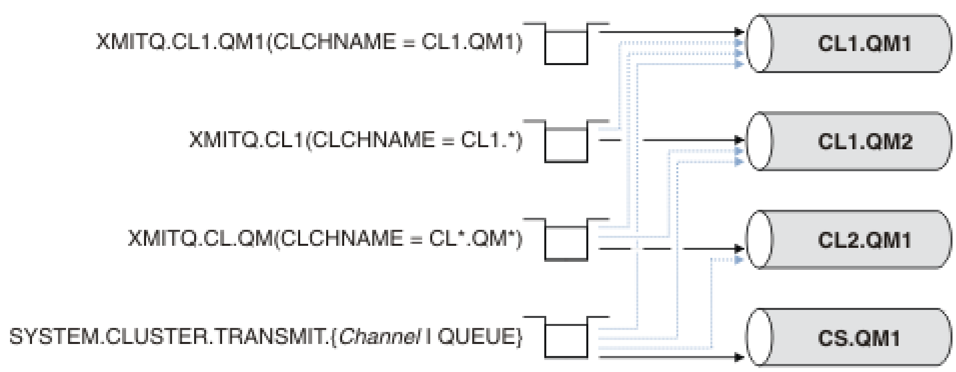


그림 *38.* 전송 큐 / 클러스터 송신자 채널 우선순위

클러스터 송신자 채널에 따라 수동으로 정의된 전송 큐 선택

수동으로 정의된 큐에는 **USAGE** 속성이 XMITQ로 설정되고 클러스터 채널 이름 속성 **CLCHNAME**이 특정 또는 일반 채널 이름으로 설정된 전송 큐가 있습니다.

CLCHNAME 큐 속성에 있는 이름이 클러스터 송신자 채널 이름과 일치하는 경우 이 채널은 큐와 연관됩니다. 이름에 와일드카드가 포함되지 않은 경우에는 정확히 일치하는 이름이고 이름에 와일드 카드가 포함된 경우에는 가장 일치하는 이름입니다.

다중 전송 큐에서 **CLCHNAME** 정의가 같은 클러스터 송신자 채널과 일치하는 경우 정의가 중첩된다고 합니다. 모호함을 해결하기 위해 일치 간 우선순위가 있습니다. 정확한 일치는 항상 우선순위를 갖습니다. 222 페이지의 그림 38에서는 전송 큐와 클러스터 송신자 채널 간 연관을 보여줍니다. 검은색 화살표는 실제 연관을 표시하고 회색 화살표는 잠재적 연관을 표시합니다. 222 페이지의 그림 38에서 전송 큐의 우선순위는 다음과 같습니다.

XMITQ.CL1.QM1
: 전송 큐 XMITQ.CL1.QM1의 **CLCHNAME** 속성은 CL1.QM1으로 설정됩니다. **CLCHNAME** 속성 CL1.QM1의 정의에는 와일드카드가 없으며 와일드카드와 일치하는 다른 전송 큐에서 정의된 다른 CLCHNAME 속성보다 우선합니다. 큐 관리자는 XMITQ.CL1.QM1 전송 큐에서 CL1.QM1 클러스터 송신자 채널에 의해 전송될 클

러스터 메시지를 저장합니다. 유일한 예외는 다중 전송 큐에서 그 **CLCHNAME** 속성이 CL1.QM1으로 설정되어 있는 경우입니다. 이러한 경우에 큐 관리자는 그러한 큐 중 임의의 큐에 CL1.QM1 클러스터 송신자 채널에 대한 메시지를 저장합니다. 채널이 시작될 때 임의로 큐를 선택합니다. 채널이 다시 시작될 때 다른 큐를 선택할 수도 있습니다.

XMITQ.CL1
: 전송 큐 XMITQ.CL1의 **CLCHNAME** 속성은 CL1.*으로 설정됩니다. **CLCHNAME** 속성 CL1.*의 정의에는 후미 문자 와일드카드가 있으며 이 정의는 CL1.으로 시작하는 클러스터 송신자 채널의 이름과 일치합니다. 큐 관리자는 큐 XMITQ.CL1.QM1과 같이 보다 구체적인 일치가 있는 전송 큐가 없는 경우 전송 큐 XMITQ.CL1에서 이름이 CL1.으로 시작하는 클러스터 송신자 채널에 의해 전송될 클러스터 메시지를 저장합니다. 한 개의 후미 문자 와일드카드가 있는 정의는 와일드카드가 없는 정의에 비해 덜 구체적이고 와일드카드가 여러 개 있거나 와일드카드 뒤에 추가 후미 문자가 있는 정의에 비해 더 구체적입니다.

XMITQ.CL.QM
: XMITQ.CL.QM은 그 **CLCHNAME** 속성이 CL*.QM*로 설정된 전송 큐의 이름입니다. CL*.QM*의 정의에는 두 개의 와일드카드가 있으며 이 정의는 CL.로 시작하고 QM을 포함하거나 QM으로 끝나는 클러스터 송신자 채널의 이름과 일치합니다. 이 일치는 한 개의 와일드카드 일치보다 덜 구체적입니다.

SYSTEM.CLUSTER.TRANSMIT. ***channelName*|QUEUE**
: 큐 관리자가 사용할 클러스터 송신자 채널의 이름과 일치하는 **CLCHNAME** 속성을 가진 전송 큐가 없는 경우 큐 관리자는 기본 클러스터 전송 큐를 사용합니다. 기본 클러스터 전송 큐는 단일 시스템 클러스터 전송 큐인 SYSTEM.CLUSTER.TRANSMIT.QUEUE이거나 큐 관리자가 특정 클러스터 송신자 채널인 SYSTEM.CLUSTER.TRANSMIT. *channelName*에 대해 작성한 시스템 클러스터 전송 큐입니다. 어떤 큐를 기본 큐로 할 것인지는 큐 관리자 **DEFXMITQ** 속성이 설정에 따라 다릅니다.

팁: 중첩되는 정의에 대한 분명한 필요성이 없다면 이해하기 어려운 복잡한 구성으로 이어질 수 있으므로 중첩되는 정의는 피하십시오.

클러스터 송신자 채널을 다른 전송 큐로 전환하는 프로세스의 작동 방식

클러스터 송신자 채널과 클러스터 전송 큐의 연관을 변경하려면 전송 큐의 **CLCHNAME** 매개변수 또는 큐 관리자 매개변수 **DEFCLXQ**를 언제든지 변경하십시오. 변경 직후에는 변화가 없습니다. 채널이 시작되는 경우에만 변경이 발생합니다. 채널이 시작되면 같은 전송 큐에서 계속 메시지를 전달하는지 여부를 검사합니다. 세 가지 유형의 변경이 클러스터 송신자 채널과 전송 큐와의 연관을 대체합니다.

1. 클러스터 송신자 채널이 현재 연관되어 있는 전송 큐의 **CLCHNAME** 매개변수를 덜 구체적으로 또는 공백으로 재정의하거나 채널이 중지될 때 클러스터 전송 큐를 삭제합니다.

   일부 다른 클러스터 전송 큐가 이제 클러스터 이름에 대해 더 나은 일치일 수 있습니다. 또는 클러스터 송신자 채널의 이름과 일치하는 다른 전송 큐가 없는 경우 연관은 기본 전송 큐로 되돌아가야 합니다.
2. 다른 클러스터 전송 큐의 **CLCHNAME** 매개변수를 재정의하거나 클러스터 전송 큐를 추가합니다.

   다른 전송 큐의 **CLCHNAME** 매개변수가 이제 클러스터 송신자 채널에 대해 클러스터 송신자 채널이 현재 연관된 전송 큐에 비해 더 나은 일치일 수 있습니다. 클러스터 송신자 채널이 현재 기본 클러스터 전송 큐와 연관되어 있는 경우 이 채널은 수동으로 정의된 클러스터 전송 큐와 연관이 될 수 있습니다.
3. 클러스터 송신자 채널이 현재 기본 클러스터 전송 큐와 연관되어 있는 경우 **DEFCLXQ** 큐 관리자 매개변수를 변경합니다.

클러스터 송신자 채널의 연관이 변경되는 경우 채널이 시작되면 이 연관이 새 전송 큐에 대한 연관으로 전환됩니다. 전환 중에 메시지가 손실되지 않아야 합니다. 메시지는 채널이 리모트 큐 관리자로 메시지를 전송하는 순서대로 새 전송 큐에 전송됩니다.

알아두기: 클러스터의 메시지를 전달할 때와 마찬가지로, 순서대로 전달되어야 하는 메시지가 순서대로 전달되도록 메시지를 그룹으로 묶어야 합니다. 드물지만 클러스터 내에서 메시지 순서가 뒤바뀔 수 있습니다.

전환 프로세스는 다음 트랜잭션 단계를 거칩니다. 전환 프로세스가 방해되는 경우 채널이 재시작될 때 현재 트랜잭션 단계가 재개됩니다.

단계 1 - 원래 전송 큐의 메시지 처리
: 클러스터 송신자 채널이 다른 클러스터 송신자 채널과 공유할 수도 있는 새 전송 큐와 연관됩니다. 클러스터 송신자 채널의 메시지는 계속해서 원래 전송 큐에 배치됩니다. 과도기적 전환 프로세스는 메시지를 원래 전

송 큐에서 새 전송 큐로 전송합니다. 클러스터 송신자 채널은 새 전송 큐에서 클러스터 수신자 채널로 메시지를 전달합니다. 채널 상태가 클러스터 송신자 채널이 여전히 이전 전송 큐와 연관되어 있음을 표시합니다.

전환 프로세스가 계속해서 새로 도착한 메시지도 전송합니다. 이 단계는 전환 프로세스에 의해 전달될 남아 있는 메시지의 수가 0에 도달할 때까지 계속됩니다. 메시지 수가 0에 도달하면 프로시저가 단계 2로 이동합니다.

단계 1 중에 채널의 디스크 활동이 늘어납니다. 지속 메시지가 첫 번째 전송 큐에서 두 번째 전송 큐로 커미트됩니다. 이 디스크 활동은 일반적인 메시지 전송의 일부로 전송 큐에서 메시지가 배치되고 제거될 때 커미트되는 메시지에 추가됩니다. 전환이 최대한 빨리 이루어질 수 있도록 전환 프로세스 중 메시지가 도착하지 않는 것이 이상적입니다. 메시지가 도착하면 전환 프로세스에 의해 처리됩니다.

단계 2 - 새 전송 큐에서 메시지 처리

클러스터 송신자 채널에 대한 원래 전송 큐에 남아 있는 메시지가 없게 되는 즉시 새 메시지가 새 전송 큐에 바로 배치됩니다. 채널 상태는 클러스터 송신자 채널이 새 전송 큐와 연관되어 있음을 표시합니다. 다음 메시지는 큐 관리자 오류 로그에 기록됩니다." AMQ7341 `The transmission queue for channel` *`ChannelName`* `is` *`QueueName`* ."

다중 클러스터 전송 큐 및 클러스터 전송 큐 속성

단일 클러스터 전송 큐 또는 다중 큐에 메시지를 저장하는 여러 큐 관리자에게 클러스터 메시지를 전달하도록 선택할 수 있습니다. 하나의 큐를 사용하면 설정하고 조회할 클러스터 전송 큐 속성이 한 세트 있고 다중 큐를 사용하면 여러 세트가 있습니다. 일부 속성에서는 다중 세트가 있을 경우 유리합니다. 예를 들어, 큐 용량을 조회하면 모든 채널에서가 아닌 한 채널 또한 한 채널 세트가 전달할 대기 중 메시지의 수를 알려줍니다. 다른 속성에서는 다중 세트가 있는 것이 불리합니다. 예를 들면, 모든 클러스터 전송 큐에 대해 같은 액세스 권한을 구성하는 것을 원치 않는 경우입니다. 이러한 이유로 액세스 권한은 항상 특정 클러스터 전송 큐의 프로파일에 대해서가 아닌 `SYSTEM.CLUSTER.TRANSMIT.QUEUE`의 프로파일에 대해 검사됩니다. 보다 세분화된 보안 검사를 적용하려는 경우 액세스 제어 및 다중 클러스터 전송 큐를 참조하십시오.

다중 클러스터 송신자 채널 및 다중 전송 큐

큐 관리자는 클러스터 송신자 채널에서 메시지를 전달하기 전에 메시지를 클러스터 전송 큐에 저장합니다. 큐 관리자는 메시지의 목적지에 연결되는 클러스터 송신자 채널을 선택합니다. 모두 같은 목적지에 연결되는 클러스터 송신자 채널을 선택할 수 있습니다. 목적지는 같은 물리적 큐일 수 있으며 다중 클러스터 송신자 채널에 의해 단일 큐 관리자에 연결됩니다. 목적지는 같은 클러스터 내 다른 큐 관리자에서 호스팅되며 같은 큐 이름을 가지는 다수의 물리적 큐일 수도 있습니다. 목적지에 연결된 클러스터 송신자 채널을 선택할 수 있는 경우 워크로드 밸런싱 알고리즘이 하나를 선택합니다. 이 선택은 다수의 요인에 따라 달라집니다. 클러스터 워크로드 관리 알고리즘을 참조하십시오.

225 페이지의 그림 39에서 `CL1.QM1`, `CL1.QM2` 및 `CS.QM1`이 같은 목적지로 이어질 수 있는 모든 채널입니다. 예를 들어, `Q1`을 `QM1` 및 `QM2`의 `CL1`에서 정의하면 `CL1.QM1` 및 `CL1.QM2` 모두가 다른 두 큐 관리자에 있는 동일 목적지, `Q1`으로의 라우트를 제공합니다. 채널 `CS.QM1`도 `CL1`에 있는 경우 이 채널도 `Q1`에 대한 메시지가 선택할 수 있는 채널입니다. `CS.QM1`의 클러스터 멤버십은 클러스터 이름 목록에서 정의될 수 있으며, 채널 이름이 그 구성에 클러스터 이름을 포함하지 않는 이유입니다. 워크로드 밸런싱 매개변수 및 송신 애플리케이션에 따라 `Q1`에 대한 메시지가 전송 큐 `XMITQ.CL1.QM1`, `XMITQ.CL1` 및 `SYSTEM.CLUSTER.TRANSMIT.CS.QM1` 각각에 배치될 수 있습니다.

동일 목적지에 대한 메시지가 다른 목적지에 대한 메시지와 큐 또는 채널을 공유하지 않도록 메시지 트래픽을 분배하려는 경우 먼저 트래픽을 여러 클러스터 송신자 채널로 나누는 방법을 고려하고 특정 채널에 대한 메시지를 여러 전송 큐로 분리하는 방법에 대해 고려해야 합니다. 같은 큐 관리자의 같은 클러스터에 있는 클러스터 큐는 일반적으로 같은 클러스터 채널을 공유합니다. 다중 클러스터 전송 큐를 단독으로 정의하는 것은 클러스터 메시지 트래픽을 여러 큐로 분리하는 데 충분하지 않습니다. 여러 목적지 큐에 대한 메시지를 여러 채널로 분리하지 않으면 메시지가 같은 클러스터 전송 큐를 공유합니다.

메시지가 선택하는 채널의 직접적인 분리 방법은 다중 클러스터를 작성하는 것입니다. 각 클러스터에 있는 큐 관리자에서 한 개의 클러스터 큐만 정의하십시오. 그런 다음 각 클러스터/큐 관리자 조합에 대해 다른 클러스터 수신자 채널을 정의하면 각 클러스터 큐에 대한 메시지가 다른 클러스터 큐에 대한 메시지와 클러스터 채널을 공유하지 않습니다. 클러스터 채널에 대해 별도의 전송 큐를 정의하면 송신 큐 관리자는 각 전송 큐에 한 개의 클러스터 큐에 대한 메시지만 저장합니다. 예를 들어, 두 개의 클러스터가 자원을 공유하지 않도록 하려는 경우 이러한

클러스터 자원을 같은 큐 관리자에 있는 다른 클러스터에 배치하거나 같은 클러스터에 있는 다른 큐 관리자에 배치할 수 있습니다.

클러스터 전송 큐의 선택은 워크로드 밸런싱 알고리즘에 영향을 주지 않습니다. 워크로드 밸런싱 알고리즘은 메시지를 전달할 클러스터 송신자 채널을 선택합니다. 이 알고리즘은 메시지를 해당 채널이 서비스를 제공하는 전송 큐에 배치합니다. 채널이 중지되는 경우와 같이 채널을 다시 선택하기 위해 워크로드 밸런싱 알고리즘이 호출되는 경우 이 알고리즘은 메시지를 전달할 다른 채널을 선택할 수 있습니다. 다른 채널을 선택하고 새 채널이 다른 클러스터 전송 큐의 메시지를 전달하는 경우 이 워크로드 밸런싱 알고리즘은 메시지를 다른 전송 큐로 전송합니다.

225 페이지의 그림 39에서 두 개의 클러스터 송신자 채널인 CS.QM1 및 CS.QM2가 기본 시스템 전송 큐와 연관됩니다. 워크로드 밸런싱 알고리즘이 메시지를 SYSTEM.CLUSTER.TRANSMIT.QUEUE 또는 다른 클러스터 전송 큐에 저장하면 메시지를 전달할 클러스터 송신자 채널의 이름이 메시지의 상관 ID에 저장됩니다. 각 채널은 채널 이름과 상관 ID가 일치하는 메시지만 전달합니다.

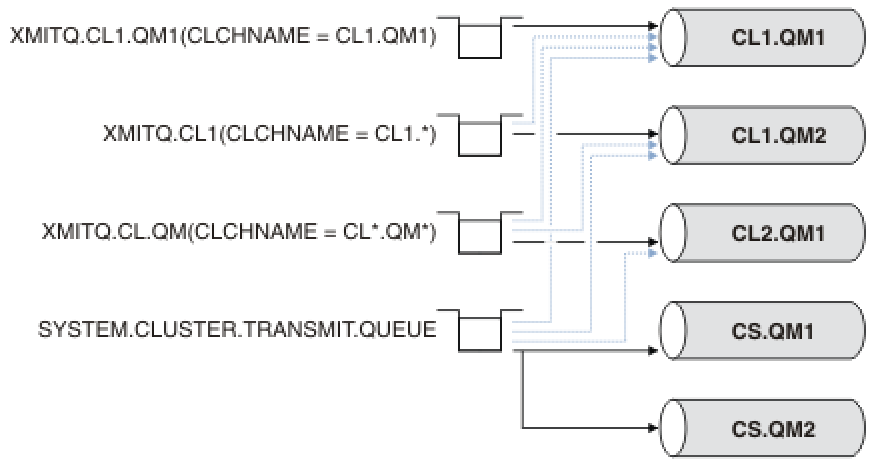


그림 *39.* 다중 클러스터 송신자 채널

CS.QM1이 중지되면 해당 클러스터 송신자 채널의 전송 큐에 있는 메시지를 조사합니다. 다른 채널이 전달할 수 있는 메시지는 워크로드 밸런싱 알고리즘에 의해 다시 처리됩니다. 해당 상관 ID는 대체 클러스터 송신자 채널 이름으로 재설정됩니다. 대체 클러스터 송신자 채널이 CS.QM2인 경우 메시지는 SYSTEM.CLUSTER.TRANSMIT.QUEUE에 남습니다. 대체 채널이 CL1.QM1인 경우 워크로드 밸런싱 알고리즘이 메시지를 XMITQ.CL1.QM1에 전송합니다. 클러스터 송신자 채널이 재시작되면 새 메시지 및 다른 클러스터 송신자 채널에 대해 플래그 지정되지 않은 메시지가 다시 이 채널에서 전송됩니다.

실행 중인 시스템에서 전송 큐와 클러스터 송신자 채널 간 연관을 변경할 수도 있습니다. 전송 큐에서 **CLCHNAME** 매개변수를 변경하거나 **DEFCLXQ** 큐 관리자 매개변수를 변경할 수 있습니다. 변경의 영향을 받는 채널이 재시작되면 이 채널은 전송 큐 전환 프로세스를 시작합니다. 223 페이지의 『클러스터 송신자 채널을 다른 전송 큐로 전환하는 프로세스의 작동 방식』의 내용을 참조하십시오.

채널이 재시작되면 전송 큐를 전환하는 프로세스가 시작됩니다. 채널이 중지되면 워크로드 재밸런싱 프로세스가 시작됩니다. 두 프로세스가 병렬로 실행될 수 있습니다.

단순한 예로는 클러스터 송신자 채널이 중지되어도 재밸런싱 프로세스가 큐의 메시지를 전송할 클러스터 송신자 채널을 변경하지 않는 경우입니다. 이 경우는 메시지를 올바른 목적지로 전달할 수 있는 다른 클러스터 송신자 채널이 없는 경우입니다. 메시지를 해당 목적지로 전송할 대체 클러스터 송신자 채널이 없는 경우 메시지는 클러스터 송신자 채널이 중지된 후 같은 클러스터 송신자 채널에 대해 플래그 지정된 상태로 남습니다. 채널이 시작될 때 전환이 보류 중이면 전환 프로세스는 같은 클러스터 송신자 채널이 메시지를 처리하는 다른 전송 큐로 메시지를 이동합니다.

이보다 복잡한 예로는 둘 이상의 클러스터 송신자 채널이 일부 메시지를 동일한 목적지로 처리할 수 있는 경우입니다. 클러스터 송신자 채널을 중지하고 재시작하여 전송 큐 전환을 트리거합니다. 대부분의 경우 채널을 재시작하는 시점에는 이미 워크로드 밸런싱 알고리즘이 원래 전송 큐에서 다른 클러스터 송신자 채널이 서비스를 제공하는 다른 전송 큐로 메시지를 이동한 상태입니다. 다른 클러스터 송신자 채널이 전달할 수 없는 메시지만 새 전송 큐로 전송되기 위해 남습니다. 일부 경우에 채널을 빠르게 재시작하면 워크로드 밸런싱 알고리즘에 의해 전송

될 수 있는 일부 메시지가 남습니다. 이 경우 남아 있는 일부 메시지는 워크로드 밸런싱 프로세스에 의해 전환되고 일부는 전송 큐 전환 프로세스에 의해 전환됩니다.

관련 개념

483 페이지의 『로그의 크기 계산』
큐 관리자가 필요로 하는 로그의 크기를 계산합니다.

관련 태스크

261 페이지의 『게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성』
태스크의 지시사항에 따라 게이트웨이 큐 관리자가 포함된 중첩 클러스터를 구성합니다. 한 애플리케이션에 대한 메시지를 클러스터 내 다른 애플리케이션에 대한 메시지로부터 격리하는 다음 예에 대한 시작점으로 이 클러스터를 사용합니다.

239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

268 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가』
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 추가 클러스터 전송 큐를 사용하여 클러스터의 단일 큐 관리자로 메시지 트래픽을 분리합니다.

271 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가』
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 솔루션은 추가 클러스터를 사용하여 메시지를 특정 클러스터 큐로 분리합니다.

관련 정보

클러스터 채널
클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리
클러스터링: 클러스터 전송 큐를 구성하는 방법을 계획

새 클러스터 설정

다음 지시사항에 따라 클러스터 예를 설정합니다. 개별 지시사항에서는 TCP/IP, LU 6.2 및 단일 전송 큐 또는 다중 전송 큐에서의 클러스터 설정을 설명합니다. 한 큐 관리자에서 다른 큐 관리자로 메시지를 송신하여 클러스터가 작동하는지 테스트합니다.

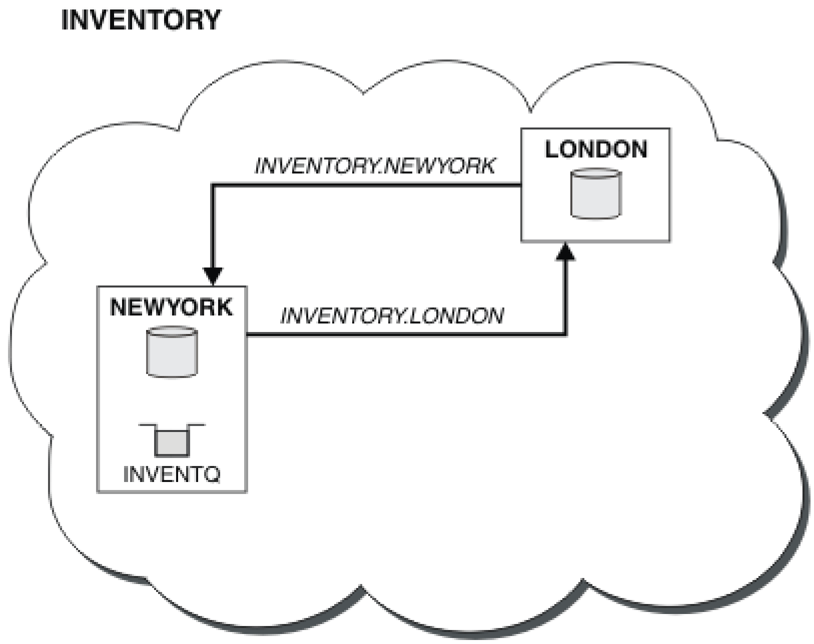
시작하기 전에

- 다음 지시사항을 따르는 대신 IBM MQ Explorer에서 제공하는 마법사 중 하나를 사용하여 이 태스크에서 작성되는 클러스터와 같은 클러스터를 작성할 수 있습니다. 큐 관리자 클러스터 폴더에서 마우스의 오른쪽 단추를 클릭한 다음 **새로 작성 > 큐 관리자 클러스터**를 클릭하고 마법사에서 제공되는 지시사항을 따르십시오.
- 클러스터를 설정하는 단계에 대한 이해를 돕기 위한 배경 정보는 216 페이지의 『클러스터 큐 정의』, 클러스터 채널 및 리스너를 참조하십시오.

이 태스크 정보

체인 저장소에 대한 새 IBM MQ 네트워크를 설정합니다. 이 저장소에는 런던 및 뉴욕에 하나씩 두 개의 분기가 있습니다. 각 저장소의 데이터 및 애플리케이션은 개별 큐 관리자를 실행하는 시스템에 의해 호스팅됩니다. 이 두 큐 관리자를 `LONDON` 및 `NEWYORK`이라고 합니다. 인벤토리 애플리케이션은 큐 관리자 `NEWYORK`에 연결된 뉴욕의 시스템에서 실행됩니다. 이 애플리케이션은 `INVENTQ` 큐에 메시지가 도착하면 구동되며 `NEWYORK`에 의해 호스팅됩니다. 두 큐 관리자 `LONDON` 및 `NEWYORK`은 `INVENTORY`라는 클러스터에서 링크되어 둘 다 `INVENTQ`에 메시지를 넣을 수 있습니다.

이 클러스터의 모양은 다음과 같습니다.



클러스터의 각 큐 관리자가 다른 클러스터 전송 큐를 사용하는 클러스터에 있는 다른 큐 관리자로 메시지를 송신하도록 구성할 수 있습니다.

클러스터를 설정하기 위한 지시사항은 전송 프로토콜, 전송 큐의 수 또는 플랫폼에 따라 다릅니다. 세 가지 조합을 선택할 수 있습니다. 확인 프로시저는 모든 조합에 대해 동일합니다.

INVENTORY 은 (는) 작은 클러스터입니다. 그러나 개념 검증용으로 유용합니다. 이 클러스터를 이해하는 데 중요한 것은 이 클러스터가 향후 개선을 위해 제공하는 범위입니다.

프로시저

- 227 페이지의 『TCP/IP를 사용하여 큐 관리자당 단일 전송 큐가 있는 클러스터 설정』
- 230 페이지의 『큐 관리자당 다중 전송 큐를 사용하여 TCP/IP에서 클러스터 설정』
- 233 페이지의 『z/OS에서 LU 6.2를 사용하여 클러스터 설정』
- 235 페이지의 『클러스터 확인』

관련 태스크

216 페이지의 『큐 관리자 클러스터 구성』
클러스터는 초기 구성 및 진행 중인 관리를 모두 단순화하는 방식으로 큐 관리자를 상호 연결하기 위한 메커니즘을 제공합니다. 클러스터 컴포넌트를 정의하고 클러스터를 작성 및 관리할 수 있습니다.

관련 정보

클러스터
클러스터링과 분산 큐잉의 비교
클러스터의 컴포넌트

*TCP/IP*를 사용하여 큐 관리자당 단일 전송 큐가 있는 클러스터 설정

이는 단순 클러스터에 대해 다른 구성을 설명하는 세 가지 주제 중 하나입니다.

시작하기 전에

작성 중인 클러스터에 대한 개요는 226 페이지의 『새 클러스터 설정』의 내용을 참조하십시오.

큐 관리자 속성 **DEFCLXQ**는 기본값 SCTQ로 두어야 합니다.

이 태스크 정보

다음 단계에 따라 전송 프로토콜 TCP/IP를 사용하여 멀티플랫폼에서 클러스터를 설정하십시오. z/OS On z/OS, you must follow the instructions in 676 페이지의 『z/OS에서 TCP 연결 정의』 to set up the TCP/IP connection , rather than defining the listeners in step 228 페이지의 『4』. 그렇지 않으면 단계는 z/OS의 경우와 동일하지만 오류 메시지는 큐 관리자 오류 로그가 아닌 콘솔로 기록됩니다.

프로시저

1. 클러스터의 조직 및 해당 이름을 결정하십시오.

   두 큐 관리자 LONDON 및 NEWYORK을 클러스터에 링크하기로 결정했습니다. 두 개의 큐 관리자만 있는 클러스터는 분산 큐잉을 사용하는 네트워크에 대해서는 약간의 이익만 제공하지만 분산 큐잉을 시작하기 좋은 방법이며 향후 확장을 위한 범위를 제공합니다. 저장소의 새 분기를 여는 경우 클러스터에 새 큐 관리자를 쉽게 추가할 수 있습니다. 새 큐 관리자를 추가해도 기존 네트워크에 지장을 주지 않습니다. 237 페이지의 『클러스터에 큐 관리자 추가』의 내용을 참조하십시오.

   잠시 동안 실행 중인 애플리케이션만 인벤토리 애플리케이션입니다. 클러스터 이름은 INVENTORY입니다.

2. 전체 저장소를 보유할 큐 관리자를 결정하십시오.

   모든 클러스터에서 전체 저장소를 보유할 최소 하나 이상의(가급적이면 둘) 큐 관리자를 지정해야 합니다. 이 예에는 둘 다 전체 저장소를 보유하는 두 큐 관리자인 LONDON 및 NEWYORK만 있습니다.

   a. 순서에 관계없이 나머지 단계를 수행할 수 있습니다.
   b. 단계에 따라 진행하는 경우 경고 메시지가 큐 관리자 로그에 기록될 수 있습니다. 이러한 메시지는 아직 추가하지 않는 정의가 누락된 결과입니다.

   ```
   Examples of the responses to the commands are shown in a box
   like this after each step in this task.
   These examples show the responses returned by IBM MQ for AIX.
   The responses vary on other platforms.
   ```

   c. 다음 단계를 진행하기 전에 큐 관리자가 시작되었는지 확인하십시오.

3. 저장소 정의를 추가하려면 큐 관리자 정의를 대체하십시오.

   전체 저장소를 보유하는 각 큐 관리자에서 ALTER QMGR 명령을 사용하고 REPOS 속성을 지정하여 로컬 큐 관리자 정의를 대체하십시오.

   ```
   ALTER QMGR REPOS(INVENTORY)
   ```

   ```
   1 : ALTER QMGR REPOS(INVENTORY)
   AMQ8005: IBM MQ queue manager changed.
   ```

   예를 들어, 다음을 입력하는 경우

   a. runmqsc LONDON
   b. ALTER QMGR REPOS(INVENTORY)

   LONDON이 전체 저장소로 변경됩니다.

4. 리스너를 정의하십시오.

   클러스터 내 모든 큐 관리자에 대해 다른 큐 관리자의 네트워크 요청을 승인하는 리스너를 정의하십시오. LONDON 큐 관리자에서 다음 명령을 실행하십시오.

   ```
   DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
   ```

   CONTROL 속성은 큐 관리자가 시작 및 중지될 때 리스너가 시작 및 중지되도록 합니다.

   리스너는 정의될 때 시작되지 않으므로 처음에는 다음 MQSC 명령을 사용하여 수동으로 시작해야 합니다.

```
START LISTENER(LONDON_LS)
```

각각에 대해 리스너 이름을 변경하여 클러스터에 있는 다른 모든 큐 관리자에 대해 유사한 명령을 발행하십시오.

리스너에 표시된 대로 이러한 리스너를 정의하는 몇 가지 방법이 있습니다.

5. LONDON 큐 관리자에 대해 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자에서 큐 관리자가 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의합니다. 클러스터 수신자 채널: CLUSRCVR을 참조하십시오. CLUSRCVR 채널은 큐 관리자의 연결 이름을 정의합니다. 연결 이름은 다른 큐 관리자가 참조할 수 있는 저장소에 저장됩니다. CLUSTER 키워드는 클러스터 내 다른 큐 관리자로부터 메시지를 수신하는 큐 관리자의 가용성을 표시합니다.

   이 예에서 채널 이름은 `INVENTORY.LONDON`이고 연결 이름 (`CONNAME`)은 큐 관리자가 상주하는 시스템의 네트워크 주소로, `LONDON.CHSTORE.COM`입니다. 이 네트워크 주소는 영숫자 DNS 호스트 이름 또는 IPv4 점분리 십진수 양식으로 된 IP 주소로 입력할 수 있습니다. 예를 들면, `192.0.2.0` 또는 IPv6 16진 양식(예: `2001:DB8:0204:acff:fe97:2c34:fde0:3485`)입니다. 포트 번호는 지정되지 않으므로 기본 포트 (1414)가 사용됩니다.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-receiver channel for queue manager LONDON')
   ```

   ```
   1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-receiver channel for queue manager LONDON')
   AMQ8014: WebSphere MQ channel created.
   07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
   ```

6. NEWYORK 큐 관리자에 대해 CLUSRCVR 채널을 정의하십시오.

   채널 리스너가 기본 포트(일반적으로 1414)를 사용 중이고 클러스터에 z/OS에서의 큐 관리자가 포함되지 않는 경우, `CONNAME`을 생략할 수 있습니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
   ```

7. LONDON 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   전체 저장소 큐 관리자 모두에서 클러스터에 있는 다른 모든 전체 저장소 큐 관리자까지 CLUSSDR 채널을 수동으로 정의합니다. 클러스터 송신자 채널: CLUSSDR을 참조하십시오. 이 경우 두 개의 큐 관리자만 있으며 둘 다 전체 저장소를 보유합니다. 이 두 큐 관리자는 각각 상대 큐 관리자에서 정의된 CLUSRCVR 채널을 가리키는 수동으로 정의된 CLUSSDR 채널이 필요합니다. CLUSSDR 정의에서 지정되는 채널 이름은 해당 CLUSRCVR 정의의 채널 이름과 일치해야 합니다. 큐 관리자에 같은 클러스터 내 클러스터 수신자 채널 및 클러스터 송신자 채널 둘 다에 대한 정의가 있는 경우 클러스터 송신자 채널이 시작됩니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   ```

   ```
   1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   AMQ8014: WebSphere MQ channel created.
   07/09/98 13:00:18  Channel program started.
   ```

8. NEWYORK 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   ```

```
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. 클러스터 큐 INVENTQ를 정의하십시오.

   CLUSTER 키워드를 지정하여 NEWYORK 큐 관리자에서 INVENTQ 큐를 정의하십시오.

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   ```

   ```
   1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   AMQ8006: WebSphere MQ queue created.
   ```

   CLUSTER 키워드를 사용하면 큐를 클러스터에 알릴 수 있습니다. 큐를 정의하면 바로 클러스터의 다른 큐 관리자가 이 큐를 사용할 수 있습니다. 해당 큐 관리자는 리모트 큐 정의를 작성하지 않고도 큐에 메시지를 송신할 수 있습니다.

   모든 정의가 완료되었습니다. 모든 플랫폼의 각 큐 관리자에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 대기하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

다음에 수행할 작업

이제 클러스터 확인을 (를) 준비할 수 있습니다.

관련 태스크

230 페이지의 『큐 관리자당 다중 전송 큐를 사용하여 TCP/IP에서 클러스터 설정』
이는 단순 클러스터에 대해 다른 구성을 설명하는 세 가지 주제 중 하나입니다.

233 페이지의 『z/OS에서 LU 6.2를 사용하여 클러스터 설정』
이는 단순 클러스터에 대해 다른 구성을 설명하는 트리 주제 중 하나입니다.

큐 관리자당 다중 전송 큐를 사용하여 *TCP/IP*에서 클러스터 설정

이는 단순 클러스터에 대해 다른 구성을 설명하는 세 가지 주제 중 하나입니다.

시작하기 전에

작성 중인 클러스터에 대한 개요는 226 페이지의 『새 클러스터 설정』의 내용을 참조하십시오.

이 태스크 정보

다음 단계에 따라 전송 프로토콜 TCP/IP를 사용하여 멀티플랫폼에서 클러스터를 설정하십시오. 저장소 큐 관리자는 다른 클러스터 전송 큐를 사용하여 클러스터 내에서 서로 및 다른 큐 관리자에게 메시지를 송신하도록 구성됩니다. 다른 전송 큐를 사용하는 클러스터에 큐 관리자를 추가할 경우 239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』 태스크를 수행하십시오.

프로시저

1. 클러스터의 조직 및 해당 이름을 결정하십시오.

   두 큐 관리자 LONDON 및 NEWYORK을 클러스터에 링크하기로 결정했습니다. 두 개의 큐 관리자만 있는 클러스터는 분산 큐잉을 사용하는 네트워크에 대해서는 약간의 이익만 제공하지만 분산 큐잉을 시작하기 좋은 방법이며 향후 확장을 위한 범위를 제공합니다. 저장소의 새 분기를 여는 경우 클러스터에 새 큐 관리자를 쉽게 추가할 수 있습니다. 새 큐 관리자를 추가해도 기존 네트워크에 지장을 주지 않습니다. 237 페이지의 『클러스터에 큐 관리자 추가』의 내용을 참조하십시오.

   잠시 동안 실행 중인 애플리케이션만 인벤토리 애플리케이션입니다. 클러스터 이름은 INVENTORY입니다.

2. 전체 저장소를 보유할 큐 관리자를 결정하십시오.

   모든 클러스터에서 전체 저장소를 보유할 최소 하나 이상의(가급적이면 둘) 큐 관리자를 지정해야 합니다. 이 예에는 둘 다 전체 저장소를 보유하는 두 큐 관리자인 LONDON 및 NEWYORK만 있습니다.

   a. 순서에 관계없이 나머지 단계를 수행할 수 있습니다.

b. 단계에 따라 진행하는 경우 경고 메시지가 큐 관리자 로그에 기록될 수 있습니다. 이러한 메시지는 아직 추가하지 않는 정의가 누락된 결과입니다.

```
Examples of the responses to the commands are shown in a box
like this after each step in this task.
These examples show the responses returned by IBM MQ for AIX.
The responses vary on other platforms.
```

c. 다음 단계를 진행하기 전에 큐 관리자가 시작되었는지 확인하십시오.

3. 저장소 정의를 추가하려면 큐 관리자 정의를 대체하십시오.

   전체 저장소를 보유하는 각 큐 관리자에서 ALTER QMGR 명령을 사용하고 REPOS 속성을 지정하여 로컬 큐 관리자 정의를 대체하십시오.

   ```
   ALTER QMGR REPOS(INVENTORY)
   ```

   ```
   1 : ALTER QMGR REPOS(INVENTORY)
   AMQ8005: IBM MQ queue manager changed.
   ```

   예를 들어, 다음을 입력하는 경우

   a. runmqsc LONDON

   b. ALTER QMGR REPOS(INVENTORY)

   LONDON이 전체 저장소로 변경됩니다.

4. 각 목적지에 대해 별도의 클러스터 전송 큐를 작성하려면 큐 관리자 정의를 대체하십시오.

   ```
   ALTER QMGR DEFCLXQ(CHANNEL)
   ```

   클러스터에 추가하는 각 큐 관리자에 대해 개별 전송 큐를 사용할지 여부를 결정하십시오. 237 페이지의 『클러스터에 큐 관리자 추가』 및 239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』 주제를 참조하십시오.

5. 리스너를 정의하십시오.

   클러스터 내 모든 큐 관리자에 대해 다른 큐 관리자의 네트워크 요청을 승인하는 리스너를 정의하십시오. LONDON 큐 관리자에서 다음 명령을 실행하십시오.

   ```
   DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
   ```

   CONTROL 속성은 큐 관리자가 시작 및 중지될 때 리스너가 시작 및 중지되도록 합니다.

   리스너는 정의될 때 시작되지 않으므로 처음에는 다음 MQSC 명령을 사용하여 수동으로 시작해야 합니다.

   ```
   START LISTENER(LONDON_LS)
   ```

   각각에 대해 리스너 이름을 변경하여 클러스터에 있는 다른 모든 큐 관리자에 대해 유사한 명령을 발행하십시오.

   리스너에 표시된 대로 이러한 리스너를 정의하는 몇 가지 방법이 있습니다.

6. LONDON 큐 관리자에 대해 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자에서 큐 관리자가 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의합니다. 클러스터 수신자 채널: CLUSRCVR을 참조하십시오. CLUSRCVR 채널은 큐 관리자의 연결 이름을 정의합니다. 연결 이름은 다른 큐 관리자가 참조할 수 있는 저장소에 저장됩니다. CLUSTER 키워드는 클러스터 내 다른 큐 관리자로부터 메시지를 수신하는 큐 관리자의 가용성을 표시합니다.

이 예에서 채널 이름은 INVENTORY.LONDON이고 연결 이름 (CONNAME)은 큐 관리자가 상주하는 시스템의 네트워크 주소로, LONDON.CHSTORE.COM입니다. 이 네트워크 주소는 영숫자 DNS 호스트 이름 또는 IPv4 점분리 십진수 양식으로 된 IP 주소로 입력할 수 있습니다. 예를 들면, 192.0.2.0 또는 IPv6 16진 양식(예: 2001:DB8:0204:acff:fe97:2c34:fde0:3485)입니다. 포트 번호는 지정되지 않으므로 기본 포트(1414)가 사용됩니다.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

7. NEWYORK 큐 관리자에 대해 CLUSRCVR 채널을 정의하십시오.

   채널 리스너가 기본 포트(일반적으로 1414)를 사용 중이고 클러스터에 z/OS에서의 큐 관리자가 포함되지 않는 경우, CONNAME을 생략할 수 있습니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
   ```

8. LONDON 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   전체 저장소 큐 관리자 모두에서 클러스터에 있는 다른 모든 전체 저장소 큐 관리자까지 CLUSSDR 채널을 수동으로 정의합니다. 클러스터 송신자 채널: CLUSSDR을 참조하십시오. 이 경우 두 개의 큐 관리자만 있으며 둘 다 전체 저장소를 보유합니다. 이 두 큐 관리자는 각각 상대 큐 관리자에서 정의된 CLUSRCVR 채널을 가리키는 수동으로 정의된 CLUSSDR 채널이 필요합니다. CLUSSDR 정의에서 지정되는 채널 이름은 해당 CLUSRCVR 정의의 채널 이름과 일치해야 합니다. 큐 관리자에 같은 클러스터 내 클러스터 수신자 채널 및 클러스터 송신자 채널 둘 다에 대한 정의가 있는 경우 클러스터 송신자 채널이 시작됩니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   ```

   ```
   1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
   AMQ8014: WebSphere MQ channel created.
   07/09/98 13:00:18  Channel program started.
   ```

9. NEWYORK 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
   ```

10. 클러스터 큐 INVENTQ를 정의하십시오.

    CLUSTER 키워드를 지정하여 NEWYORK 큐 관리자에서 INVENTQ 큐를 정의하십시오.

    ```
    DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
    ```

    ```
    1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
    AMQ8006: WebSphere MQ queue created.
    ```

CLUSTER 키워드를 사용하면 큐를 클러스터에 알릴 수 있습니다. 큐를 정의하면 바로 클러스터의 다른 큐 관리자가 이 큐를 사용할 수 있습니다. 해당 큐 관리자는 리모트 큐 정의를 작성하지 않고도 큐에 메시지를 송신할 수 있습니다.

모든 정의가 완료되었습니다. 모든 플랫폼의 각 큐 관리자에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 대기하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

다음에 수행할 작업

이제 클러스터 확인을 (를) 준비할 수 있습니다.

관련 태스크

227 페이지의 『TCP/IP를 사용하여 큐 관리자당 단일 전송 큐가 있는 클러스터 설정』
이는 단순 클러스터에 대해 다른 구성을 설명하는 세 가지 주제 중 하나입니다.

233 페이지의 『z/OS에서 LU 6.2를 사용하여 클러스터 설정』
이는 단순 클러스터에 대해 다른 구성을 설명하는 트리 주제 중 하나입니다.

*z/OS*에서 *LU 6.2*를 사용하여 클러스터 설정

이는 단순 클러스터에 대해 다른 구성을 설명하는 트리 주제 중 하나입니다.

시작하기 전에

작성 중인 클러스터에 대한 개요는 226 페이지의 『새 클러스터 설정』의 내용을 참조하십시오.

프로시저

1. 클러스터의 조직 및 해당 이름을 결정하십시오.

   두 큐 관리자 LONDON 및 NEWYORK을 클러스터에 링크하기로 결정했습니다. 두 개의 큐 관리자만 있는 클러스터는 분산 큐잉을 사용하는 네트워크에 대해서는 약간의 이익만 제공하지만 분산 큐잉을 시작하기 좋은 방법이며 향후 확장을 위한 범위를 제공합니다. 저장소의 새 분기를 여는 경우 클러스터에 새 큐 관리자를 쉽게 추가할 수 있습니다. 새 큐 관리자를 추가해도 기존 네트워크에 지장을 주지 않습니다. 237 페이지의 『클러스터에 큐 관리자 추가』의 내용을 참조하십시오.

   잠시 동안 실행 중인 애플리케이션만 인벤토리 애플리케이션입니다. 클러스터 이름은 INVENTORY입니다.
2. 전체 저장소를 보유할 큐 관리자를 결정하십시오.

   모든 클러스터에서 전체 저장소를 보유할 최소 하나 이상의(가급적이면 둘) 큐 관리자를 지정해야 합니다. 이 예에는 둘 다 전체 저장소를 보유하는 두 큐 관리자인 LONDON 및 NEWYORK만 있습니다.

   a. 순서에 관계없이 나머지 단계를 수행할 수 있습니다.
   b. 단계에 따라 진행하는 중에 z/OS 시스템 콘솔에 경고 메시지가 기록될 수 있습니다. 이러한 메시지는 아직 추가하지 않는 정의가 누락된 결과입니다.
   c. 다음 단계를 진행하기 전에 큐 관리자가 시작되었는지 확인하십시오.
3. 저장소 정의를 추가하려면 큐 관리자 정의를 대체하십시오.

   전체 저장소를 보유하는 각 큐 관리자에서 ALTER QMGR 명령을 사용하고 REPOS 속성을 지정하여 로컬 큐 관리자 정의를 대체하십시오.

   ```
   ALTER QMGR REPOS(INVENTORY)
   ```

   ```
   1 : ALTER QMGR REPOS(INVENTORY)
   AMQ8005: IBM MQ queue manager changed.
   ```

   예를 들어, 다음을 입력하는 경우

   a. `runmqsc LONDON`
   b. `ALTER QMGR REPOS(INVENTORY)`

LONDON이 전체 저장소로 변경됩니다.

4. 리스너를 정의하십시오.

   z/OS z/OS의 채널 시작기 및 679 페이지의 『LU 6.2에서 수신』의 내용을 참조하십시오.

   리스너는 정의될 때 시작되지 않으므로 처음에는 다음 MQSC 명령을 사용하여 수동으로 시작해야 합니다.

```
START LISTENER(LONDON_LS)
```

   각각에 대해 리스너 이름을 변경하여 클러스터에 있는 다른 모든 큐 관리자에 대해 유사한 명령을 발행하십시오.

5. LONDON 큐 관리자에 대해 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자에서 큐 관리자가 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의합니다. 클러스터 수신자 채널: CLUSRCVR을 참조하십시오. CLUSRCVR 채널은 큐 관리자의 연결 이름을 정의합니다. 연결 이름은 다른 큐 관리자가 참조할 수 있는 저장소에 저장됩니다. CLUSTER 키워드는 클러스터 내 다른 큐 관리자로부터 메시지를 수신하는 큐 관리자의 가용성을 표시합니다.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. NEWYORK 큐 관리자에 대해 CLUSRCVR 채널을 정의하십시오.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager NEWYORK')
```

7. LONDON 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   전체 저장소 큐 관리자 모두에서 클러스터에 있는 다른 모든 전체 저장소 큐 관리자까지 CLUSSDR 채널을 수동으로 정의합니다. 클러스터 송신자 채널: CLUSSDR을 참조하십시오. 이 경우 두 개의 큐 관리자만 있으며 둘 다 전체 저장소를 보유합니다. 이 두 큐 관리자는 각각 상대 큐 관리자에서 정의된 CLUSRCVR 채널을 가리키는 수동으로 정의된 CLUSSDR 채널이 필요합니다. CLUSSDR 정의에서 지정되는 채널 이름은 해당 CLUSRCVR 정의의 채널 이름과 일치해야 합니다. 큐 관리자에 같은 클러스터 내 클러스터 수신자 채널 및 클러스터 송신자 채널 둘 다에 대한 정의가 있는 경우 클러스터 송신자 채널이 시작됩니다.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(CPIC) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: WebSphere MQ channel created.
07/09/98 13:00:18  Channel program started.
```

8. NEWYORK 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
```

```
CONNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. 클러스터 큐 INVENTQ를 정의하십시오.

   CLUSTER 키워드를 지정하여 NEWYORK 큐 관리자에서 INVENTQ 큐를 정의하십시오.

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   ```

   ```
   1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   AMQ8006: WebSphere MQ queue created.
   ```

   CLUSTER 키워드를 사용하면 큐를 클러스터에 알릴 수 있습니다. 큐를 정의하면 바로 클러스터의 다른 큐 관리자가 이 큐를 사용할 수 있습니다. 해당 큐 관리자는 리모트 큐 정의를 작성하지 않고도 큐에 메시지를 송신할 수 있습니다.

   모든 정의가 완료되었습니다. 모든 플랫폼의 각 큐 관리자에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 대기하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

다음에 수행할 작업

이제 클러스터 확인을 (를) 준비할 수 있습니다.

관련 태스크

227 페이지의 『TCP/IP를 사용하여 큐 관리자당 단일 전송 큐가 있는 클러스터 설정』
이는 단순 클러스터에 대해 다른 구성을 설명하는 세 가지 주제 중 하나입니다.

230 페이지의 『큐 관리자당 다중 전송 큐를 사용하여 TCP/IP에서 클러스터 설정』
이는 단순 클러스터에 대해 다른 구성을 설명하는 세 가지 주제 중 하나입니다.

클러스터 확인

피어 주제에서는 단순 클러스터에 대해 서로 다른 세 가지 구성을 설명합니다. 이 주제에서는 클러스터를 확인하는 방법에 대해 설명합니다.

시작하기 전에

이 주제에서는 다음 태스크 중 하나를 통해 작성한 클러스터를 확인한다고 가정합니다.

- 227 페이지의 『TCP/IP를 사용하여 큐 관리자당 단일 전송 큐가 있는 클러스터 설정』.
- 230 페이지의 『큐 관리자당 다중 전송 큐를 사용하여 TCP/IP에서 클러스터 설정』.
- 233 페이지의 『z/OS에서 LU 6.2를 사용하여 클러스터 설정』.

작성된 클러스터에 대한 개요는 226 페이지의 『새 클러스터 설정』의 내용을 참조하십시오.

이 태스크 정보

다음 방법 중 하나 이상을 사용하여 클러스터를 확인할 수 있습니다.

1. 클러스터 및 채널 속성을 표시하는 관리 명령을 실행합니다.
2. 샘플 프로그램을 실행하여 클러스터 큐에서 메시지를 송신 및 수신합니다.
3. 요청 메시지를 클러스터에 송신하고 비클러스터 응답 큐에 응답 메시지로 응답하는 자체 프로그램을 작성합니다.

프로시저

DISPLAY **runmqsc** 명령을 발행하여 클러스터를 확인하십시오.

표시되는 응답은 다음 단계의 응답과 유사해야 합니다.

1. NEWYORK 큐 관리자에서 **DISPLAY CLUSQMGR** 명령을 실행하십시오.

```
dis clusqmgr(*)
```

```
1 : dis clusqmgr(*)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(NEWYORK)       CLUSTER(INVENTORY)
CHANNEL(INVENTORY.NEWYORK)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(LONDON)       CLUSTER(INVENTORY)
CHANNEL(INVENTORY.LONDON)
```

2. NEWYORK 큐 관리자에서 **DISPLAY CHANNEL STATUS** 명령을 실행하십시오.

```
dis chstatus(*)
```

```
1 : dis chstatus(*)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.NEWYORK)  XMITQ( )
CONNAME(192.0.2.0)      CURRENT
CHLTYPE(CLUSRCVR)       STATUS(RUNNING)
RQMNAME(LONDON)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.LONDON) XMITQ(SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON)
CONNAME(192.0.2.1)      CURRENT
CHLTYPE(CLUSSDR)       STATUS(RUNNING)
RQMNAME(LONDON)
```

amqsput을 사용하여 두 큐 관리자 사이에서 메시지를 송신하십시오.

3. LONDON에서 **amqsput INVENTQ LONDON** 명령을 실행하십시오.

   몇 가지 메시지를 입력하고 공백 행을 입력하십시오.

4. NEWYORK에서 **amqsget INVENTQ NEWYORK** 명령을 실행하십시오.

   이제 LONDON에 입력한 메시지가 표시됩니다. 15초 후에 프로그램이 종료됩니다.

자체 프로그램을 사용하여 두 큐 관리자 사이에서 메시지를 송신하십시오.

다음 단계에서 LONDON은 메시지를 NEWYORK의 INVENTQ에 넣고 해당 큐 LONDON_reply에서 응답을 수신합니다.

5. LONDON에서 메시지를 클러스터 큐에 넣으십시오.
   a) LONDON_reply이라는 로컬 큐를 정의하십시오.
   b) MQOPEN 옵션을 MQOO_OUTPUT로 설정하십시오.
   c) MQOPEN 호출을 실행하여 INVENTQ큐를 여십시오.
   d) 메시지 설명자의 *ReplyToQ* 이름을 LONDON_reply로 설정하십시오.
   e) 메시지를 넣으려면 MQPUT 호출을 실행하십시오.
   f) 메시지를 커미트하십시오.
6. NEWYORK의 클러스터 큐에서 메시지를 수신하고 응답을 응답 큐에 넣으십시오.
   a) MQOPEN 옵션을 MQOO_BROWSE로 설정하십시오.
   b) MQOPEN 호출을 실행하여 INVENTQ큐를 여십시오.
   c) MQGET 호출을 실행하여 INVENTQ에서 메시지를 가져오십시오.
   d) 메시지 설명자에서 *ReplyToQ* 이름을 검색합니다.
   e) 오브젝트 설명자의 ObjectName 필드에 *ReplyToQ* 이름을 입력하십시오.
   f) MQOPEN 옵션을 MQOO_OUTPUT로 설정하십시오.
   g) MQOPEN 호출을 실행하여 큐 관리자 LONDON에서 LONDON_reply 을 여십시오.
   h) MQPUT 호출을 실행하여 LONDON_reply에 메시지를 넣으십시오.

7. LONDON에서 응답을 수신하십시오.
   a) MQOPEN 옵션을 MQOO_BROWSE로 설정하십시오.
   b) MQOPEN 호출을 실행하여 LONDON_reply큐를 여십시오.
   c) MQGET 호출을 실행하여 LONDON_reply에서 메시지를 가져오십시오.

클러스터에 큐 관리자 추가

다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 단일 클러스터 전송 큐 SYSTEM.CLUSTER.TRANSMIT.QUEUE를 사용하여 전송됩니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 226 페이지의 『새 클러스터 설정』에서 설명된 대로 설정됩니다. 이 클러스터는 둘 다 전체 저장소를 보유하는 두 개의 큐 관리자 LONDON 및 NEWYORK을 포함합니다.
- 큐 관리자 PARIS는 기본 설치에 속합니다. 그렇지 않은 경우 **setmqenv** 명령을 실행하여 PARIS가 속한 설치에 대한 명령 환경을 설정해야 합니다.
- TCP 연결성은 세 시스템 사이에 모두 존재하며 큐 관리자는 큐 관리자의 제어 하에 시작하는 TCP 리스너로 구성됩니다.

이 태스크 정보

1. 체인 저장소의 새 분기가 Paris에서 설정되고 PARIS라고 하는 큐 관리자를 클러스터에 추가하려고 합니다.
2. 큐 관리자 PARIS는 메시지를 INVENTQ 큐에 넣어 New York의 시스템에서 실행되는 애플리케이션에 인벤토리 업데이트를 송신합니다.

다음 단계에 따라 큐 관리자를 클러스터에 추가하십시오.

프로시저

1. PARIS가 처음 참조할 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 전체 저장소 중 하나를 참조해야 합니다. 큐 관리자는 전체 저장소에서 클러스터에 대한 정보를 수집하고 자체 부분 저장소를 빌드합니다. 저장소 중 하나를 전체 저장소로 선택하십시오. 새 큐 관리자는 클러스터에 추가되는 즉시 다른 저장소에 대해서도 학습합니다. 큐 관리자 변경에 대한 정보는 두 저장소에 직접 송신됩니다. 이 예에서는 순전히 지리적 이유로 PARIS를 큐 관리자 LONDON에 링크합니다.

   참고: 큐 관리자 PARIS가 시작된 후에는 순서에 관계없이 나머지 단계를 수행하십시오.
2. 큐 관리자 PARIS에서 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의해야 합니다. PARIS에서 다음을 정의하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for queue manager PARIS')
   ```

   클러스터 수신자 채널은 클러스터 INVENTORY에 있는 다른 큐 관리자로부터 메시지를 수신할 큐 관리자의 가용성을 알립니다. 클러스터 수신자 채널 INVENTORY.PARIS에 대한 송신 측의 다른 큐 관리자에서 정의를 작성하지 마십시오. 기타 정의는 필요한 경우 자동으로 작성됩니다. 클러스터 채널을 참조하십시오.
3. z/OS
   IBM MQ for z/OS에서 채널 시작기를 시작하십시오.

4. 큐 관리자 PARIS에서 CLUSSDR 채널을 정의하십시오.

   전체 저장소가 아닌 큐 관리자를 클러스터에 추가할 때 전체 저장소에 대한 초기 연결을 수행하기 위해 한 개의 클러스터 송신자 채널만 정의합니다. 클러스터 송신자 채널: CLUSSDR을 참조하십시오.

   PARIS에서 네트워크 주소가 LONDON.CHSTORE.COM인 큐 관리자에 대한 INVENTORY.LONDON이라고 하는 CLUSSDR 채널에 대해 다음 정의를 작성하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from PARIS to repository at LONDON')
   ```

5. 옵션: 이전에 같은 클러스터에서 제거되었던 큐 관리자를 클러스터에 추가하는 경우 이 큐 관리자가 이제 클러스터 멤버로 표시되는지 확인하십시오. 그렇지 않은 경우 다음 추가 단계를 완료하십시오.

   a) 추가할 큐 관리자에서 **REFRESH CLUSTER** 명령을 발행하십시오.

      이 단계는 클러스터 채널을 중지하고 로컬 클러스터 캐시에 클러스터의 나머지 부분에서 최신임이 보장되는 새로운 순서 번호 세트를 제공합니다.

      ```
      REFRESH CLUSTER(INVENTORY) REPOS(YES)
      ```

      참고: 대형 클러스터의 경우, **REFRESH CLUSTER** 명령을 사용하면 진행 중에 클러스터에 혼란을 줄 수 있으며, 클러스터 오브젝트가 모든 관심 있는 큐 관리자에 자동으로 상태 업데이트를 보낸 이후 27일 간격으로 다시 수행됩니다. 대형 클러스터를 새로 고치면 클러스터의 성능 및 가용성에 영향을 줄 수 있음을 참조하십시오.

   b) CLUSSDR 채널을 재시작하십시오.

      (예: START CHANNEL 명령 사용).

   c) CLUSRCVR 채널을 재시작하십시오.

결과

다음 그림은 이 태스크를 통해 설정되는 클러스터를 보여줍니다.

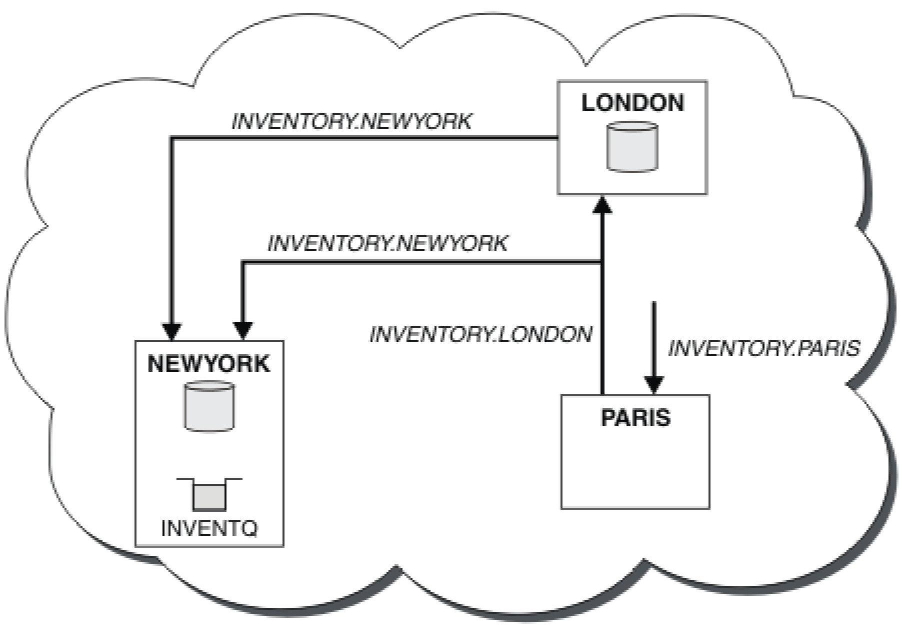


그림 *40.* 세 개의 큐 관리자가 있는 *INVENTORY* 클러스터

CLUSRCVR 정의 및 CLUSSDR 정의 두 개만 작성하여 큐 관리자 PARIS를 클러스터에 추가했습니다.

이제 PARIS 큐 관리자는 LONDON의 전체 저장소에서 INVENTQ 큐가 큐 관리자 NEWYORK에 의해 호스트됨을 학습합니다. Paris의 시스템에 의해 호스팅되는 애플리케이션이 INVENTQ에 메시지를 넣으려고 시도하면 PARIS가 클러스터 수신자 채널 INVENTORY.NEWYORK에 연결하기 위한 클러스터 송신자 채널을 자동으로 정의합니다. 이 애플리케이션은 해당 큐 관리자 이름이 대상 큐 관리자로 지정되고 응답 대상 큐가 제공되는 경우에 응답을 수신할 수 있습니다.

클러스터에 큐 관리자 추가*:* 개별 전송 큐

다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

시작하기 전에

- 큐 관리자는 클러스터의 멤버가 아닙니다.
- 클러스터가 존재하고 이 큐 관리자가 직접 연결할 수 있는 전체 저장소가 있으며 이 저장소가 사용 가능합니다. 클러스터를 작성하기 위한 단계는 226 페이지의 『새 클러스터 설정』의 내용을 참조하십시오.

이 태스크 정보

이 태스크는 237 페이지의 『클러스터에 큐 관리자 추가』에 대한 대안으로, 클러스터 메시지를 단일 전송 큐에 배치하는 클러스터에 큐 관리자를 추가합니다.

이 태스크에서는 각 클러스터 송신자 채널에 대해 개별 클러스터 전송 큐를 자동으로 작성하는 클러스터에 큐 관리자를 추가합니다.

큐의 정의 수를 작게 유지하기 위해 기본값은 단일 전송 큐를 사용하는 것입니다. 여러 큐 관리자와 여러 클러스터를 목적지로 하는 트래픽을 모니터하려는 경우 개별 전송 큐를 사용하는 것이 유리합니다. 또한 다른 목적지로의 트래픽을 분리하여 격리 또는 성능 목표를 달성할 수도 있습니다.

프로시저

1. 기본 클러스터 채널 전송 큐 유형을 대체하십시오.

   큐 관리자 PARIS를 다음과 같이 대체하십시오.

   ```
   ALTER QMGR DEFCLXQ(CHANNEL)
   ```

   큐 관리자는 큐 관리자에게 메시지를 송신하기 위한 클러스터 송신자 채널을 작성할 때마다 클러스터 전송 큐를 작성합니다. 이 전송 큐는 이 클러스터 송신자 채널에서만 사용됩니다. 전송 큐는 영구적이고 동적입니다. 전송 큐는 모델 큐 SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE로부터 이름 SYSTEM.CLUSTER.TRANSMIT. *ChannelName*으로 작성됩니다.

   주의: IBM WebSphere MQ 7.5보다 이전 버전의 제품에서 업그레이드된 큐 관리자와 함께 전용 SYSTEM.CLUSTER.TRANSMIT.QUEUES을(를) 사용하는 경우, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE에 SHARE/NOSHARE 옵션이 **SHARE**로 설정되어 있는지 확인하십시오.

2. PARIS가 처음 참조할 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 전체 저장소 중 하나를 참조해야 합니다. 큐 관리자는 전체 저장소에서 클러스터에 대한 정보를 수집하고 자체 부분 저장소를 빌드합니다. 저장소 중 하나를 전체 저장소로 선택하십시오. 새 큐 관리자는 클러스터에 추가되는 즉시 다른 저장소에 대해서도 학습합니다. 큐 관리자 변경에 대한 정보는 두 저장소에 직접 송신됩니다. 이 예에서는 순전히 지리적 이유로 PARIS를 큐 관리자 LONDON에 링크합니다.

   참고: 큐 관리자 PARIS가 시작된 후에는 순서에 관계없이 나머지 단계를 수행하십시오.

3. 큐 관리자 PARIS에서 CLUSRCVR 채널을 정의하십시오.

클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의해야 합니다. PARIS에서 다음을 정의하십시오.

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

클러스터 수신자 채널은 클러스터 INVENTORY에 있는 다른 큐 관리자로부터 메시지를 수신할 큐 관리자의 가용성을 알립니다. 클러스터 수신자 채널 INVENTORY.PARIS에 대한 송신 측의 다른 큐 관리자에서 정의를 작성하지 마십시오. 기타 정의는 필요한 경우 자동으로 작성됩니다. 클러스터 채널을 참조하십시오.

4. 큐 관리자 PARIS에서 CLUSSDR 채널을 정의하십시오.

   전체 저장소가 아닌 큐 관리자를 클러스터에 추가할 때 전체 저장소에 대한 초기 연결을 수행하기 위해 한 개의 클러스터 송신자 채널만 정의합니다. 클러스터 송신자 채널: CLUSSDR을 참조하십시오.

   PARIS에서 네트워크 주소가 LONDON.CHSTORE.COM인 큐 관리자에 대한 INVENTORY.LONDON이라고 하는 CLUSSDR 채널에 대해 다음 정의를 작성하십시오.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

   큐 관리자는 모델 큐 SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE로부터 영구적 동적 클러스터 전송 큐 SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON을 자동으로 작성합니다. 큐 관리자는 전송 큐의 CLCHNAME 속성을 INVENTORY.LONDON으로 설정합니다.

결과

다음 그림은 이 태스크를 통해 설정되는 클러스터를 보여줍니다.

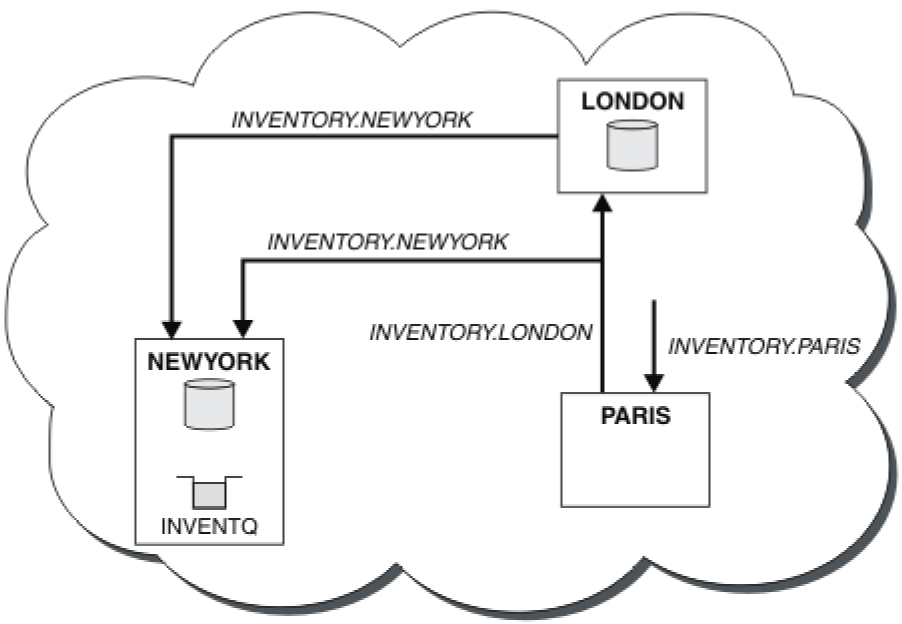


그림 *41.* 세 개의 큐 관리자가 있는 *INVENTORY* 클러스터

CLUSRCVR 정의 및 CLUSSDR 정의 두 개만 작성하여 큐 관리자 PARIS를 클러스터에 추가했습니다.

이제 PARIS 큐 관리자는 LONDON의 전체 저장소에서 INVENTQ 큐가 큐 관리자 NEWYORK에 의해 호스트됨을 학습합니다. Paris의 시스템에 의해 호스팅되는 애플리케이션이 INVENTQ에 메시지를 넣으려고 시도하면 PARIS가 클러스터 수신자 채널 INVENTORY.NEWYORK에 연결하기 위한 클러스터 송신자 채널을 자동으로 정의합니

다. 이 애플리케이션은 해당 큐 관리자 이름이 대상 큐 관리자로 지정되고 응답 대상 큐가 제공되는 경우에 응답을 수신할 수 있습니다.

관련 태스크

DHCP를 사용하여 클러스터에 큐 관리자 추가
DHCP를 사용하여 클러스터에 큐 관리자를 추가합니다. 이 태스크는 CLUSRCVR 정의에서 CONNAME 값을 생략하는 것을 보여줍니다.

*DHCP*를 사용하여 클러스터에 큐 관리자 추가

DHCP를 사용하여 클러스터에 큐 관리자를 추가합니다. 이 태스크는 CLUSRCVR 정의에서 CONNAME 값을 생략하는 것을 보여줍니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

이 태스크는 두 개의 특수 기능을 보여줍니다.

- CLUSRCVR 정의에서 CONNAME 값을 생략하는 기능
- CLUSSDR 정의에서 +QMNAME+을 사용하는 기능

이 기능 둘 다 z/OS에서는 제공되지 않습니다.

시나리오:

- INVENTORY 클러스터가 226 페이지의 『새 클러스터 설정』에서 설명된 대로 설정되었습니다. 이 클러스터는 둘 다 전체 저장소를 보유하는 두 개의 큐 관리자 LONDON 및 NEWYORK을 포함합니다.
- 체인 저장소의 새 분기가 Paris에서 설정되고 PARIS라고 하는 큐 관리자를 클러스터에 추가하려고 합니다.
- 큐 관리자 PARIS는 메시지를 INVENTQ 큐에 넣어 New York의 시스템에서 실행되는 애플리케이션에 인벤토리 업데이트를 송신합니다.
- 세 시스템 사이에는 모두 네트워크 연결성이 있습니다.
- 네트워크 프로토콜은 TCP입니다.
- PARIS 큐 관리자 시스템은 DHCP를 사용하여 이는 IP 주소가 시스템 재시작 시 변경될 수 있음을 의미합니다.
- PARIS와 LONDON 시스템 간 채널의 이름은 정의된 이름 지정 규칙에 따라 지정됩니다. 이 규칙에서는 LONDON의 전체 저장소 큐 관리자의 큐 관리자 이름을 사용합니다.
- PARIS 큐 관리자의 관리자에게는 LONDON 저장소에 있는 큐 관리자의 이름에 대한 정보가 없습니다. LONDON 저장소에서 큐 관리자의 이름은 변경될 수 있습니다.

이 태스크 정보

다음 단계에 따라 DHCP를 사용하여 클러스터에 큐 관리자를 추가하십시오.

프로시저

1. PARIS가 처음 참조할 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 전체 저장소 중 하나를 참조해야 합니다. 큐 관리자는 전체 저장소에서 클러스터에 대한 정보를 수집하고 자체 부분 저장소를 빌드합니다. 저장소 중 하나를 전체 저장소로 선택하십시오. 새 큐 관리자는 클러스터에 추가되는 즉시 다른 저장소에 대해서도 학습합니다. 큐 관리자 변경에 대한 정보는 두 저장소에 직접 송신됩니다. 이 예에서는 순전히 지리적 이유로 PARIS를 큐 관리자 LONDON에 링크하도록 선택합니다.

   참고: 큐 관리자 PARIS가 시작된 후에는 순서에 관계없이 나머지 단계를 수행하십시오.
2. 큐 관리자 PARIS에서 CLUSRCVR 채널을 정의하십시오.

클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의해야 합니다. PARIS에서 다음을 정의하십시오.

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR)
TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

클러스터 수신자 채널은 클러스터 INVENTORY에 있는 다른 큐 관리자로부터 메시지를 수신할 큐 관리자의 가용성을 알립니다. 클러스터 수신자 채널에서 CONNAME을 지정하지 않아도 됩니다. CONNAME을 생략하거나 CONNAME(' ')을 지정하여 시스템에서 연결 이름을 찾도록 IBM MQ을(를) 요청할 수 있습니다. IBM MQ은(는) 시스템의 현재 IP 주소를 사용하여 CONNAME 값을 생성합니다. CONNAME을(를) 참조하십시오. 클러스터 수신자 채널 INVENTORY.PARIS에 대한 송신 측의 다른 큐 관리자에서 정의를 작성하지 않아도 됩니다. 기타 정의는 필요한 경우 자동으로 작성됩니다.

3. 큐 관리자 PARIS에서 CLUSSDR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 해당 초기 전체 저장소에 송신할 수 있는 한 개의 클러스터 송신자 채널을 정의해야 합니다. PARIS에서 네트워크 주소가 LONDON.CHSTORE.COM인 큐 관리자에 대한 INVENTORY.+QMNAME+이라고 하는 채널에 대해 다음 정의를 작성하십시오.

```
DEFINE CHANNEL(INVENTORY.+QMNAME+) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. 옵션: 이전에 같은 클러스터에서 제거되었던 큐 관리자를 클러스터에 추가하는 경우 이 큐 관리자가 이제 클러스터 멤버로 표시되는지 확인하십시오. 그렇지 않은 경우 다음 추가 단계를 완료하십시오.

   a) 추가할 큐 관리자에서 **REFRESH CLUSTER** 명령을 발행하십시오.

      이 단계는 클러스터 채널을 중지하고 로컬 클러스터 캐시에 클러스터의 나머지 부분에서 최신임이 보장되는 새로운 순서 번호 세트를 제공합니다.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

      참고: 대형 클러스터의 경우, **REFRESH CLUSTER** 명령을 사용하면 진행 중에 클러스터에 혼란을 줄 수 있으며, 클러스터 오브젝트가 모든 관심 있는 큐 관리자에 자동으로 상태 업데이트를 보낸 이후 27일 간격으로 다시 수행됩니다. 대형 클러스터를 새로 고치면 클러스터의 성능 및 가용성에 영향을 줄 수 있음을 참조하십시오.

   b) CLUSSDR 채널을 재시작하십시오.

      (예: START CHANNEL 명령 사용).

   c) CLUSRCVR 채널을 재시작하십시오.

결과

이 태스크에서 설정된 클러스터는 237 페이지의 『클러스터에 큐 관리자 추가』에 대한 클러스터와 같습니다.

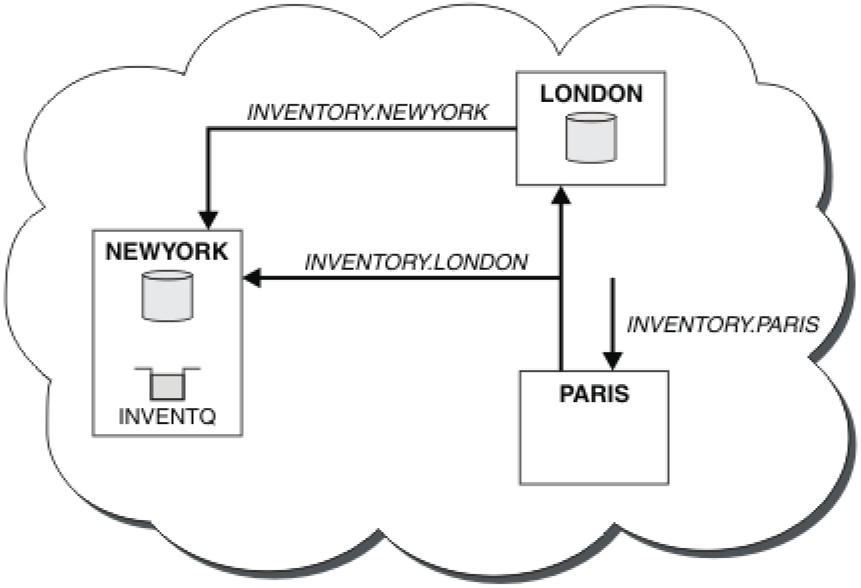


그림 *42.* 세 개의 큐 관리자가 있는 *INVENTORY* 클러스터

CLUSRCVR 정의 및 CLUSSDR 정의 두 개만 작성하여 큐 관리자 PARIS를 클러스터에 추가했습니다.

PARIS 큐 관리자에서 문자열 +QMNAME+를 포함하는 CLUSSDR이 시작됩니다. LONDON 시스템에서 IBM MQ은(는) +QMNAME+을 큐 관리자 이름(LONDON)으로 해석합니다. IBM MQ 그런 다음 INVENTORY.LONDON이라는 채널에 대한 정의를 해당 CLUSRCVR 정의와 일치시킵니다.

IBM MQ는 해석된 채널 이름을 PARIS 큐 관리자로 다시 송신합니다. PARIS에서 INVENTORY.+QMNAME+이라고 하는 채널에 대한 CLUSSDR 채널 정의는 INVENTORY.LONDON에 대해 내부적으로 생성된 CLUSSDR 정의로 바뀝니다. 이 정의는 해석된 채널 이름을 포함하지만 그렇지 않은 경우 사용자가 작성한 +QMNAME+ 정의와 같습니다. 또한 클러스터 저장소는 새로 해석된 채널 이름을 가진 채널 정의로 최신화됩니다.

참고:

1. +QMNAME+ 이름을 사용하여 작성된 채널은 즉시 비활성 상태가 됩니다. 이 채널은 데이터를 전송하는 데에는 사용되지 않습니다.
2. 채널 엑시트는 한 호출과 다음 호출 사이의 채널 이름 변경을 확인할 수 있습니다.

이제 PARIS 큐 관리자는 LONDON의 저장소에서 INVENTQ 큐가 큐 관리자 NEWYORK에 의해 호스트됨을 학습합니다. Paris의 시스템에 의해 호스팅되는 애플리케이션이 INVENTQ에 메시지를 넣으려고 시도하면 PARIS가 클러스터 수신자 채널 INVENTORY.NEWYORK에 연결하기 위한 클러스터 송신자 채널을 자동으로 정의합니다. 이 애플리케이션은 해당 큐 관리자 이름이 대상 큐 관리자로 지정되고 응답 대상 큐가 제공되는 경우에 응답을 수신할 수 있습니다.

관련 태스크

클러스터에 큐 관리자 추가: 개별 전송 큐
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

관련 정보

DEFINE CHANNEL

큐를 호스팅하는 큐 관리자 추가

다른 INVENTQ 큐를 호스팅하기 위해 다른 큐 관리자를 클러스터에 추가합니다. 요청은 각 큐 관리자의 큐에 번갈아 송신됩니다. 기존 INVENTQ 호스트에 대한 변경은 필요하지 않습니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되었습니다. 이 클러스터는 세 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS는 부분 저장소를 보유합니다. 인벤토리 애플리케이션은 NEWYORK 큐 관리자에 연결된 New York의 시스템에서 실행됩니다. 애플리케이션은 INVENTQ 큐에 메시지가 도착하면 구동됩니다.
- 새 저장소가 Toronto에서 설정됩니다. 추가 용량을 제공하기 위해 New York뿐만 아니라 Toronto의 시스템에서도 인벤토리 애플리케이션을 실행하려고 합니다.
- 네 시스템 사이에는 모두 네트워크 연결성이 있습니다.
- 네트워크 프로토콜은 TCP입니다.

참고: 큐 관리자 TORONTO는 부분 저장소만 포함합니다. 클러스터에 전체 저장소 큐 관리자를 추가하려는 경우 248 페이지의 『다른 큐 관리자로 전체 저장소 이동』의 내용을 참조하십시오.

이 태스크 정보

다음 단계에 따라 큐를 호스팅하는 큐 관리자를 추가하십시오.

프로시저

1. TORONTO가 처음 참조할 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 전체 저장소 중 하나를 참조해야 합니다. 큐 관리자는 전체 저장소에서 클러스터에 대한 정보를 수집하고 자체 부분 저장소를 빌드합니다. 저장소 선택이 특별히 중요한 것은 아닙니다. 이 예에서는 NEWYORK을 선택합니다. 새 큐 관리자는 클러스터에 조인하고 나면 두 저장소와 모두 통신합니다.
2. CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의해야 합니다. TORONTO에서 CLUSRCVR 채널을 정의하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.TORONTO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(TORONTO.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for TORONTO')
   ```

   TORONTO 큐 관리자는 해당 클러스터 수신자 채널을 사용하여 INVENTORY 클러스터에 있는 다른 큐 관리자로부터 메시지를 수신하기 위해 그 가용성을 알립니다.
3. 큐 관리자 TORONTO에서 CLUSSDR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 해당 첫 번째 전체 저장소에 송신할 수 있는 한 개의 클러스터 송신자 채널을 정의해야 합니다. 이 경우에서는 NEWYORK을 선택하십시오. TORONTO에는 다음 정의가 필요합니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from TORONTO to repository at NEWYORK')
   ```
4. 옵션: 이전에 같은 클러스터에서 제거되었던 큐 관리자를 클러스터에 추가하는 경우 이 큐 관리자가 이제 클러스터 멤버로 표시되는지 확인하십시오. 그렇지 않은 경우 다음 추가 단계를 완료하십시오.

   a) 추가할 큐 관리자에서 **REFRESH CLUSTER** 명령을 발행하십시오.

      이 단계는 클러스터 채널을 중지하고 로컬 클러스터 캐시에 클러스터의 나머지 부분에서 최신임이 보장되는 새로운 순서 번호 세트를 제공합니다.

      ```
      REFRESH CLUSTER(INVENTORY) REPOS(YES)
      ```

참고: 대형 클러스터의 경우, **REFRESH CLUSTER** 명령을 사용하면 진행 중에 클러스터에 혼란을 줄 수 있으며, 클러스터 오브젝트가 모든 관심 있는 큐 관리자에 자동으로 상태 업데이트를 보낸 이후 27일 간격으로 다시 수행됩니다. 대형 클러스터를 새로 고치면 클러스터의 성능 및 가용성에 영향을 줄 수 있음을 참조하십시오.

b) CLUSSDR 채널을 재시작하십시오.

(예: START CHANNEL 명령 사용).

c) CLUSRCVR 채널을 재시작하십시오.

5. 메시지 연관관계에 대해 인벤토리 애플리케이션을 검토하십시오.

   계속하기 전에 메시지 처리 순서에 따라 인벤토리 애플리케이션에 종속 항목이 없는지 확인하고 이 애플리케이션을 Toronto의 시스템에 설치하십시오.

6. 클러스터 큐 INVENTQ를 정의하십시오.

   이미 NEWYORK 큐 관리자가 호스팅하고 있는 INVENTQ 큐도 TORONTO가 호스팅합니다. 다음과 같이 이 큐를 TORONTO 큐 관리자에서 정의하십시오.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

결과

245 페이지의 그림 43에서는 이 태스크에서 설정되는 INVENTORY 클러스터를 보여줍니다.

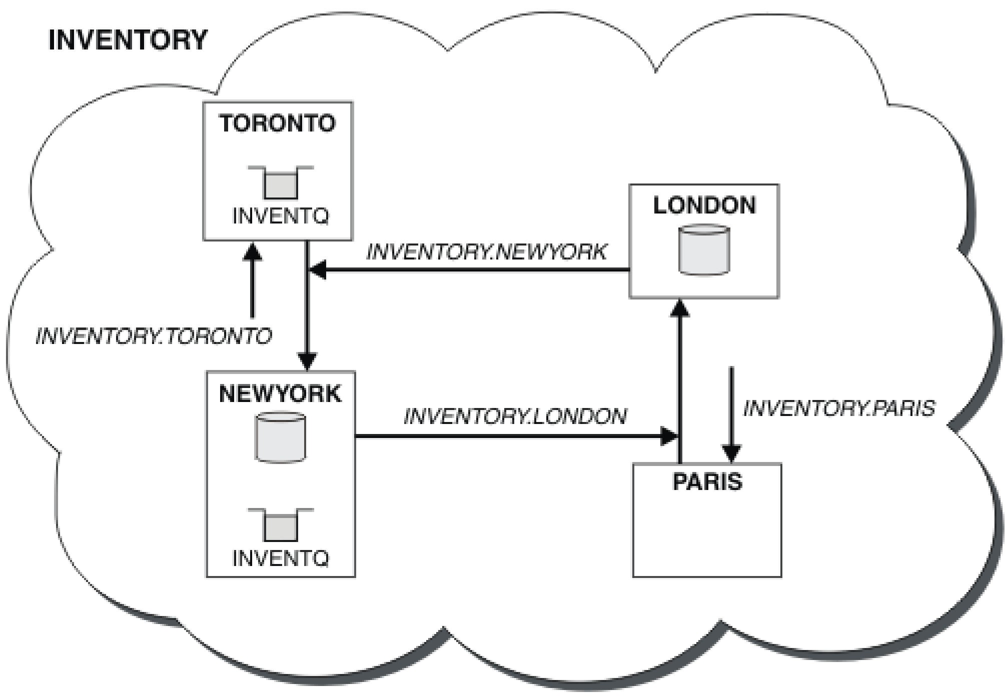


그림 *43.* 네 개의 큐 관리자가 있는 *INVENTORY* 클러스터

INVENTQ 큐 및 인벤토리 애플리케이션은 이제 클러스터에 있는 두 개의 큐 관리자에서 호스팅됩니다. 이 경우 두 큐 관리자의 가용성이 증가되고 메시지 처리 속도가 빨라지며 두 큐 관리자 사이에 워크로드를 분산시킬 수 있습니다. TORONTO 또는 NEWYORK에 의해 INVENTQ에 넣어진 메시지는 가능할 때마다 로컬 큐 관리자에서 인스턴스에 의해 핸들링됩니다. LONDON 또는 PARIS에 의해 넣어진 메시지는 워크로드 밸런싱이 이루어지도록 TORONTO 또는 NEWYORK에 교대로 라우트됩니다.

클러스터에 대한 이러한 수정은 사용자가 큐 관리자 NEWYORK, LONDON 및 PARIS에 대한 정의를 변경하지 않고도 수행할 수 있습니다. 이러한 큐 관리자의 전체 저장소는 메시지를 TORONTO의 INVENTQ에 송신하기 위해 필요한 정보로 자동 업데이트됩니다. 인벤토리 애플리케이션은 NEWYORK 또는 TORONTO 큐 관리자 중 하나가 사

용 불가능한 상태가 되더라도 계속해서 기능을 하며 충분한 용량을 가집니다. 인벤토리 애플리케이션은 두 위치 모두에서 호스팅되는 경우에도 올바르게 작동할 수 있어야 합니다.

이 태스크의 결과에서 볼 수 있듯이 둘 이상의 큐 관리자에서 실행되는 동일한 애플리케이션을 가질 수 있습니다. 고른 분산 워크로드를 위해 클러스터링할 수 있습니다.

애플리케이션은 두 위치에서 레코드를 처리하지 못할 수 있습니다. 예를 들어, 고객 계정 조회를 추가하고 LONDON 및 NEWYORK에서 실행 중인 애플리케이션을 업데이트하기로 결정한 것으로 가정하십시오. 계정 레코드는 한 위치에만 보유될 수 있습니다. 데이터 파티션 기술을 사용하여 요청 분배의 제어를 결정할 수 있습니다. 레코드의 분배를 분할할 수 있습니다. 레코드의 절반에 대해 배열할 수 있습니다. 예를 들어, 계정 번호 00000 - 49999는 LONDON에 보유됩니다. 나머지 반인 범위 50000 - 99999는 NEWYORK에 보유됩니다. 그런 다음 모든 메시지의 계정 필드를 조사하고 메시지를 적절한 큐 관리자에게 라우트하는 클러스터 워크로드 엑시트 프로그램을 작성할 수 있습니다.

다음에 수행할 작업

모든 정의를 마쳤으므로 아직 수행하지 않은 경우 IBM MQ for z/OS에서 채널 시작기를 시작하십시오. 모든 플랫폼의 큐 관리자 TORONTO에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 대기하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

z/OS 기존 클러스터에 큐 공유 그룹 추가

기존 클러스터에 z/OS의 큐 공유 그룹을 추가합니다.

시작하기 전에

참고:

1. 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.
2. 큐 공유 그룹은 IBM MQ for z/OS에서만 지원됩니다. 이 태스크는 다른 플랫폼에 적용할 수 없습니다.

시나리오:

- INVENTORY 클러스터가 226 페이지의 『새 클러스터 설정』에서 설명된 대로 설정되었습니다. 이 클러스터는 큐 관리자 LONDON 및 NEWYORK을 포함합니다.
- 이 클러스터에 큐 공유 그룹을 추가하려고 합니다. 이 그룹 QSGP는 세 큐 관리자 P1, P2 및 P3로 구성됩니다. 이들 큐 관리자는 INVENTQ 큐의 인스턴스를 공유하며 이는 P1에 의해 정의되어야 합니다.

이 태스크 정보

다음 단계에 따라 공유 큐를 호스팅하는 새 큐 관리자를 추가하십시오.

프로시저

1. 큐 관리자가 처음으로 참조하는 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 전체 저장소 중 하나를 참조해야 합니다. 큐 관리자는 전체 저장소에서 클러스터에 대한 정보를 수집하고 자체 부분 저장소를 빌드합니다. 전체 저장소 선택이 특별히 중요한 것은 아닙니다. 이 예에서는 NEWYORK을 선택하십시오. 일단 클러스터에 조인한 큐 공유 그룹은 전체 저장소 둘 다와 통신합니다.
2. CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의해야 합니다. P1, P2 및 P3에서 다음과 같이 정의하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.Pn) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(Pn.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for sharing queue manager')
   ```

클러스터 수신자 채널은 클러스터 INVENTORY에 있는 다른 큐 관리자로부터 메시지를 수신할 각 큐 관리자의 가용성을 알립니다.

3. 큐 공유 그룹에 대해 CLUSSDR 채널을 정의하십시오.

   클러스터의 모든 멤버는 메시지를 해당 첫 번째 전체 저장소에 송신할 수 있는 한 개의 클러스터 송신자 채널을 정의해야 합니다. 이 경우에서 NEWYORK을 선택했습니다. 큐 공유 그룹의 큐 관리자 중 한 큐 관리자에는 다음 그룹 정의가 필요합니다. 이 정의는 모든 큐 관리자가 클러스터 송신자 채널 정의를 가지도록 보장합니다.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) QSGDISP(GROUP)
DESCR('Cluster-sender channel to repository at NEWYORK')
```

4. 공유 큐를 정의하십시오.

   다음과 같이 P1에서 INVENTQ 큐를 정의하십시오.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) QSGDISP(SHARED) CFSTRUCT(STRUCTURE)
```

   새 큐 관리자에서 채널 시작기 및 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 청취하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

결과

247 페이지의 그림 44에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

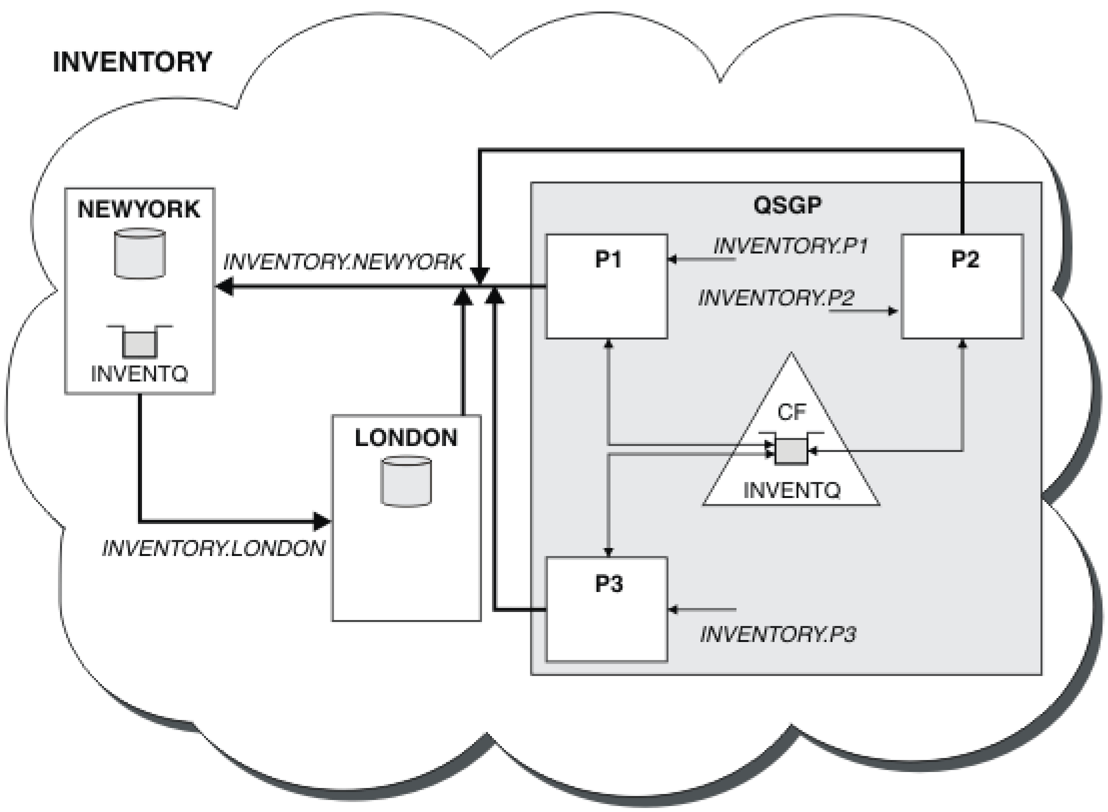


그림 *44.* 클러스터 및 큐 공유 그룹

이제 LONDON이 INVENTQ 큐에 넣은 메시지가 큐를 호스팅하는 것으로 알려진 네 큐 관리자에 교대로 라우트됩니다.

다음에 수행할 작업

큐 공유 그룹의 멤버가 클러스터 큐를 호스팅하도록 하는 것의 이점은 그룹의 어느 멤버나 요청에 응답할 수 있다는 것입니다. 이 경우 공유 큐에서 메시지를 수신한 후에 아마도 P1이 사용 불가능하게 될 것입니다. 큐 공유 그룹의 다른 멤버가 대신 응답할 수 있습니다.

다른 큐 관리자로 전체 저장소 이동

두 번째 저장소에 보유된 정보로부터 새 저장소를 빌드하여 한 큐 관리자에서 다른 큐 관리자로 전체 저장소를 이동합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되었습니다.
- 비즈니스상의 이유로 큐 관리자 LONDON에서 전체 저장소를 제거하고 이를 큐 관리자 PARIS의 전체 저장소로 대체하려고 합니다. NEWYORK 큐 관리자는 계속해서 전체 저장소를 보유하게 됩니다.

이 태스크 정보

다음 단계에 따라 다른 큐 관리자로 전체 저장소 이동하십시오.

프로시저

1. 전체 저장소 큐 관리자가 되도록 PARIS를 변경하십시오.

   PARIS에서 다음 명령을 실행하십시오.

   ```
   ALTER QMGR REPOS(INVENTORY)
   ```

2. PARIS에서 CLUSSDR 채널을 추가하십시오.

   PARIS에는 현재 LONDON을 가리키는 클러스터 송신자 채널이 있습니다. LONDON은 클러스터에 대한 전체 저장소를 더 이상 보유하지 않습니다. PARIS에는 현재 다른 전체 저장소를 보유하고 있는 NEWYORK을 가리키는 새 클러스터 송신자 채널이 있어야 합니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from PARIS to repository at NEWYORK')
   ```

3. NEWYORK에서 PARIS를 가리키는 CLUSSDR 채널을 정의하십시오.

   현재 NEWYORK에는 LONDON을 가리키는 클러스터 송신자 채널이 있습니다. 다른 전체 저장소가 PARIS로 이동했으므로 PARIS를 가리키는 새 클러스터 송신자 채널을 NEWYORK에서 추가해야 합니다.

   ```
   DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from NEWYORK to repository at PARIS')
   ```

   PARIS에 클러스터 송신자 채널을 추가하면 PARIS는 NEWYORK의 클러스터에 대해 학습합니다. 이는 NEWYORK의 정보를 사용하여 자체 전체 저장소를 빌드합니다.

4. 큐 관리자 PARIS에 이제 전체 저장소가 있는지 확인하십시오.

큐 관리자 PARIS가 큐 관리자 NEWYORK의 전체 저장소로부터 자체 전체 저장소를 빌드했는지 확인하십시오. 다음 명령을 실행하십시오.

```
DIS QCLUSTER(*) CLUSTER (INVENTORY)
DIS CLUSQMGR(*) CLUSTER (INVENTORY)
```

이러한 명령이 이 클러스터에서 NEWYORK에서와 같은 자원의 세부사항을 표시하는지 확인하십시오.

참고: 큐 관리자 NEWYORK이 사용 불가능한 경우 이 정보 빌드를 완료할 수 없습니다. 이 태스크가 완료될 때까지 다음 단계로 이동하지 마십시오.

5. LONDON의 큐 관리자 정의를 대체하십시오.

   마지막으로 클러스터에 대한 전체 저장소를 더 이상 보유하지 않도록 LONDON에서 이 큐 관리자를 대체하십시오. LONDON에서 다음 명령을 실행하십시오.

   ```
   ALTER QMGR REPOS(' ')
   ```

   큐 관리자가 더 이상 클러스터 정보를 수신하지 않습니다. 30일 후에는 해당 전체 저장소에 저장되는 정보가 만료됩니다. 이제 큐 관리자 LONDON이 자체 부분 저장소를 빌드합니다.

6. 미해결 정의를 제거하거나 변경하십시오.

   클러스터의 새 배열이 예상대로 작동하는 것으로 확인되면 더 이상 정확하지 않은 수동으로 정의된 CLUSSDR 정의를 제거하거나 변경하십시오.

   - PARIS 큐 관리자에서 LONDON에 대한 클러스터 송신자 채널을 중지하고 삭제한 다음 이 클러스터가 자동 채널을 다시 사용할 수 있도록 채널 시작 명령을 발행해야 합니다.

     ```
     STOP CHANNEL(INVENTORY.LONDON)
     DELETE CHANNEL(INVENTORY.LONDON)
     START CHANNEL(INVENTORY.LONDON)
     ```

   - NEWYORK 큐 관리자에서 LONDON에 대한 클러스터 송신자 채널을 중지하고 삭제한 다음 이 클러스터가 자동 채널을 다시 사용할 수 있도록 채널 시작 명령을 발행해야 합니다.

     ```
     STOP CHANNEL(INVENTORY.LONDON)
     DELETE CHANNEL(INVENTORY.LONDON)
     START CHANNEL(INVENTORY.LONDON)
     ```

   - 클러스터 내 모든 큐 관리자에서 LONDON을 가리키는 다른 모든 수동으로 정의된 클러스터 송신자 채널을 NEWYORK 또는 PARIS를 가리키는 채널로 바꾸십시오. 채널을 삭제한 후에 클러스터가 자동 채널을 다시 사용할 수 있도록 항상 **start channel** 명령을 발행하십시오. 이 작은 예에는 다른 클러스터 송신자 채널이 없습니다. 사용자가 잊어버린 클러스터 송신자 채널이 있는지 여부를 검사하려면 TYPE(CLUSSDR)을 지정하여 각 큐 관리자에서 DISPLAY CHANNEL 명령을 발행하십시오. 예를 들면 다음과 같습니다.

     ```
     DISPLAY CHANNEL(*) TYPE(CLUSSDR)
     ```

   전체 저장소를 LONDON에서 PARIS로 이동한 후에 최대한 빨리 이 태스크를 수행하는 것이 중요합니다. 이 태스크를 수행하기 전에 INVENTORY.LONDON이라는 이름의 수동으로 정의된 CLUSSDR 채널이 있는 큐 관리자가 이 채널을 사용하여 정보에 대한 요청을 송신할 수 있습니다.

   LONDON이 전체 저장소가 되기 위해 중지된 후에 그러한 요청을 수신하면 해당 큐 관리자 오류 로그에 오류 메시지를 기록합니다. 다음 예는 LONDON에 표시될 수 있는 오류 메시지를 보여줍니다.

   - AMQ9428: Unexpected publication of a cluster queue object received
   - AMQ9432: Query received by a non-repository queue manager

큐 관리자 LONDON은 더 이상 전체 저장소가 아니므로 정보 요청에 응답하지 않습니다. LONDON에서 정보를 요청하는 큐 관리자는 수동으로 정의된 CLUSSDR 정의가 PARIS를 가리키도록 수정될 때까지 클러스터 정보에 대해 NEWYORK에 의존해야 합니다. 이 상황은 장기적으로는 유효한 구성으로 허용되지 않아야 합니다.

결과

250 페이지의 그림 45에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

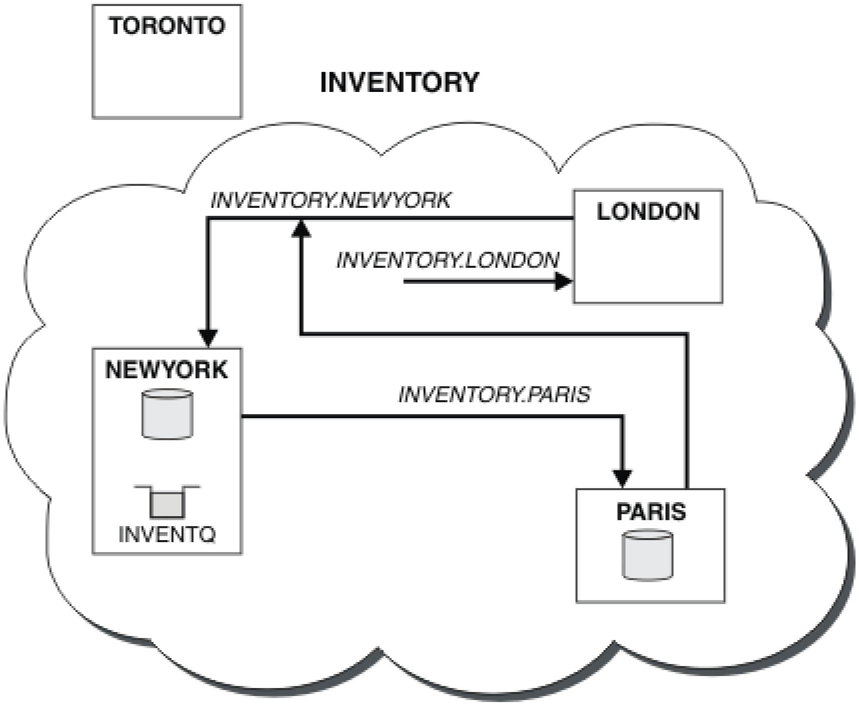


그림 *45.* 전체 저장소가 *PARIS*로 이동된 *INVENTORY* 클러스터

클러스터에서 통신 설정

전달할 메시지가 있을 때 통신 채널을 시작하기 위한 채널 시작기가 필요합니다. 채널 리스너는 메시지를 수신하기 위해 채널의 다른 한 측이 시작할 때까지 대기합니다.

시작하기 전에

클러스터 내 큐 관리자 간 통신을 설정하려면 지원되는 통신 프로토콜 중 하나를 사용하여 링크를 구성하십시오. 모든 플랫폼에서 지원되는 프로토콜은 TCP 또는 LU 6.2이고 Windows 시스템에서 지원되는 프로토콜은 NetBIOS 또는 SPX입니다. 분산 큐잉에서와 마찬가지로 이 구성의 일부로 채널 시작기 및 채널 리스너가 필요합니다.

이 태스크 정보

모든 클러스터 큐 관리자는 시스템 정의 시작 큐인 SYSTEM.CHANNEL.INITQ를 모니터하기 위한 채널 시작기가 필요합니다. SYSTEM.CHANNEL.INITQ는 클러스터 전송 큐를 포함한 모든 전송 큐에 대한 시작 큐입니다.

각 큐 관리자에는 채널 리스너가 있어야 합니다. 채널 리스너 프로그램은 수신되는 네트워크 요청을 대기하고 필요할 때 적절한 수신자 채널을 시작합니다. 채널 리스너의 구현은 플랫폼별로 다르지만 일부 공통 기능이 있습니다. 모든 IBM MQ 플랫폼에서 START LISTENER 명령을 사용하여 리스너를 시작할 수 있습니다. IBM MQ for IBM i, Windows, UNIX and Linux 시스템에서 큐 관리자와 동시에 자동으로 리스너를 시작할 수 있습니다. 리스너를 자동으로 시작하려면 LISTENER 오브젝트의 CONTROL 속성을 QMGR 또는 STARTONLY로 설정하십시오.

z/OS 비공유 리스너 포트(INDISP(QMGR))는 z/OS의 CLUSRCVR 채널 및 CLUSSDR 채널에서 z/OS(으)로 사용되어야 합니다.

프로시저

1. 채널 시작기를 시작하십시오.

   - z/OS

     IBM MQ for z/OS
     각 큐 관리자에는 한 개의 채널 시작기가 있으며 이 채널 시작기는 개별 주소 공간으로 실행됩니다. 채널 시작기는 큐 관리자 시동의 일부로 발행하는 **MQSC** `START CHINIT` 명령을 사용하여 시작합니다.

   - ULW

     IBM MQ for UNIX, Linux, and Windows
     큐 관리자를 시작할 때 큐 관리자 속성 SCHINIT가 QMGR로 설정되면 채널 시작기가 자동으로 시작됩니다. 그렇지 않은 경우 **runmqsc** `START CHINIT` 명령 또는 **runmqchi** 제어 명령을 사용하여 시작할 수 있습니다.

   - IBM i

     IBM MQ for IBM i
     큐 관리자를 시작할 때 큐 관리자 속성 SCHINIT가 QMGR로 설정되면 채널 시작기가 자동으로 시작됩니다. 그렇지 않은 경우 **runmqsc** `START CHINIT` 명령 또는 **runmqchi** 제어 명령을 사용하여 시작할 수 있습니다.

2. 채널 리스너를 시작하십시오.

   - z/OS

     IBM MQ for z/OS
     IBM MQ에서 제공하는 채널 리스너 프로그램을 사용하십시오. IBM MQ 채널 리스너를 시작하려면 **MQSC** 명령인 `START LISTENER`를 사용하십시오. 이 명령은 채널 시작기 시동의 일부로 발행합니다. 예를 들면, 다음과 같습니다.

     ```
     START LISTENER PORT(1414) TRPTYPE(TCP)
     ```

     또는:

     ```
     START LISTENER LUNAME(LONDON.LUNAME) TRPTYPE(LU62)
     ```

     큐 공유 그룹의 멤버는 각 큐 관리자의 리스너 대신 공유 리스너를 사용할 수 있습니다. 공유 리스너를 클러스터와 사용하지 마십시오. 특히 CLUSRCVR 채널의 CONNAME을 큐 공유 그룹의 공유 리스너 주소로 작성하지 마십시오. 이를 수행할 경우 큐 관리자는 정의가 없는 큐에 대한 메시지를 수신할 수 있습니다.

   - IBM i

     IBM MQ for IBM i
     IBM MQ에서 제공하는 채널 리스너 프로그램을 사용하십시오. IBM MQ 채널 리스너를 시작하려면 **CL** 명령인 STRMQMLSR을 사용하십시오. 예를 들면, 다음과 같습니다.

     ```
     STRMQMLSR MQMNAME(QM1) PORT(1414)
     ```

   - Windows

     IBM MQ for Windows
     IBM MQ에서 제공되는 채널 리스너 또는 운영 체제에서 제공되는 기능을 사용하십시오.

     IBM MQ 채널 리스너를 시작하려면 RUNMQLSR 명령을 사용하십시오. 예를 들면, 다음과 같습니다.

     ```
     RUNMQLSR -t tcp -p 1414 -m QM1
     ```

   - Linux UNIX

UNIX and Linux에서 IBM MQ

IBM MQ에서 제공되는 채널 리스너 프로그램 또는 운영 체제에서 제공되는 기능(예: TCP 통신을 위한 **inetd**)을 사용하십시오.

IBM MQ 채널 리스너를 시작하려면 **runmqlsr** 명령을 사용하십시오. 예를 들면, 다음과 같습니다.

```
runmqlsr -t tcp -p 1414 -m QM1
```

inetd를 사용하여 채널을 시작하려면 다음 두 파일을 구성하십시오.

a. `/etc/services` 파일을 편집하십시오. superuser 또는 루트로 로그인해야 합니다. 다음 행이 파일에 없는 경우 이 행을 다음과 같이 추가하십시오.

   ```
   MQSeries    1414/tcp   # WebSphere MQ channel listener
   ```

   여기서 1414은(는) IBM MQ에 필요한 포트 번호입니다. 포트 번호를 변경할 수 있지만 포트 번호는 송신 측에서 지정된 포트 번호와 일치해야 합니다.

b. `/etc/inetd.conf` 파일을 편집하십시오. 해당 파일에 다음 행이 없는 경우에는 다음과 같이 추가하십시오.

   ```
   MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta
   -m queue.manager.name
   ```

   여기서, *MQ_INSTALLATION_PATH*는 IBM MQ가 설치되는 상위 레벨 디렉토리로 바뀝니다.

업데이트는 **inetd**가 구성 파일을 다시 읽은 후에 활성이 됩니다. 루트 사용자 ID에서 다음 명령을 실행하십시오.

AIX의 경우:

```
refresh -s inetd
```

HP-UX의 경우:

```
inetd -c
```

Solaris 또는 Linux의 경우:

a. 다음 명령을 사용하여 **inetd**의 프로세스 ID를 찾으십시오.

   ```
   ps -ef | grep inetd
   ```

b. 다음과 같이 적절한 명령을 실행하십시오.

   - Solaris 9 및 Linux의 경우:

     ```
     kill -1 inetd processid
     ```

   - Solaris 10 이상 버전의 경우:

     ```
     inetconv
     ```

기존 네트워크를 클러스터로 변환

기존 분산 큐잉 네트워크를 클러스터로 변환하고 용량을 늘리기 위해 추가 큐 관리자를 추가합니다.

시작하기 전에

226 페이지의 『새 클러스터 설정』에서 248 페이지의 『다른 큐 관리자로 전체 저장소 이동』까지 새 클러스터를 작성하고 확장했습니다. 다음 두 태스크에서는 큐 관리자의 기존 네트워크를 클러스터로 변환하는 접근법에 대해 알아봅니다.

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- 체인 저장소의 전국 분기를 연결하는 IBM MQ 네트워크가 이미 준비되어 있습니다. 이 네트워크는 허브 및 스포크 구조를 가지며 모든 큐 관리자는 하나의 중앙 큐 관리자에 연결됩니다. 중앙 큐 관리자는 인벤토리 애플리케이션이 실행되는 시스템에 있습니다. 애플리케이션은 각 큐 관리자가 리모트 큐 정의를 가지는 `INVENTQ` 큐에 메시지가 도착하면 구동됩니다.

  이 네트워크는 253 페이지의 그림 46에서 설명됩니다.

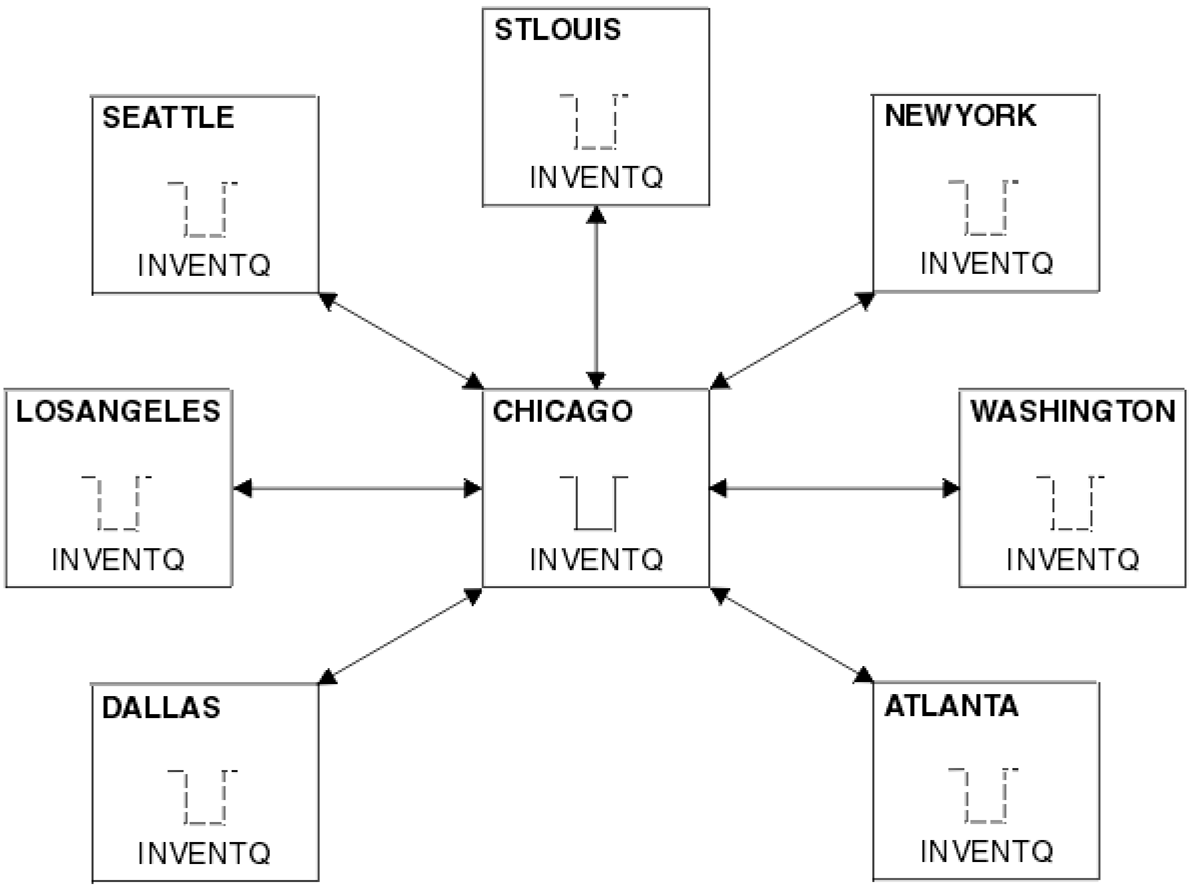


  그림 *46.* 허브 및 스포크 네트워크

- 관리를 쉽게 하기 위해 이 네트워크를 클러스터로 변환하고 중앙 사이트에서 워크로드를 공유할 다른 큐 관리자를 작성합니다.

  이 클러스터 이름은 `CHNSTORE`입니다.

  참고: 최대 20자 길이를 초과하지 않는 *cluster-name.queue_manager_name* 형식으로 된 이름(예: `CHNSTORE.WASHINGTON`)을 사용하여 클러스터 수신자 채널 이름을 작성할 수 있도록 클러스터 이름 `CHNSTORE`가 선택되었습니다.

- 중앙 큐 관리자 둘 다 전체 저장소를 호스팅하고 인벤토리 애플리케이션에서 액세스 가능합니다.
- 인벤토리 애플리케이션은 중앙 큐 관리자 중 하나가 호스팅하는 `INVENTQ` 큐에 메시지가 도착하면 구동됩니다.

- 인벤토리 애플리케이션은 둘 이상의 큐 관리자에 의해 액세스 가능하고 병렬로 실행되는 유일한 애플리케이션이어야 합니다. 다른 모든 애플리케이션은 전과 같이 계속해서 실행됩니다.
- 모든 분기는 두 중앙 큐 관리자에 대한 네트워크 연결성을 가집니다.
- 네트워크 프로토콜은 TCP입니다.

이 태스크 정보

다음 단계에 따라 기존 네트워크를 클러스터로 변환하십시오.

프로시저

1. 메시지 연관관계에 대해 인벤토리 애플리케이션을 검토하십시오.

   진행하기 전에 애플리케이션이 메시지 연관관계를 핸들링할 수 있는지 확인하십시오. 메시지 연관관계는 두 애플리케이션 간 교환되는 대화형 메시지 간 관계이며 이 관계에서는 메시지가 특정 큐 관리자 또는 특정 순서로 처리되어야 합니다. 메시지 연관관계에 대한 자세한 정보는 324 페이지의 『메시지 연관관계 핸들링』의 내용을 참조하십시오.
2. 전체 저장소 큐 관리자가 되도록 두 중앙 큐 관리자를 대체하십시오.

   이 두 큐 관리자 CHICAGO 및 CHICAGO2는 이 네트워크의 허브에 있습니다. 이러한 체인 저장소 클러스터와 연관된 모든 활동을 이 두 큐 관리자에 집중하기로 결정했습니다. 인벤토리 애플리케이션 및 INVENTQ 큐 외에도 이러한 큐 관리자가 클러스터에 대한 두 개의 전체 저장소를 호스팅하게 하려고 합니다. 두 큐 관리자 각각에서 다음 명령을 실행하십시오.

   ```
   ALTER QMGR REPOS(CHNSTORE)
   ```
3. 각 큐 관리자에서 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 각 큐 관리자에서 클러스터 수신자 채널 및 클러스터 송신자 채널을 정의하십시오. 어떤 채널을 처음 정의할지는 중요하지 않습니다.

   CLUSRCVR 정의를 작성하여 각 큐 관리자, 해당 네트워크 주소 및 기타 정보를 클러스터에 알리십시오. 예를 들어, 큐 관리자 ATLANTA의 경우:

   ```
   DEFINE CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(ATLANTA.CHSTORE.COM) CLUSTER(CHNSTORE)
   DESCR('Cluster-receiver channel')
   ```
4. 각 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   각 큐 관리자에서 CLUSSDR 정의를 작성하여 해당 큐 관리자를 전체 저장소 큐 관리자 중 어느 하나에 링크하십시오. 예를 들어, ATLANTA를 CHICAGO2에 링크할 수 있습니다.

   ```
   DEFINE CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(CHICAGO2.CHSTORE.COM) CLUSTER(CHNSTORE)
   DESCR('Cluster-sender channel to repository queue manager')
   ```
5. CHICAGO2에서 인벤토리 애플리케이션을 설치하십시오.

   큐 관리자 CHICAGO에 이미 인벤토리 애플리케이션이 있습니다. 이제 큐 관리자 CHICAGO2에서 이 애플리케이션의 사본을 작성해야 합니다.
6. 중앙 큐 관리자에서 INVENTQ 큐를 정의하십시오.

   CHICAGO에서 큐 INVENTQ의 로컬 큐 정의를 수정하여 이 큐를 클러스터에서 사용 가능하게 하십시오. 다음 명령을 실행하십시오.

   ```
   ALTER QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
   ```

CHICAGO2에서 같은 큐에 대한 정의를 작성하십시오.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

z/OS에서 **CSQUTIL**의 COMMAND 함수 MAKEDEF 옵션을 사용하여 CHICAGO에서 INVENTQ의 CHICAGO2에 대한 정확한 사본을 작성할 수 있습니다.

이러한 정의를 작성하면 메시지가 CHICAGO 및 CHICAGO2에 있는 전체 저장소에 송신되어 해당 정보가 업데이트됩니다. INVENTQ에 메시지를 넣을 때 큐 관리자는 전체 저장소에서 메시지의 목적지를 선택할 수 있는지 확인합니다.

7. 클러스터 변경사항이 전파되었는지 확인하십시오.

   이전 단계에서 사용자가 작성한 정의가 클러스터 전체에 전파되었는지 확인하십시오. 전체 저장소 큐 관리자에서 다음 명령을 실행하십시오.

   ```
   DIS QCLUSTER(INVENTQ)
   ```

상호 연결된 새 클러스터 추가

기존 클러스터와 일부 큐 관리자를 공유하는 새 클러스터를 추가합니다.

시작하기 전에

참고:

1. 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.
2. 이 태스크를 시작하기 전에 큐 이름 충돌이 있는지 확인하고 그 결과를 파악합니다. 진행하기 위해 큐의 이름을 바꾸거나 큐 알리어스를 설정해야 할 수 있습니다.

시나리오:

- IBM MQ 클러스터가 252 페이지의 『기존 네트워크를 클러스터로 변환』에서 설명된 대로 설정되었습니다.
- MAILORDER라고 하는 새 클러스터가 구현되어야 합니다. 이 클러스터는 CHNSTORE 클러스터에 있는 네 큐 관리자 CHICAGO, CHICAGO2, SEATTLE 및 ATLANTA와 두 추가 큐 관리자 HARTFORD 및 OMAHA로 구성됩니다. MAILORDER 애플리케이션은 큐 관리자 OMAHA에 연결된 Omaha에 있는 시스템에서 실행됩니다. 이 애플리케이션은 클러스터 내 다른 큐 관리자에 의해 구동되어 메시지를 MORDERQ 큐에 넣습니다.
- MAILORDER 클러스터의 전체 저장소는 두 큐 관리자 CHICAGO 및 CHICAGO2에서 유지보수됩니다.
- 네트워크 프로토콜은 TCP입니다.

이 태스크 정보

다음 단계에 따라 상호 연결된 새 클러스터를 추가하십시오.

프로시저

1. 클러스터 이름의 이름 목록을 작성하십시오.

   이제 CHICAGO 및 CHICAGO2의 전체 저장소 큐 관리자는 CHNSTORE 및 MAILORDER 클러스터 둘 다에 대한 전체 저장소를 보유합니다. 먼저 클러스터의 이름을 포함하는 이름 목록을 작성하십시오. 다음과 같이 CHICAGO 및 CHICAGO2에서 이름 목록을 정의하십시오.

   ```
   DEFINE NAMELIST(CHAINMAIL)
   DESCR('List of cluster names')
   NAMES(CHNSTORE, MAILORDER)
   ```

2. 두 큐 관리자 정의를 대체하십시오.

이제 CHICAGO 및 CHICAGO2에서 두 큐 관리자 정의를 대체하십시오. 현재 이러한 정의는 큐 관리자가 클러스터 CHNSTORE에 대한 전체 저장소를 보유함을 표시합니다. 큐 관리자가 CHAINMAIL 이름 목록에 나열된 모든 클러스터에 대한 전체 저장소를 보유함을 표시하도록 해당 정의를 변경하십시오. 다음과 같이 CHICAGO 및 CHICAGO2 큐 관리자 정의를 대체하십시오.

```
ALTER QMGR REPOS(' ') REPOSNL(CHAINMAIL)
```

3. CHICAGO 및 CHICAGO2에서 CLUSRCVR 채널을 대체하십시오.

   CHICAGO 및 CHICAGO2의 CLUSRCVR 채널 정의는 채널이 CHNSTORE 클러스터에서 사용 가능함을 표시합니다. CHAINMAIL 이름 목록에 나열된 모든 클러스터에서 채널을 사용할 수 있음을 표시하도록 클러스터 수신자 정의를 변경해야 합니다. 다음과 같이 CHICAGO에서 클러스터 수신자 정의를 변경하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

   CHICAGO2에서 다음 명령을 입력하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSRCVR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

4. CHICAGO 및 CHICAGO2에서 CLUSSDR 채널을 대체하십시오.

   이름 목록을 추가하려면 두 CLUSSDR 채널 정의를 변경하십시오. CHICAGO에서 다음 명령을 입력하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

   CHICAGO2에서 다음 명령을 입력하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

5. SEATTLE 및 ATLANTA에서 이름 목록을 작성하십시오.

   SEATTLE 및 ATLANTA가 둘 이상의 클러스터의 멤버가 되기 때문에 클러스터의 이름을 포함하는 이름 목록을 작성해야 합니다. 다음과 같이 SEATTLE 및 ATLANTA에서 이름 목록을 정의하십시오.

   ```
   DEFINE NAMELIST(CHAINMAIL)
   DESCR('List of cluster names')
   NAMES(CHNSTORE, MAILORDER)
   ```

6. SEATTLE 및 ATLANTA에서 CLUSRCVR 채널을 대체하십시오.

   SEATTLE 및 ATLANTA의 CLUSRCVR 채널 정의는 채널이 CHNSTORE 클러스터에서 사용 가능함을 표시합니다. CHAINMAIL 이름 목록에 나열된 모든 클러스터에서 클러스터 수신 채널을 사용할 수 있음을 표시하도록 이 채널의 정의를 변경하십시오. SEATTLE에서 다음 명령을 입력하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.SEATTLE) CHLTYPE(CLUSRCVR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

   ATLANTA에서 다음 명령을 입력하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR)
   CLUSTER(' ') CLUSNL(CHAINMAIL)
   ```

7. SEATTLE 및 ATLANTA에서 CLUSSDR 채널을 대체하십시오.

이름 목록을 추가하려면 두 CLUSSDR 채널 정의를 변경하십시오. SEATTLE에서 다음 명령을 입력하십시오.

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

ATLANTA에서 다음 명령을 입력하십시오.

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

8. HARTFORD 및 OMAHA에서 CLUSRCVR 및 CLUSSDR 채널을 정의하십시오.

   두 새 큐 관리자 HARTFORD 및 OMAHA에서 클러스터 수신자 및 클러스터 송신자 채널을 정의하십시오. 정의를 작성하는 순서는 문제가 되지 않습니다. HARTFORD에서 다음을 입력하십시오.

```
DEFINE CHANNEL(MAILORDER.HARTFORD) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(HARTFORD.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for HARTFORD')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from HARTFORD to repository at CHICAGO')
```

   OMAHA에서 다음을 입력하십시오.

```
DEFINE CHANNEL(MAILORDER.OMAHA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(OMAHA.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for OMAHA')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from OMAHA to repository at CHICAGO')
```

9. OMAHA에서 MORDERQ 큐를 정의하십시오.

   이 태스크를 완료하는 첫 번째 단계는 큐 관리자 OMAHA에서 MORDERQ 큐를 정의하는 것입니다. OMAHA에서 다음을 입력하십시오.

```
DEFINE QLOCAL(MORDERQ) CLUSTER(MAILORDER)
```

10. 클러스터 변경사항이 전파되었는지 확인하십시오.

    이전 단계에서 사용자가 작성한 정의가 클러스터 전체에 전파되었는지 확인하십시오. 전체 저장소 큐 관리자에서 다음 명령을 실행하십시오.

```
DIS QCLUSTER (MORDERQ)
DIS CLUSQMGR
```

11.

결과

이 태스크에서 설정되는 클러스터는 258 페이지의 그림 47에 표시됩니다.

이제 두 개의 중첩 클러스터가 있습니다. 두 클러스터 모두에 대한 전체 저장소는 CHICAGO 및 CHICAGO2에 보유됩니다. OMAHA에서 실행되는 메일 오더 애플리케이션은 CHICAGO에서 실행되는 인벤토리 애플리케이션과는 독립적입니다. 그러나 CHNSTORE 클러스터에 있는 일부 큐 관리자는 MAILORDER 클러스터에도 있으므로 메시지를 애플리케이션에 송신할 수 있습니다. 이 태스크를 수행하여 두 클러스터를 중첩하기 전에 큐 이름 충돌 가능성을 확인하십시오.

CHNSTORE 클러스터의 NEWYORK에서와 MAILORDER 클러스터의 OMAHA에는 ACCOUNTQ라고 하는 큐가 있다고 가정하십시오. 클러스터를 중첩한 다음 SEATTLE의 애플리케이션이 메시지를 ACCOUNTQ 큐에 넣는 경우 이 메시지는 ACCOUNTQ의 인스턴스 중 하나로 이동할 수 있습니다.

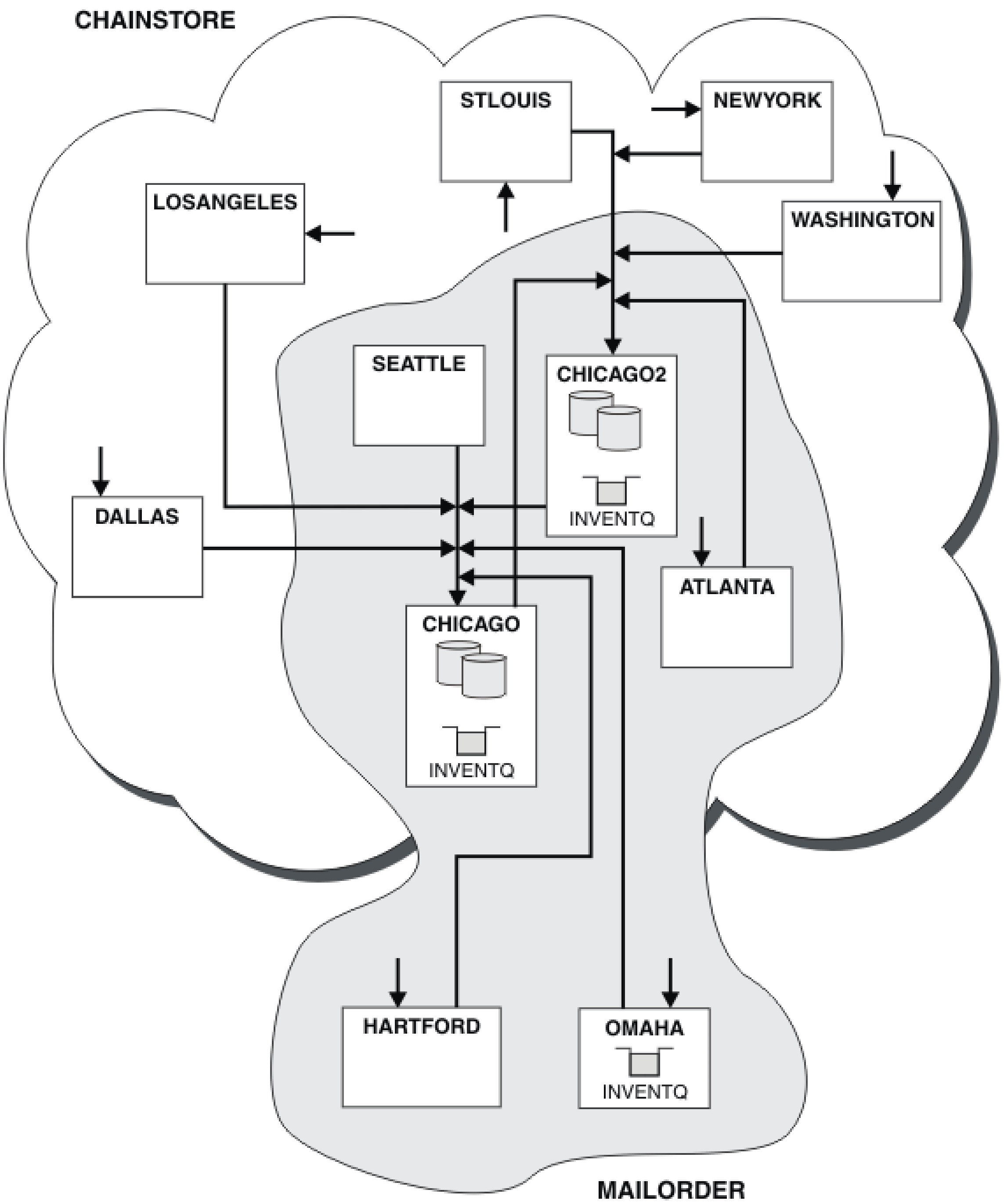


그림 *47.* 상호 연결된 클러스터

다음에 수행할 작업

CHNSTORE라고 하는 한 개의 대형 클러스터를 구성하기 위해 MAILORDER 클러스터와 CHNSTORE 클러스터를 병합하기로 결정했다고 가정하십시오.

CHICAGO와 CHICAGO2가 전체 저장소를 보유하도록 MAILORDER 클러스터와 CHNSTORE 클러스터를 병합하려면 다음을 수행하십시오.

- 이름 목록(CHAINMAIL)을 지정하는 REPOSNL 속성을 제거하고 이를 클러스터 이름(CHNSTORE)을 지정하는 REPOS 속성으로 대체하여 CHICAGO 및 CHICAGO2에 대한 큐 관리자 정의를 대체하십시오. 예를 들면 다음과 같습니다.

```
ALTER QMGR(CHICAGO) REPOSNL(' ') REPOS(CHNSTORE)
```

- MAILORDER 클러스터의 각 큐 관리자에서 모든 채널 정의 및 큐 정의를 대체하여 CLUSTER 속성의 값을 MAILORDER에서 CHNSTORE로 변경하십시오. 예를 들어, HARTFORD에 다음을 입력하십시오.

```
ALTER CHANNEL(MAILORDER.HARTFORD) CLUSTER(CHNSTORE)
```

  OMAHA에서 다음을 입력하십시오.

```
ALTER QLOCAL(MORDERQ) CLUSTER(CHNSTORE)
```

- 클러스터 이름 목록 CHAINMAIL을 지정하는 모든 정의 즉, CHICAGO, CHICAGO2, SEATTLE 및 ATLANTA의 CLUSRCVR 및 CLUSSDR 채널 정의를 대체하여 CHNSTORE 클러스터를 대신 지정하십시오.

이 예에서 이름 목록 사용의 장점을 확인할 수 있습니다. CHICAGO 및 CHICAGO2에 대한 큐 관리자 정의를 대체하는 대신 이름 목록 CHAINMAIL의 값을 대체할 수 있습니다. 마찬가지로 CHICAGO, CHICAGO2, SEATTLE 및 ATLANTA의 CLUSRCVR 및 CLUSSDR 채널 정의를 대체하는 대신 이름 목록을 대체하여 필요한 결과를 얻을 수 있습니다.

관련 태스크

클러스터 네트워크 제거
네트워크에서 클러스터를 제거하고 분산 큐잉 구성을 복원합니다.

클러스터 네트워크 제거

네트워크에서 클러스터를 제거하고 분산 큐잉 구성을 복원합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- IBM MQ 클러스터가 252 페이지의 『기존 네트워크를 클러스터로 변환』에서 설명된 대로 설정되었습니다.
- 이 클러스터를 이제 시스템에서 제거하려고 합니다. 큐 관리자의 네트워크는 클러스터가 구현되기 전과 같이 계속해서 작동합니다.

이 태스크 정보

다음 단계에 따라 클러스터 네트워크를 제거하십시오.

프로시저

1. CHNSTORE 클러스터에서 클러스터 큐를 제거하십시오.

   CHICAGO 및 CHICAGO2 둘 다에서 큐 INVENTQ에 대한 로컬 큐 정의를 수정하여 클러스터에서 큐를 제거하십시오. 다음 명령을 실행하십시오.

   ```
   ALTER QLOCAL(INVENTQ) CLUSTER(' ')
   ```

   큐를 대체하면 전체 저장소의 정보가 업데이트되고 클러스터 전체에 전파됩니다. MQOO_BIND_NOT_FIXED를 사용하는 활성 애플리케이션과 큐가 DEFBIND(NOTFIXED)로 정의된 MQOO_BIND_AS_Q_DEF를 사용하는 애플리케이션이 다음으로 시도된 MQPUT 또는 MQPUT1 호출에 실패합니다. 이유 코드 MQRC_UNKNOWN_OBJECT_NAME이 리턴됩니다.

단계 1을 처음 수행할 필요는 없지만 대신 단계 4 후에 단계 1을 수행하십시오.

2. 클러스터 큐에 대한 액세스가 있는 모든 애플리케이션을 중지하십시오.

   클러스터 큐에 대한 액세스가 있는 모든 애플리케이션을 중지하십시오. 그렇지 않으면 5단계에서 클러스터를 새로 고칠 때 일부 클러스터 정보가 로컬 큐 관리자에 남아 있을 수 있습니다. 모든 애플리케이션이 중지되고 클러스터 채널의 연결이 끊어진 경우 이 정보가 제거됩니다.

3. 전체 저장소 큐 관리자에서 저장소 속성을 제거하십시오.

   CHICAGO 및 CHICAGO2 둘 다에서 저장소 속성을 제거하도록 큐 관리자 정의를 수정하십시오. 이를 수행하려면 다음 명령을 실행하십시오.

   ```
   ALTER QMGR REPOS(' ')
   ```

   큐 관리자는 클러스터 내 다른 큐 관리자에게 더 이상 전체 저장소를 보유하지 않음을 알립니다. 다른 큐 관리자가 이 정보를 수신하면 전체 저장소가 종료되었음을 나타내는 메시지가 표시됩니다. 또한 CHNSTORE 클러스터에 대해 사용 가능한 저장소가 없음을 나타내는 하나 이상의 메시지도 표시됩니다.

4. 클러스터 채널을 제거하십시오.

   CHICAGO에서 클러스터 채널을 제거하십시오.

   ```
   ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) CLUSTER(' ')
   ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR) CLUSTER(' ')
   ```

   참고: CLUSSDR 명령을 먼저 발행한 다음 CLUSRCVR 명령을 발행하는 것이 중요합니다. CLUSRCVR 명령을 먼저 발행한 다음 CLUSSDR 명령을 발행해서는 안됩니다. 이렇게 할 경우 상태가 STOPPED인 인다우트(in-doubt) 채널이 작성됩니다. 그러면 START CHANNEL 명령을 발행하여 중지된 채널을 복구해야 합니다. 예를 들면, 다음과 같습니다. START CHANNEL(CHNSTORE.CHICAGO).

   CHNSTORE 클러스터에 대한 저장소가 없음을 나타내는 메시지가 표시됩니다.

   단계 1에서 설명한 대로 클러스터 큐를 제거하지 않은 경우 지금 제거하십시오.

5. 클러스터 채널을 중지하십시오.

   CHICAGO에서 다음 명령을 사용하여 클러스터 채널을 중지하십시오.

   ```
   STOP CHANNEL(CHNSTORE.CHICAGO2)
   STOP CHANNEL(CHNSTORE.CHICAGO)
   ```

6. 클러스터에 있는 각 큐 관리자에 대해 단계 4와 5를 반복하십시오.
7. 클러스터 채널을 중지한 다음 각 큐 관리자에서 클러스터 채널 및 클러스터 큐에 대한 모든 정의를 제거하십시오.
8. 옵션: 큐 관리자가 보유하는 캐시된 클러스터 정보를 지우십시오.

   큐 관리자가 더 이상 클러스터의 멤버가 아니더라도 큐 관리자 각각은 클러스터에 대한 정보의 캐시된 사본을 보유합니다. 이 데이터를 제거하려는 경우 287 페이지의 『큐 관리자를 해당 클러스터 전 상태로 복원』 태스크를 참조하십시오.

9. INVENTQ에 대한 리모트 큐 정의를 바꾸십시오.

   네트워크가 계속해서 작동할 수 있도록 모든 큐 관리자에서 INVENTQ에 대한 리모트 큐 정의를 바꾸십시오.

10. 클러스터를 정리하십시오.

    더 이상 필요하지 않은 큐 또는 채널 정의를 삭제하십시오.

관련 태스크

상호 연결된 새 클러스터 추가
기존 클러스터와 일부 큐 관리자를 공유하는 새 클러스터를 추가합니다.

게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성

태스크의 지시사항에 따라 게이트웨이 큐 관리자가 포함된 중첩 클러스터를 구성합니다. 한 애플리케이션에 대한 메시지를 클러스터 내 다른 애플리케이션에 대한 메시지로부터 격리하는 다음 예에 대한 시작점으로 이 클러스터를 사용합니다.

이 태스크 정보

클러스터 메시지 트래픽 격리를 설명하는 데 사용되는 클러스터 구성 예가 261 페이지의 그림 48에 표시되어 있습니다. 이 예는 클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리에서 설명됩니다.

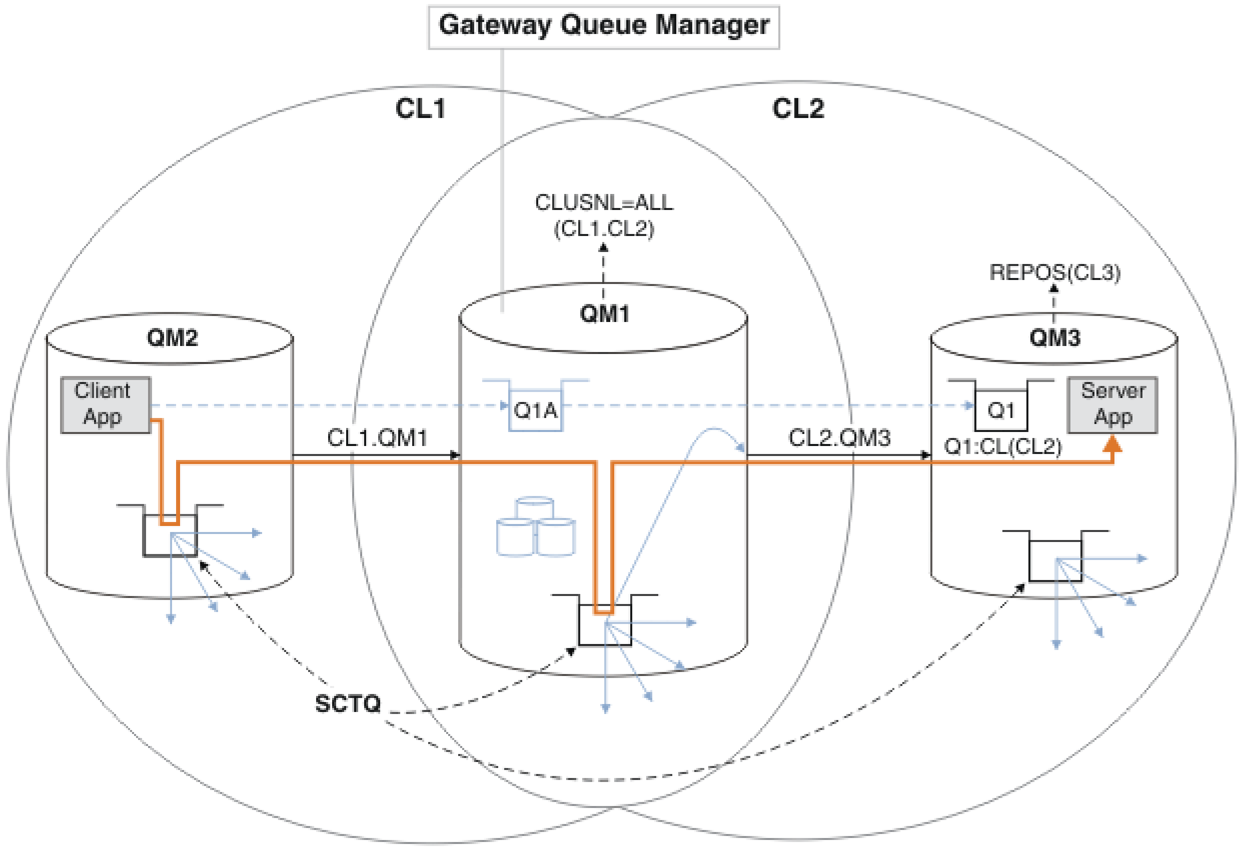


그림 *48. IBM MQ* 클러스터를 사용하여 허브 및 스포크 아키텍처에 배치되는 클라이언트 서버 애플리케이션

예를 구성하기 위한 단계의 수를 최대한 적게 하려면 구성을 실제적이 아닌 단순하게 유지합니다. 이 예는 두 개의 개별 조직에서 작성된 두 개의 클러스터 통합을 나타낼 수 있습니다. 보다 현실적인 시나리오는 클러스터링: 클러스터 전송 큐를 구성하는 방법 계획을 참조하십시오.

단계에 따라 클러스터를 구성하십시오. 클러스터는 클라이언트 애플리케이션에서 서버 애플리케이션으로의 메시지 트래픽을 격리하는 다음 예에서 사용됩니다.

지시사항은 각 클러스터가 두 개의 저장소를 가지도록 2개의 추가 큐 관리자를 추가합니다. 게이트웨이 큐 관리자는 성능상의 이유로 저장소로 사용되지 않습니다.

프로시저

1. 큐 관리자 QM1, QM2, QM3, QM4, QM5를 작성하고 시작하십시오.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QM n
strmqm QmgrName
```

참고: QM4 및 QM5는 클러스터의 백업 전체 저장소입니다.

2. 각 큐 관리자에 대해 리스너를 정의하고 시작하십시오.

```
*... On QM n
DEFINE LISTENER(TCP141 n) TRPTYPE(TCP) IPADDR(hostname) PORT(141 n) CONTROL(QMGR) REPLACE
START LISTENER(TCP141 n)
```

3. 모든 클러스터에 대해 클러스터 이름 목록을 작성하십시오.

```
*... On QM1
DEFINE NAMELIST(ALL) NAMES(CL1, CL2) REPLACE
```

4. CL1에 대한 QM2 및 QM4 전체 저장소를 작성하고 CL2에 대한 QM3 및 QM5 전체 저장소를 작성하십시오.

   a) CL1의 경우:

   ```
   *... On QM2 and QM4
   ALTER QMGR REPOS(CL1) DEFCLXQ(SCTQ)
   ```

   b) CL2의 경우:

   ```
   *... On QM3 and QM5
   ALTER QMGR REPOS(CL2) DEFCLXQ(SCTQ)
   ```

5. 각 큐 관리자 및 클러스터의 클러스터 송신자 및 클러스터 수신자 채널을 추가하십시오.

   다음 명령을 QM2, QM3, QM4 및 QM5에서 실행하십시오. 여기서, *c*, *n* 및 *m*은 각 큐 관리자의 262 페이지의 표 24에 표시된 값을 사용합니다.

표 *24.* 클러스터 *1*과 *2*를 작성하기 위한 매개변수 값

큐 관리자	군집 *c*	기타 저장소 *n*	이 저장소 *m*
QM2	1	4	2
QM4	1	2	4
QM3	2	5	3
QM5	2	3	5

```
*... On QM m
DEFINE CHANNEL(CL c.QM n) CHLTYPE(CLUSSDR) CONNAME('localhost(141 n)') CLUSTER(CL c) REPLACE
DEFINE CHANNEL(CL c.QM m) CHLTYPE(CLUSRCVR) CONNAME('localhost(141 m)') CLUSTER(CL c) REPLACE
```

6. 게이트웨이 큐 관리자 QM1을 각 클러스터에 추가하십시오.

```
*... On QM1
DEFINE CHANNEL(CL1.QM2) CHLTYPE(CLUSSDR) CONNAME('localhost(1412)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL1.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL2.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL2) REPLACE
DEFINE CHANNEL(CL2.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL2) REPLACE
```

7. 로컬 큐 Q1을 클러스터 CL2의 큐 관리자 QM3에 추가하십시오.

```
*... On QM3
DEFINE QLOCAL(Q1) CLUSTER(CL2) REPLACE
```

8. 클러스터된 큐 관리자 알리어스 Q1A를 게이트웨이 큐 관리자에 추가하십시오.

```
*... On QM1
DEFINE QALIAS(Q1A) CLUSNL(ALL) TARGET(Q1) TARGTYPE(QUEUE) DEFBIND(NOTFIXED) REPLACE
```

참고: QM1 외의 큐 관리자에서 큐 관리자 알리어스를 사용하는 애플리케이션은 알리어스 큐를 열 때 DEFBIND(NOTFIXED)를 지정해야 합니다. **DEFBIND**는 애플리케이션이 큐를 열 때 메시지 헤더의 라우팅 정보가 고정되는지 여부를 지정합니다. 기본값 OPEN으로 설정되는 경우 메시지가 Q1@QM1으로 라우트됩니다. Q1@QM1은 존재하지 않기 때문에 다른 큐 관리자의 메시지가 데드-레터 큐에 있게 됩니다. 이 큐 속성을 DEFBIND(NOTFIXED)로 설정하면 큐 설정 **DEFBIND**로 기본값 설정되는 **amqsput**과 같은 애플리케이션은 올바른 방식으로 작동합니다.

9. 게이트웨이 큐 관리자 QM1에 모든 클러스터된 큐 관리자의 클러스터 큐 관리자 알리어스 정의를 추가하십시오.

```
*... On QM1
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSNL(ALL) REPLACE
```

팁: 게이트웨이 큐 관리자의 큐 관리자 알리어스 정의는 다른 클러스터에 있는 큐 관리자를 참조하는 메시지를 전송합니다. 클러스터된 큐 관리자 알리어스를 참조하십시오.

다음에 수행할 작업

1. 큐 알리어스 정의 Q1A를 통해 QM2에서 QM3의 Q1으로 메시지를 송신하여 큐 알리어스 정의를 테스트하십시오.

   a. 메시지를 넣으려면 QM2에서 샘플 프로그램 **amqsput**를 실행하십시오.

   ```
   C:\IBM\MQ>amqsput Q1A QM2
   Sample AMQSPUT0 start
   target queue is Q1A
   Sample request message from QM2 to Q1 using Q1A

   Sample AMQSPUT0 end
   ```

   b. QM3의 Q1에서 메시지를 가져오려면 샘플 프로그램 **amqsget**를 실행하십시오.

   ```
   C:\IBM\MQ>amqsget Q1 QM3
   Sample AMQSGET0 start
   message <Sample request message from QM2 to Q1 using Q1A>
   no more messages
   Sample AMQSGET0 end
   ```

2. 임시 동적 응답 큐에서 요청 메시지를 송신하고 응답 메시지를 수신하여 큐 관리자 알리어스 정의를 테스트하십시오.

   이 다이어그램은 RQ라고 하는 응답 메시지가 다시 임시 동적 큐로 돌아가기 위해 따라가는 경로를 표시합니다. QM3에 연결되는 서버 애플리케이션은 큐 관리자 이름 QM2를 사용하여 응답 큐를 엽니다. 큐 관리자 이름 QM2는 QM1에서 클러스터된 큐 관리자 알리어스로 정의됩니다. QM3는 응답 메시지를 QM1으로 라우트합니다. QM1은 그 메시지를 QM2로 라우트합니다.

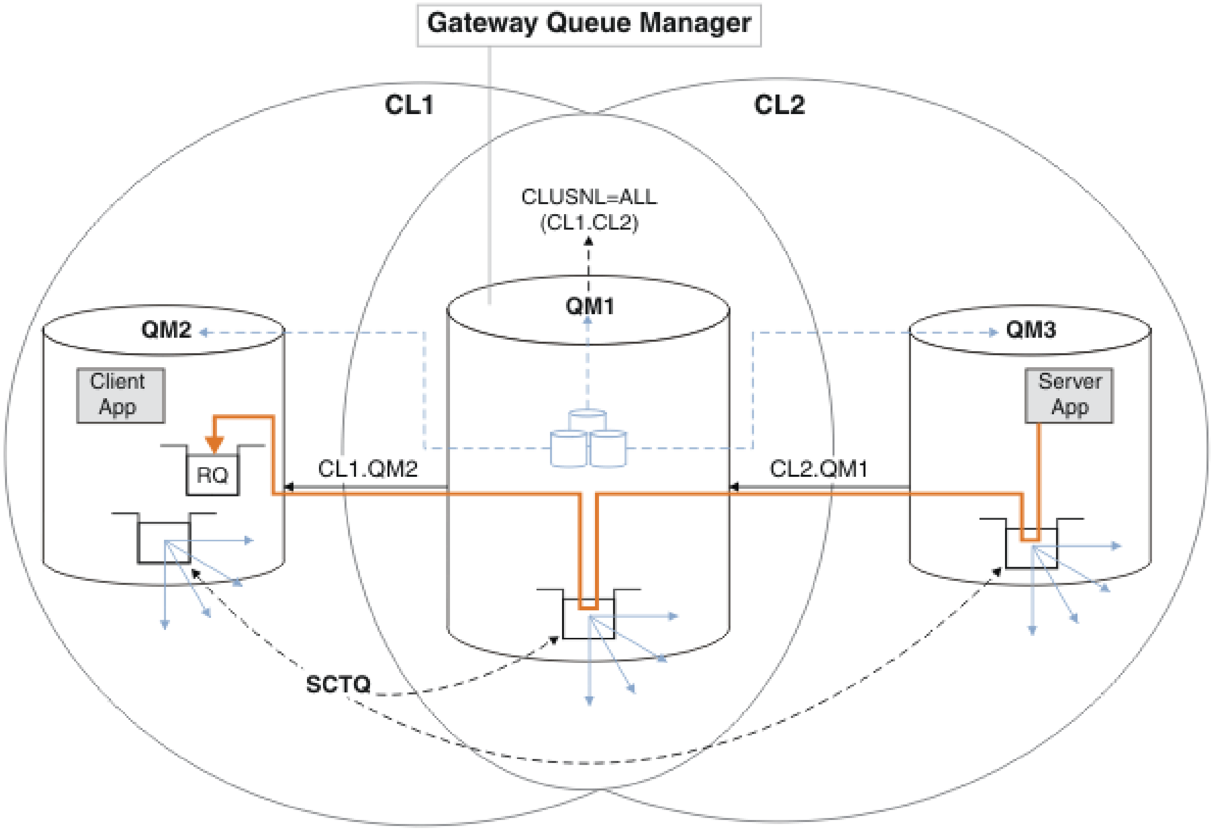


그림 *49.* 큐 관리자 알리어스를 사용하여 응답 메시지를 다른 클러스터로 리턴

라우팅이 작동하는 방식은 다음과 같습니다. 각 클러스터의 모든 큐 관리자에게는 QM1에 큐 관리자 알리어스 정의가 있습니다. 이러한 알리어스는 모든 클러스터에서 클러스터됩니다. 각 알리어스에서 큐 관리자로의 회색 파선 화살표는 각 큐 관리자 알리어스가 최소 하나 이상의 클러스터에 있는 실제 큐 관리자로 해석됨을 표시합니다. 이 경우 QM2 알리어스는 두 클러스터 CL1 및 CL2 모두에서 클러스터되어 CL1에 있는 실제 큐 관리자 QM2로 해석됩니다. 서버 애플리케이션은 큐 이름 RQ에 대한 응답 및 큐 관리자 이름 QM2에 대한 응답을 사용하여 응답 메시지를 작성합니다. 이 메시지는 QM1으로 라우트됩니다. 큐 관리자 알리어스 정의 QM2가 CL2에 있는 QM1에서 정의되고 큐 관리자 QM2는 클러스터 CL2에 없기 때문입니다. 메시지를 대상 큐 관리자로 송신할 수 없으므로 이 메시지는 알리어스 정의가 있는 큐 관리자에게 송신됩니다.

QM1은 QM2로 전송할 메시지를 QM1의 클러스터 전송 큐에 배치합니다. QM1은 메시지를 QM2로 라우트합니다. QM2에 대한 QM1의 큐 관리자 알리어스 정의는 QM2를 실제 대상 큐 관리자로 정의하기 때문입니다. 정의는 순환되지 않습니다. 알리어스 정의는 실제 정의만 참조할 수 있고 알리어스는 그 자체를 가리킬 수 없기 때문입니다. 실제 정의는 QM1에 의해 해석됩니다. QM1 및 QM2 둘 다 같은 클러스터 CL1에 있기 때문입니다. QM1은 CL1의 저장소에서 QM2에 대한 연결 정보를 찾아내고 메시지를 QM2에 라우트합니다. 메시지가 QM1에 의해 재라우트되도록 하려면 서버 애플리케이션은 DEFBIND 옵션 MQBND_BIND_NOT_FIXED로 설정하여 응답 큐를 열어야 합니다. 서버 애플리케이션이 MQBND_BIND_ON_OPEN 옵션을 사용하여 응답 큐를 연 경우 메시지는 재라우트되지 않고 데드-레터 큐에 있게 됩니다.

a. QM3에서 트리거를 사용하여 클러스터된 요청 큐를 작성하십시오.

```
*... On QM3
DEFINE QLOCAL(QR) CLUSTER(CL2) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE)
PROCESS(ECHO) REPLACE
```

b. 게이트웨이 큐 관리자 QM1에서 QR의 클러스터된 큐 알리어스 정의를 작성하십시오.

```
*... On QM1
DEFINE QALIAS(QRA) CLUSNL(ALL) TARGET(QR) TARGTYPE(QUEUE) DEFBIND(NOTFIXED) REPLACE
```

c. QM3에서 샘플 에코 프로그램 **amqsech**를 시작하기 위한 프로세스 정의를 작성하십시오.

```
*... On QM3
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

d. 임시 동적 응답 큐를 작성하려면 QM2에서 샘플 프로그램 **amqsreq**에 대한 모델 큐를 작성하십시오.

```
*... On QM2
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

e. 큐 알리어스 정의 QRA를 통해 QM2에서 QM3의 QR로 요청을 송신하여 큐 관리자 알리어스 정의를 테스트하십시오.

i) QM3에서 트리거 모니터 프로그램을 실행하십시오.

```
runmqtrm -m QM3
```

출력은 다음과 같습니다.

```
C:\IBM\MQ>runmqtrm -m QM3
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
01/02/2012  16:17:15: IBM MQ trigger monitor started.

__________________________________________________
01/02/2012  16:17:15: Waiting for a trigger message
```

ii) 요청을 넣고 응답을 기다리려면 QM2에서 샘플 프로그램 **amqsreq**를 실행하십시오.

```
C:\IBM\MQ>amqsreq QRA QM2
Sample AMQSREQ0 start
server queue is QRA
replies to 4F2961C802290020
A request message from QM2 to QR on QM3

response <A request message from QM2 to QR on QM3>
no more replies
Sample AMQSREQ0 end
```

관련 태스크

239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

관련 정보

액세스 제어 및 다중 클러스터 전송 큐
클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리
클러스터링: 클러스터 전송 큐를 구성하는 방법을 계획

게이트웨이 큐 관리자에서 송신된 메시지를 격리하기 위한 리모트 큐 정의 추가

게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 클러스터된 큐 리모트 정의 및 개별 송신자 채널과 전송 큐를 사용합니다.

시작하기 전에

해당 태스크의 단계에 따라 261 페이지의 『게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성』에 있는 IBM MQ 클러스터를 사용하여 허브 및 스포크 아키텍처에 배치된 클라이언트 서버 애플리케이션에 표시된 중첩 클러스터를 구성하십시오.

이 태스크 정보

솔루션은 분산 큐잉을 사용하여 게이트웨이 큐 관리자의 다른 메시지 트래픽으로부터 `Server App` 애플리케이션에 대한 메시지를 분리합니다. QM1에서 클러스터된 리모트 큐 정의를 정의하여 메시지를 다른 전송 큐와 다른 채널로 우회시켜야 합니다. 리모트 큐 정의는 QM3의 Q1에 대한 메시지만 저장하는 특정 전송 큐에 대한 참조를 포함해야 합니다. 266 페이지의 그림 50에서 클러스터 큐 알리어스 Q1A는 리모트 큐 정의 Q1R에 의해 보완되고 전송 큐 및 송신자 채널이 추가됩니다.

이 솔루션에서 응답 메시지는 공통 SYSTEM.CLUSTER.TRANSMIT.QUEUE를 사용하여 리턴됩니다.

이 솔루션의 장점은 같은 클러스터에 있는 같은 큐 관리자에서 여러 목적지 큐에 대한 트래픽을 쉽게 분리할 수 있다는 것입니다. 이 솔루션의 단점은 서로 다른 큐 관리자에서 Q1의 여러 사본 사이에 클러스터 워크로드 밸런싱을 사용할 수 없다는 것입니다. 이 단점을 극복하려면 268 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가』의 내용을 참조하십시오. 또한 한 전송 큐에서 다른 전송 큐로의 전환도 관리해야 합니다.

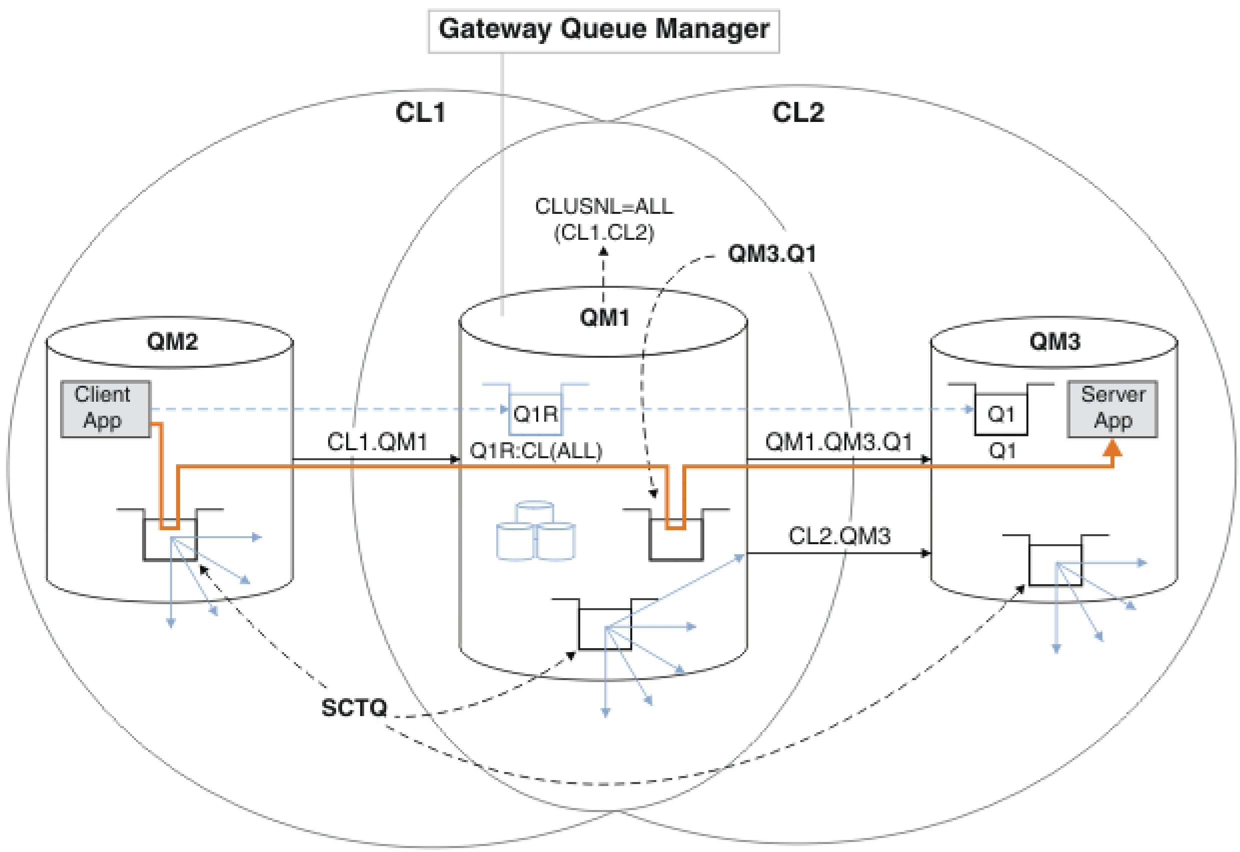


그림 *50.* 리모트 큐 정의를 사용하여 허브 및 스포크 클러스터 아키텍처에 배치되는 클러스터 서버 애플리케이션

프로시저

1. Q1에 대한 메시지 트래픽을 게이트웨이 큐 관리자로부터 분리하기 위한 채널을 작성하십시오.
   a) 게이트웨이 큐 관리자 QM1에서 대상 큐 관리자 QM3에 대한 송신자 채널을 작성하십시오.

   ```
   DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(SDR) CONNAME(QM3HostName(1413)) XMITQ(QM3.Q1) REPLACE
   ```

   b) 대상 큐 관리자 QM3에서 수신자 채널을 작성하십시오.

   ```
   DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(RCVR) REPLACE
   ```

2. 게이트웨이 큐 관리자에서 Q1에 대한 메시지 트래픽용 전송 큐를 작성하십시오.

```
DEFINE QLOCAL(QM3.Q1) USAGE(XMITQ) REPLACE
START CHANNEL(QM1.QM3.Q1)
```

전송 큐와 연관된 채널을 시작할 때 이 전송 큐를 채널과 연관시키십시오. 전송 큐가 채널과 연관되는 즉시 채널이 자동으로 시작됩니다.

3. 게이트웨이 큐 관리자에서 클러스터된 리모트 큐 정의를 사용하여 Q1에 대한 클러스터된 큐 알리어스 정의를 보완하십시오.

```
DEFINE QREMOTE CLUSNL(ALL) RNAME(Q1) RQMNAME(QM3) XMITQ(QM3.Q1) REPLACE
```

다음에 수행할 작업

게이트웨이 큐 관리자 QM1에서 클러스터된 큐 리모트 정의 Q1R을 통해 QM2에서 QM3의 Q1으로 메시지를 송신하여 구성을 테스트하십시오.

1. 메시지를 넣으려면 QM2에서 샘플 프로그램 **amqsput**를 실행하십시오.

```
C:\IBM\MQ>amqsput Q1R QM2
Sample AMQSPUT0 start
target queue is Q1R
Sample request message from QM2 to Q1 using Q1R

Sample AMQSPUT0 end
```

2. QM3의 Q1에서 메시지를 가져오려면 샘플 프로그램 **amqsget**를 실행하십시오.

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message &lt;Sample request message from QM2 to Q1 using Q1R>
no more messages
Sample AMQSGET0 end
```

관련 태스크

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 추가 클러스터 전송 큐를 사용하여 클러스터의 단일 큐 관리자로 메시지 트래픽을 분리합니다.

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 솔루션은 추가 클러스터를 사용하여 메시지를 특정 클러스터 큐로 분리합니다.

메시지 트래픽을 격리하기 위해 클러스터 전송 큐를 분리하도록 기본값 변경
클러스터된 큐 또는 전송 큐의 토픽에 대해 큐 관리자가 메시지를 저장하는 기본 방식을 변경할 수 있습니다. 기본값을 변경하여 게이트웨이 큐 관리자에서 클러스터 메시지를 격리하는 방식을 제공합니다.

239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

관련 정보

클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리
클러스터링: 클러스터 전송 큐를 구성하는 방법을 계획
액세스 제어 및 다중 클러스터 전송 큐

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가

게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 추가 클러스터 전송 큐를 사용하여 클러스터의 단일 큐 관리자로 메시지 트래픽을 분리합니다.

시작하기 전에

1. 게이트웨이 큐 관리자는 IBM WebSphere MQ 7.5 이상에 있어야 합니다.
2. 해당 태스크의 단계에 따라 261 페이지의 『게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성』에 있는 IBM MQ 클러스터를 사용하여 허브 및 스포크 아키텍처에 배치된 클라이언트 서버 애플리케이션에 표시된 중첩 클러스터를 구성하십시오.

이 태스크 정보

게이트웨이 큐 관리자 QM1에서 전송 큐를 추가하고 그 큐 속성 CLCHNAME을 설정하십시오. CLCHNAME을 QM3의 클러스터 수신자 채널의 이름으로 설정하십시오. 269 페이지의 그림 51의 내용을 참조하십시오.

이 솔루션에는 265 페이지의 『게이트웨이 큐 관리자에서 송신된 메시지를 격리하기 위한 리모트 큐 정의 추가』에서 설명한 솔루션에 비해 여러 장점이 있습니다.

- 적은 수의 추가 정의가 필요합니다.
- 같은 클러스터 CL2의 여러 다른 큐 관리자에서 대상 큐 Q1의 여러 사본 간 워크로드 밸런싱을 지원합니다.
- 채널이 재시작되면 게이트웨이 큐 관리자가 메시지를 손실하지 않고 새 구성으로 자동 전환됩니다.
- 게이트웨이 큐 관리자가 메시지를 수신한 순서와 같은 순서로 계속해서 메시지를 전달합니다. 이러한 메시지 전달은 여전히 SYSTEM.CLUSTER.TRANSMIT.QUEUE에 있는 QM3의 큐 Q1에 대한 메시지에 대해 전환이 발생하는 경우에도 실행됩니다.

269 페이지의 그림 51에서 클러스터 메시지 트래픽을 격리하는 구성은 265 페이지의 『게이트웨이 큐 관리자에서 송신된 메시지를 격리하기 위한 리모트 큐 정의 추가』에서 리모트 큐를 사용하는 구성만큼의 트래픽 격리 효과를 얻지 못합니다. CL2에서 큐 관리자 QM3가 여러 다른 클러스터 큐와 서버 애플리케이션을 호스팅하는 경우 호스팅되는 모든 큐는 QM1을 QM3에 연결하는 클러스터 채널 CL2.QM3를 공유합니다. 추가 플로우는 SYSTEM.CLUSTER.TRANSMIT.QUEUE에서 클러스터 송신자 채널 CL2.QM3로의 잠재적 클러스터 메시지 트래픽을 나타내는 회색 화살표로 269 페이지의 그림 51에 표시됩니다.

해결책은 큐 관리자를 특정 클러스터의 한 클러스터 큐를 호스팅하는 것으로만 제한하는 것입니다. 큐 관리자가 이미 다수의 클러스터 큐를 호스트 중인 경우 이 제한을 충족시키려면 다른 큐 관리자를 작성하거나 다른 클러스터를 작성해야 합니다. 271 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가』의 내용을 참조하십시오.

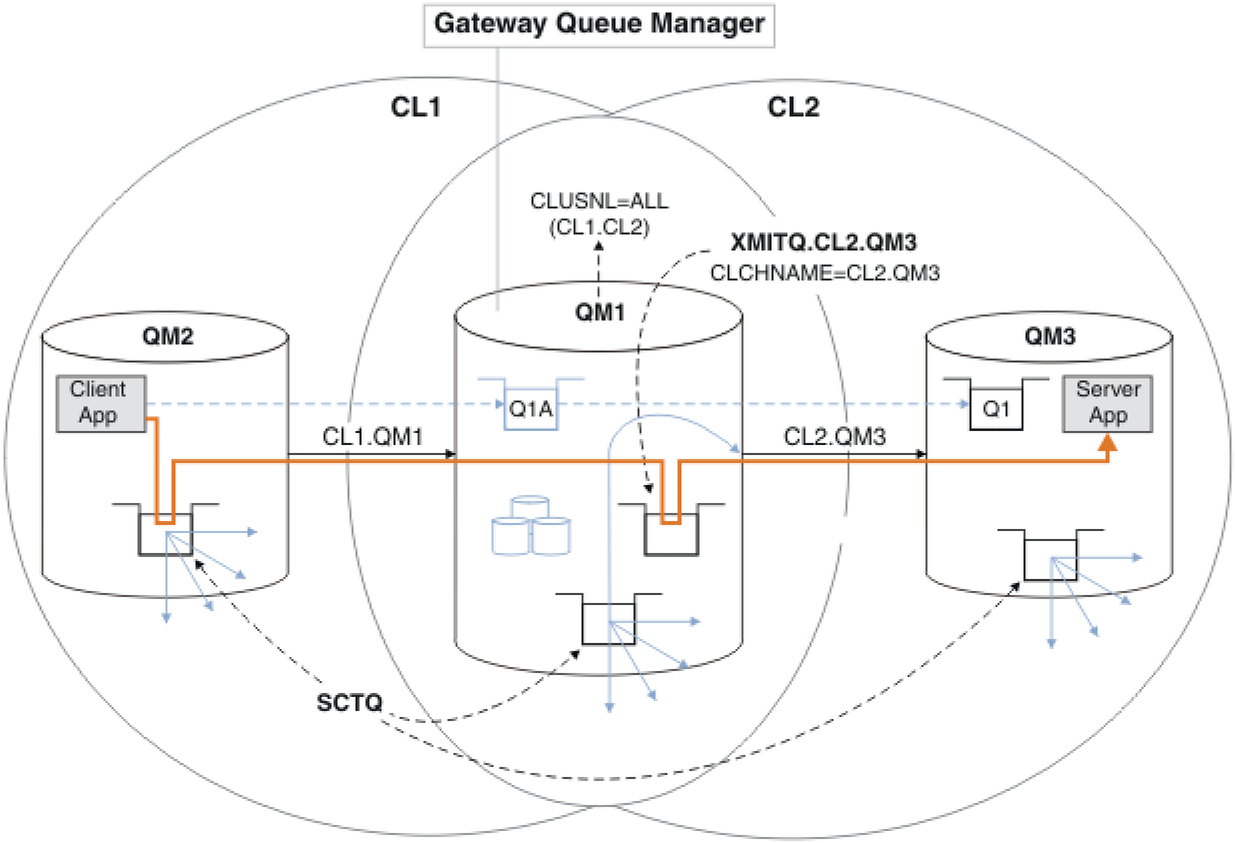


그림 *51.* 추가 클러스터 전송 큐를 사용하여 허브 및 스포크 아키텍처에 배치되는 클라이언트 서버 애플리케이션

프로시저

1. 게이트웨이 큐 관리자 QM1에서 클러스터 송신자 채널 CL2.QM3에 대한 추가 클러스터 전송 큐를 작성하십시오.

```
*... on QM1
DEFINE QLOCAL(XMITQ.CL2.QM3) USAGE(XMITQ) CLCHNAME(CL2.QM3)
```

2. 전송 큐 XMITQ.CL2.QM3 사용으로 전환하십시오.

   a) 클러스터 송신자 채널 CL2.QM3를 중지하십시오.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

   응답은 명령이 승인됨입니다.

```
AMQ8019: Stop IBM MQ channel accepted.
```

   b) 채널 CL2.QM3가 중지되는지 확인하십시오.

   채널이 중지되지 않으면 **STOP CHANNEL** 명령을 FORCE 옵션과 함께 다시 실행할 수 있습니다. FORCE 옵션 설정의 예는 채널이 중지되지 않고 채널을 동기화하기 위해 다른 큐 관리자를 재시작할 수 없는 경우입니다.

```
*... On QM1
start
```

   응답은 채널 상태의 요약입니다.

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))    CURRENT
RQMNAME(QM3)               STATUS(STOPPED)
SUBSTATE (MQGET) XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

c) 채널 CL2.QM3를 시작하십시오.

```
*... On QM1
START CHANNEL(CL2.QM3)
```

응답은 명령이 승인됨입니다.

```
AMQ8018: Start IBM MQ channel accepted.
```

d) 채널이 시작되었는지 확인하십시오.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

응답은 채널 상태의 요약입니다.

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))    CURRENT
RQMNAME(QM3)                            STATUS(RUNNING)
SUBSTATE(MQGET)                    XMITQ(XMITQ.CL2.QM3)
```

e) 전송 큐가 전환되었는지 확인하십시오.

메시지 " AMQ7341 The transmission queue for channel CL2.QM3 is XMITQ.CL2.QM3 "에 대해 게이트웨이 큐 관리자 오류 로그를 모니터하십시오.

다음에 수행할 작업

큐 알리어스 정의 Q1A를 사용하여 QM2의 메시지를 QM3의 Q1로 전송하여 개별 전송 큐를 테스트하십시오.

1. 메시지를 넣으려면 QM2에서 샘플 프로그램 **amqsput**를 실행하십시오.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end
```

2. QM3의 Q1에서 메시지를 가져오려면 샘플 프로그램 **amqsget**를 실행하십시오.

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

관련 개념

221 페이지의 『클러스터 전송 큐 및 클러스터 송신자 채널에 대한 작업』
클러스터된 큐 관리자 간 메시지는 클러스터 전송 큐에 저장되고 클러스터 송신자 채널에서 전달됩니다. 임의의 시점에서 클러스터 송신자 채널은 한 개의 전송 큐와 연관됩니다. 채널의 구성을 변경하는 경우 다음에 시작할 때 다른 전송 큐로 전환할 수도 있습니다. 이 전환의 처리는 자동화되고 트랜잭션 가능합니다.

관련 태스크

게이트웨이 큐 관리자에서 송신된 메시지를 격리하기 위한 리모트 큐 정의 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 클러스터된 큐 리모트 정의 및 개별 송신자 채널과 전송 큐를 사용합니다.

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 솔루션은 추가 클러스터를 사용하여 메시지를 특정 클러스터 큐로 분리합니다.

메시지 트래픽을 격리하기 위해 클러스터 전송 큐를 분리하도록 기본값 변경
클러스터된 큐 또는 전송 큐의 토픽에 대해 큐 관리자가 메시지를 저장하는 기본 방식을 변경할 수 있습니다. 기본값을 변경하여 게이트웨이 큐 관리자에서 클러스터 메시지를 격리하는 방식을 제공합니다.

239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

관련 정보

액세스 제어 및 다중 클러스터 전송 큐

클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리

클러스터링: 클러스터 전송 큐를 구성하는 방법을 계획

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가

게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 솔루션은 추가 클러스터를 사용하여 메시지를 특정 클러스터 큐로 분리합니다.

시작하기 전에

이 태스크의 단계는 269 페이지의 그림 51에 설명된 구성을 수정하기 위해 작성됩니다.

1. 게이트웨이 큐 관리자는 IBM WebSphere MQ 7.5 이상에 있어야 합니다.
2. 해당 태스크의 단계에 따라 261 페이지의 『게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성』에 있는 IBM MQ 클러스터를 사용하여 허브 및 스포크 아키텍처에 배치된 클라이언트 서버 애플리케이션에 표시된 중첩 클러스터를 구성하십시오.
3. 268 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가』에 있는 269 페이지의 그림 51의 단계를 수행하여 추가 클러스터 없이 솔루션을 작성하십시오. 이를 이 태스크의 단계를 위한 기반으로 사용하십시오.

이 태스크 정보

268 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가』에서 메시지 트래픽을 단일 애플리케이션으로 격리하는 솔루션은 대상 클러스터 큐가 큐 관리자에 있는 유일한 클러스터 큐인 경우에 작동합니다. 그렇지 않은 경우에는 다음 두 가지 선택사항이 있습니다. 큐를 다른 큐 관리자로 이동하거나 큐 관리자의 다른 클러스터 큐에서 큐를 격리하는 클러스터를 작성하십시오.

이 태스크는 대상 큐를 격리하기 위한 클러스터를 추가하는 단계에 대해 설명합니다. 클러스터는 해당 용도로만 추가됩니다. 실제로 클러스터 및 클러스터 이름 지정 체계를 설계하는 과정에 있을 때 특정 애플리케이션을 체계적으로 격리하는 태스크를 시작하십시오. 큐가 격리를 요구할 때마다 클러스터를 추가하면 관리할 클러스터만 많아질 수 있습니다. 이 태스크에서는 QM3의 Q1을 격리하기 위해 클러스터 CL3를 추가하여 268 페이지의 『게

이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가』에서 구성을 변경합니다. 애플리케이션은 변경 전 과정 동안 계속해서 실행됩니다.

새로운 정의 및 변경된 정의는 272 페이지의 그림 52에서 강조표시됩니다. 변경사항의 요약은 다음과 같습니다. 클러스터를 작성합니다. 새 전체 클러스터 저장소도 작성해야 함을 의미합니다. 예에서 QM3가 CL3의 전체 저장소 중 하나가 됩니다. QM1에 대해 클러스터 송신자 및 클러스터 수신자 채널을 작성하여 게이트웨이 큐 관리자를 새 클러스터에 추가하십시오. Q1의 정의를 변경하여 이 정의를 CL3로 전환하십시오. 게이트웨이 큐 관리자에서 클러스터 이름 목록을 수정하고 새 클러스터 채널을 사용할 클러스터 전송 큐를 추가하십시오. 마지막으로 큐 알리어스 Q1A를 새 클러스터 이름 목록으로 전환하십시오.

IBM MQ는 268 페이지의 『게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가』에서 추가한 전송 큐 XMITQ.CL2.QM3에서 새 전송 큐 XMITQ.CL3.QM3로 메시지를 자동 전송할 수 없습니다. 두 전송 큐 모두 같은 클러스터 송신자 채널에 의해 지원되는 경우에만 메시지를 자동 전송할 수 있습니다. 대신 이 태스크는 전환을 수동으로 수행하는 한 가지 방식을 설명하며 이 방식이 사용자에게 적합할 수 있습니다. 전송이 완료된 경우 QM3에 있는 다른 CL2 클러스터 큐에 대해 기본 클러스터 전송 큐를 사용하도록 되돌아가는 옵션이 있습니다. 또는 계속해서 XMITQ.CL2.QM3를 사용할 수 있습니다. 기본 클러스터 전송 큐로 되돌아가기로 결정한 경우 게이트웨이 큐 관리자는 전환을 자동으로 관리합니다.

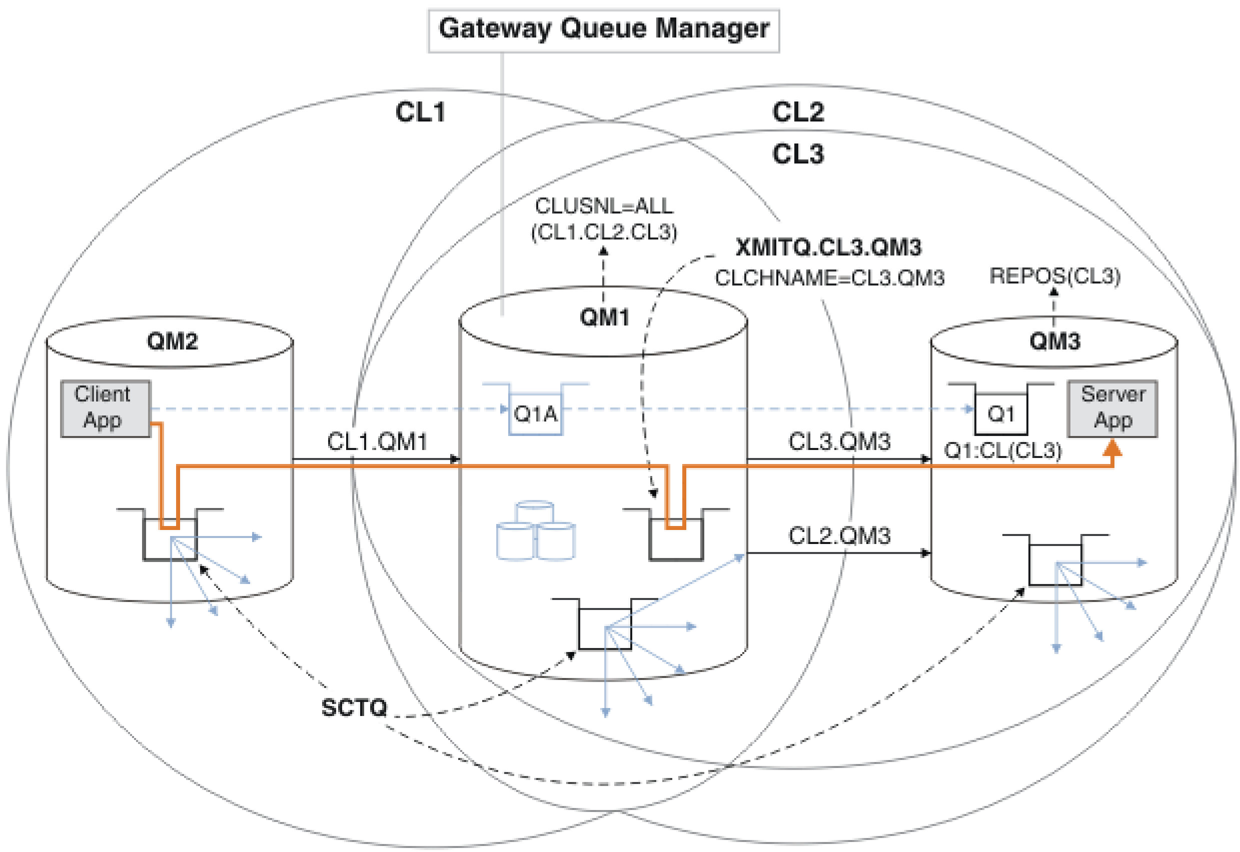


그림 *52.* 같은 큐 관리자에 있는 다수의 클러스터 중 하나로 이동하는 메시지 트래픽을 게이트웨이 큐 관리자에서 분리하기 위해 추가 클러스터 사용

프로시저

1. 큐 관리자 QM3 및 QM5를 변경하여 CL2 및 CL3 둘 다를 위한 저장소로 만드십시오.

   큐 관리자를 다중 클러스터의 멤버가 되게 하려면 큐 관리자가 멤버인 클러스터를 식별하는 클러스터 이름 목록을 사용해야 합니다.

```
*... On QM3 and QM5
DEFINE NAMELIST(CL23) NAMES(CL2, CL3) REPLACE
ALTER QMGR REPOS(' ') REPOSNL(CL23)
```

2. CL3에 대한 큐 관리자 QM3 및 QM5 간 채널을 정의하십시오.

```
*... On QM3
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSRCVR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE

*... On QM5
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
```

3. CL3에 게이트웨이 큐 관리자를 추가하십시오.

   QM1을 부분 저장소로 CL3에 추가하여 게이트웨이 큐 관리자를 추가하십시오. 클러스터 송신자 및 클러스터 수신자 채널을 QM1에 추가하여 부분 저장소를 작성하십시오.

   또한 CL3를 게이트웨이 큐 관리자에 연결된 모든 클러스터의 이름 목록에 추가하십시오.

```
*... On QM1
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL3) REPLACE
ALTER NAMELIST(ALL) NAMES(CL1, CL2, CL3)
```

4. QM3에서 CL3로 이동하는 메시지에 대해 게이트웨이 큐 관리자 QM1에 클러스터 전송 큐를 추가하십시오.

   처음에는 전송 큐를 전환할 준비가 될 때까지 전송 큐에서 메시지를 전송하는 클러스터 송신자 채널을 중지하십시오.

```
*... On QM1
DEFINE QLOCAL(XMITQ.CL3.QM3) USAGE(XMITQ) CLCHNAME(CL3.QM3) GET(DISABLED) REPLACE
```

5. 기존 클러스터 전송 큐 XMITQ.CL2.QM3에서 메시지를 배출하십시오.

   이 서브프로시저는 메시지가 게이트웨이 큐 관리자에 도착한 순서와 일치하도록 Q1에서 메시지의 순서를 유지하는 것을 목적으로 합니다. 클러스터에서 메시지 순서 지정은 완전히는 아니더라도 어느 정도는 보장됩니다. 보장된 메시지 순서 지정이 필요한 경우 애플리케이션이 메시지의 순서를 정의해야 합니다. 큐에서 메시지가 검색되는 순서를 참조하십시오.

   a) QM3의 대상 큐 Q1을 CL2에서 CL3로 변경하십시오.

   ```
   *... On QM3
   ALTER QLOCAL(Q1) CLUSTER(CL3)
   ```

   b) 메시지가 전달되기 시작할 때까지 XMITQ.CL3.QM3를 모니터하십시오.

      Q1에서 CL3로의 전환이 게이트웨이 큐 관리자에게 전파되면 메시지가 XMITQ.CL3.QM3에 전달되기 시작합니다.

   ```
   *... On QM1
   DISPLAY QUEUE(XMITQ.CL3.QM3) CURDEPTH
   ```

   c) QM3의 Q1에 전달되기 위해 대기 중인 메시지가 없을 때까지 XMITQ.CL2.QM3를 모니터하십시오.

      참고: XMITQ.CL2.QM3는 CL2의 멤버인 QM3에서 다른 큐에 대한 메시지를 저장할 수 있으며 이 경우에 용량은 0이 되지 않을 수 있습니다.

   ```
   *... On QM1
   DISPLAY QUEUE(XMITQ.CL2.QM3) CURDEPTH
   ```

   d) 새 클러스터 전송 큐 XMITQ.CL3.QM3에서 가져오기를 사용으로 설정하십시오.

   ```
   *... On QM1
   ALTER QLOCAL(XMITQ.CL3.QM3) GET(ENABLED)
   ```

6. 더 이상 필요하지 않은 경우 오래된 클러스터 송신 큐 XMITQ.CL2.QM3를 제거하십시오.

   QM3의 CL2에서 클러스터 큐에 대한 메시지는 게이트웨이 큐 관리자 QM1에서 기본 클러스터 전송 큐를 사용하는 것으로 되돌아갑니다. 기본 클러스터 전송 큐는 SYSTEM.CLUSTER.TRANSMIT.QUEUE 또는

SYSTEM.CLUSTER.TRANSMIT.CL2.QM3입니다. 이 중 사용되는 것은 QM1에서 큐 관리자 속성 **DEFCLXQ**의 값이 SCTQ인지 또는 CHANNEL인지에 따라 달라집니다. 큐 관리자는 클러스터 송신자 채널 CL2.QM3가 다음에 시작될 때 XMITQ.CL2.QM3에서 자동으로 메시지를 전송합니다.

a) 전송 큐 XMITQ.CL2.QM3를 클러스터 전송 큐인 상태에서 일반 전송 큐인 상태로 변경하십시오.

이렇게 하면 클러스터 송신자 채널과 전송 큐의 연관이 해제됩니다. 이에 따라 IBM MQ는 클러스터 송신자 채널이 다음에 시작될 때 XMITQ.CL2.QM3에서 기본 클러스터 전송 큐로 메시지를 자동 전송합니다. 그 때까지 QM3의 CL2에 대한 메시지는 계속해서 XMITQ.CL2.QM3에 배치됩니다.

```
*... On QM1
ALTER QLOCAL(XMITQ.CL2.QM3) CLCHNAME(' ')
```

b) 클러스터 송신자 채널 CL2.QM3를 중지하십시오.

클러스터 송신자 채널을 중지하고 재시작하면 XMITQ.CL2.QM3에서 기본 클러스터 전송 큐로의 메시지 전송이 시작됩니다. 일반적으로 전송을 시작하려면 수동으로 채널을 중지하고 시작합니다. 연결 끊기 간격이 만료되어 종료된 후에 채널이 재시작되는 경우 전송이 자동으로 시작됩니다.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

응답은 명령이 승인됨입니다.

```
AMQ8019: Stop IBM MQ channel accepted.
```

c) 채널 CL2.QM3가 중지되는지 확인하십시오.

채널이 중지되지 않으면 **STOP CHANNEL** 명령을 FORCE 옵션과 함께 다시 실행할 수 있습니다. FORCE 옵션 설정의 예는 채널이 중지되지 않고 채널을 동기화하기 위해 다른 큐 관리자를 재시작할 수 없는 경우입니다.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

응답은 채널 상태의 요약입니다.

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                          CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))                  CURRENT
RQMNAME(QM3)                              STATUS(STOPPED)
SUBSTATE(MQGET)                           XMITQ(XMITQ.CL2.QM3)
```

d) 채널 CL2.QM3를 시작하십시오.

```
*... On QM1
START CHANNEL(CL2.QM3)
```

응답은 명령이 승인됨입니다.

```
AMQ8018: Start IBM MQ channel accepted.
```

e) 채널이 시작되었는지 확인하십시오.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

응답은 채널 상태의 요약입니다.

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)      CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))  CURRENT
RQMNAME(QM3)        STATUS(RUNNING)
SUBSTATE(MQGET)      XMITQ(SYSTEM.CLUSTER.TRANSMIT. QUEUE|CL2.QM3)
```

f) 메시지 " AMQ7341 `The transmission queue for channel CL2.QM3 is SYSTEM.CLUSTER.TRANSMIT.` *QUEUE|CL2.QM3* ".

g) 클러스터 전송 큐 `XMITQ.CL2.QM3`를 삭제하십시오.

```
*... On QM1
DELETE QLOCAL(XMITQ.CL2.QM3)
```

다음에 수행할 작업

큐 알리어스 정의 Q1A를 사용하여 QM2의 메시지를 QM3의 Q1로 전송함으로써 개별적으로 클러스터 큐를 테스트하십시오.

1. 메시지를 넣으려면 QM2에서 샘플 프로그램 **amqsput**를 실행하십시오.

```
C:\IBM\MQ>amqsput Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end
```

2. QM3의 Q1에서 메시지를 가져오려면 샘플 프로그램 **amqsget**를 실행하십시오.

```
C:\IBM\MQ>amqsget Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

관련 개념

221 페이지의 『클러스터 전송 큐 및 클러스터 송신자 채널에 대한 작업』
클러스터된 큐 관리자 간 메시지는 클러스터 전송 큐에 저장되고 클러스터 송신자 채널에서 전달됩니다. 임의의 시점에서 클러스터 송신자 채널은 한 개의 전송 큐와 연관됩니다. 채널의 구성을 변경하는 경우 다음에 시작할 때 다른 전송 큐로 전환할 수도 있습니다. 이 전환의 처리는 자동화되고 트랜잭션 가능합니다.

관련 태스크

게이트웨이 큐 관리자에서 송신된 메시지를 격리하기 위한 리모트 큐 정의 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 클러스터된 큐 리모트 정의 및 개별 송신자 채널과 전송 큐를 사용합니다.

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 추가 클러스터 전송 큐를 사용하여 클러스터의 단일 큐 관리자로 메시지 트래픽을 분리합니다.

메시지 트래픽을 격리하기 위해 클러스터 전송 큐를 분리하도록 기본값 변경
클러스터된 큐 또는 전송 큐의 토픽에 대해 큐 관리자가 메시지를 저장하는 기본 방식을 변경할 수 있습니다. 기본값을 변경하여 게이트웨이 큐 관리자에서 클러스터 메시지를 격리하는 방식을 제공합니다.

239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

관련 정보
액세스 제어 및 다중 클러스터 전송 큐
클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리
클러스터링: 클러스터 전송 큐를 구성하는 방법을 계획

메시지 트래픽을 격리하기 위해 클러스터 전송 큐를 분리하도록 기본값 변경

클러스터된 큐 또는 전송 큐의 토픽에 대해 큐 관리자가 메시지를 저장하는 기본 방식을 변경할 수 있습니다. 기본값을 변경하여 게이트웨이 큐 관리자에서 클러스터 메시지를 격리하는 방식을 제공합니다.

시작하기 전에

1. 게이트웨이 큐 관리자는 IBM WebSphere MQ 7.5 이상에 있어야 합니다.
2. 해당 태스크의 단계에 따라 261 페이지의 『게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성』에 있는 IBM MQ 클러스터를 사용하여 허브 및 스포크 아키텍처에 배치된 클라이언트 서버 애플리케이션에 표시된 중첩 클러스터를 구성하십시오.

이 태스크 정보

다중 클러스터 큐가 포함된 아키텍처를 구현하려면 게이트웨이 큐 관리자는 IBM WebSphere MQ 7.5 이상에 있어야 합니다. 다중 클러스터 전송 큐를 사용하기 위해 수행해야 하는 작업은 게이트웨이 큐 관리자에서 기본 클러스터 전송 큐를 변경하는 것입니다. `QM1`에서 큐 관리자 속성 **DEFCLXQ**의 값을 `SCTQ`에서 `CHANNEL`로 변경하십시오. 277 페이지의 그림 53의 내용을 참조하십시오. 이 다이어그램은 한 개의 메시지 플로우를 표시합니다. 다른 큐 관리자 또는 다른 클러스터로의 플로우에 대해서 큐 관리자는 추가 영구 동적 클러스터 전송 큐를 작성합니다. 각 클러스터 송신자 채널은 다른 클러스터 전송 큐에서 메시지를 전송합니다.

이러한 변경은 게이트웨이 큐 관리자를 클러스터에 처음으로 연결하지 않는 한 즉시 적용되지 않습니다. 이 태스크는 기존 구성에 대한 일반적인 변경 관리 단계를 포함합니다. 처음 클러스터에 조인할 때 개별 클러스터 전송 큐를 사용하도록 큐 관리자를 설정하려면 239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』의 내용을 참조하십시오.

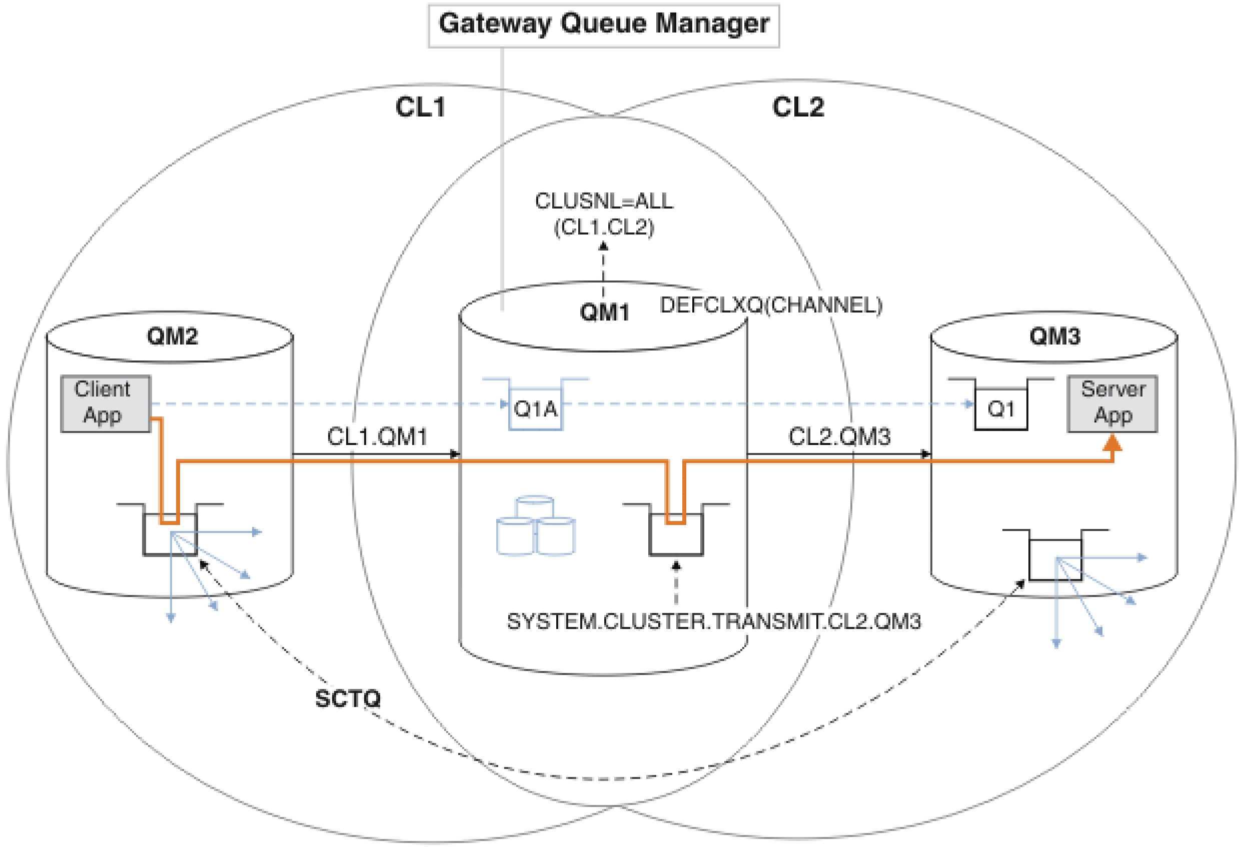


그림 *53.* 게이트웨이 큐 관리자에서 개별 클러스터 전송 큐를 사용하여 허브 및 스포크 아키텍처에 배치되는 클라이언트 서버 애플리케이션

프로시저

1. 개별 클러스터 전송 큐를 사용하도록 게이트웨이 큐 관리자를 변경하십시오.

```
*... On QM1
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. 개별 클러스터 전송 큐로 전환하십시오.

   실행 중이 아닌 클러스터 송신자 채널은 다음에 시작할 때 개별 클러스터 전송 큐를 사용하도록 전환됩니다.

   실행 중인 채널을 전환하려면 큐 관리자를 재시작하거나 다음 단계를 따르십시오.

   a) SYSTEM.CLUSTER.TRANSMIT.QUEUE로 실행되고 있는 클러스터 송신자 채널을 나열하십시오.

   ```
   *... On QM1
   DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
   ```

   응답은 채널 상태 보고서 목록입니다.

   ```
   AMQ8417: Display Channel Status details.
   CHANNEL(CL1.QM2)             CHLTYPE(CLUSSDR)
   CONNAME(127.0.0.1(1412))    CURRENT
   RQMNAME(QM2)                 STATUS(RUNNING)
   SUBSTATE(MQGET)              XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
   AMQ8417: Display Channel Status details.
   CHANNEL(CL2.QM3)             CHLTYPE(CLUSSDR)
   CONNAME(127.0.0.1(1413))     CURRENT
   ```

```
RQMNAME(QM3)                            STATUS(RUNNING)
SUBSTATE(MQGET)             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1415))                CURRENT
RQMNAME(QM5)                            STATUS(RUNNING)
SUBSTATE(MQGET)             XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1414))    CURRENT
RQMNAME(QM4)                STATUS(RUNNING)
SUBSTATE(MQGET)              XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

b) 실행 중인 채널을 중지하십시오.

목록에 있는 각 채널에 대해 다음 명령을 실행하십시오.

```
*... On QM1
STOP CHANNEL(ChannelName)
```

여기서, *ChannelName*은 각각 `CL1.QM2, CL1.QM4, CL1.QM3, CL1.QM5`입니다.

응답은 명령이 승인됨입니다.

```
AMQ8019: Stop IBM MQ channel accepted.
```

c) 어떤 채널이 중지되는지 모니터하십시오.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

응답은 여전히 실행 중인 채널과 중지되는 채널의 목록입니다.

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1412))   CURRENT
RQMNAME(QM2)           STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))    CURRENT
RQMNAME(QM3)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1415))                CURRENT
RQMNAME(QM5)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1414))    CURRENT
RQMNAME(QM4)               STATUS(STOPPED)
SUBSTATE( )                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

d) 중지된 각 채널을 시작하십시오.

이 단계를 실행되고 있는 모든 채널에 대해 수행하십시오. 채널이 중지되지 않으면 **STOP CHANNEL** 명령을 FORCE 옵션과 함께 다시 실행할 수 있습니다. FORCE 옵션 설정의 예는 채널이 중지되지 않고 채널을 동기화하기 위해 다른 큐 관리자를 재시작할 수 없는 경우입니다.

```
*... On QM1
START CHANNEL(CL2.QM5)
```

응답은 명령이 승인됨입니다.

```
AMQ8018: Start IBM MQ channel accepted.
```

e) 전환되는 전송 큐를 모니터하십시오.

메시지 " `AMQ7341 The transmission queue for channel CL2.QM3 is SYSTEM.CLUSTER.TRANSMIT.` *QUEUE|CL2.QM3* ".

f) `SYSTEM.CLUSTER.TRANSMIT.QUEUE`가 더 이상 사용되지 않는지 확인하십시오.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
DISPLAY QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE) CURDEPTH
```

응답은 채널 상태 보고서 및 `SYSTEM.CLUSTER.TRANSMIT.QUEUE`의 용량 목록입니다.

```
AMQ8420: Channel Status not found.
AMQ8409: Display Queue details.
QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE)    TYPE(QLOCAL)
CURDEPTH(0)
```

g) 어떤 채널이 시작되는지 모니터하십시오.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

응답은 이 경우 새 기본 클러스터 전송 큐에서 이미 실행 중인 채널의 목록입니다.

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1412))   CURRENT
RQMNAME(QM2)                STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL1.QM2)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)            CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))    CURRENT
RQMNAME(QM3)                            STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL2.QM3)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1415))                CURRENT
RQMNAME(QM5)                            STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL2.QM5)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)           CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1414))    CURRENT
RQMNAME(QM4)                STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL1.QM4)
```

다음에 수행할 작업

1. 큐 알리어스 정의 Q1A로 큐 이름을 해석하고 QM2에서 QM3의 Q1으로 메시지를 송신하여 자동 정의된 클러스터 전송 큐를 테스트하십시오.
   a. 메시지를 넣으려면 QM2에서 샘플 프로그램 **amqsput**를 실행하십시오.

   ```
   C:\IBM\MQ>amqsput Q1A QM2
   Sample AMQSPUT0 start
   target queue is Q1A
   Sample request message from QM2 to Q1 using Q1A

   Sample AMQSPUT0 end
   ```

   b. QM3의 Q1에서 메시지를 가져오려면 샘플 프로그램 **amqsget**를 실행하십시오.

   ```
   C:\IBM\MQ>amqsget Q1 QM3
   Sample AMQSGET0 start
   message <Sample request message from QM2 to Q1 using Q1A>
   no more messages
   Sample AMQSGET0 end
   ```

2. 클러스터 큐에 대한 메시지가 시작되는 큐 관리자에서 클러스터 큐에 대한 보안을 구성하여 보안을 재구성할지 여부를 고려해보십시오.

관련 태스크

게이트웨이 큐 관리자에서 송신된 메시지를 격리하기 위한 리모트 큐 정의 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 클러스터된 큐 리모트 정의 및 개별 송신자 채널과 전송 큐를 사용합니다.

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 전송 큐 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 이 솔루션은 추가 클러스터 전송 큐를 사용하여 클러스터의 단일 큐 관리자로 메시지 트래픽을 분리합니다.

게이트웨이 큐 관리자에서 송신된 클러스터 메시지 트래픽을 격리하기 위한 클러스터 및 클러스터 전송 큐 추가
게이트웨이 큐 관리자를 사용하는 겹치는 클러스터의 구성을 수정하십시오. 게이트웨이 큐 관리자를 사용하는 중첩 클러스터의 구성을 수정합니다. 솔루션은 추가 클러스터를 사용하여 메시지를 특정 클러스터 큐로 분리합니다.

239 페이지의 『클러스터에 큐 관리자 추가: 개별 전송 큐』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 다중 클러스터 전송 큐를 사용하여 전송됩니다.

관련 정보

액세스 제어 및 다중 클러스터 전송 큐
클러스터링: 다중 클러스터 전송 큐를 사용하여 애플리케이션 격리
클러스터링: 클러스터 전송 큐를 구성하는 방법을 계획

큐 관리자에서 클러스터 큐 제거

Toronto에서 INVENTQ 큐를 사용 안함으로 설정합니다. 모든 인벤토리 메시지를 New York으로 송신하고 Toronto에서 비어 있는 INVENTQ 큐를 삭제합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 243 페이지의 『큐를 호스팅하는 큐 관리자 추가』에서 설명된 대로 설정되었습니다. 네 개의 큐 관리자를 포함합니다. LONDON 및 NEWYORK 둘 다 전체 저장소를 보유합니다. PARIS 및 TORONTO는 부분 저장소를 보유합니다. 인벤토리 애플리케이션은 New York 및 Toronto에 있는 시스템에서 실행되며 INVENTQ 큐에 메시지가 도착하면 구동됩니다.
- 감소된 워크로드로 인해 Toronto에서 인벤토리 애플리케이션을 더 이상 실행하지 않으려고 합니다. 큐 관리자 TORONTO가 호스팅하는 INVENTQ 큐를 사용 안함으로 설정하고 TORONTO가 NEWYORK의 INVENTQ 큐에 메시지를 공급하게 하려고 합니다.
- 네 시스템 사이에는 모두 네트워크 연결성이 있습니다.
- 네트워크 프로토콜은 TCP입니다.

이 태스크 정보

다음 단계에 따라 클러스터 큐를 제거하십시오.

프로시저

1. 큐가 더 이상 사용 가능하지 않음을 표시하십시오.

   클러스터에서 큐를 제거하려면 로컬 큐 정의에서 클러스터 이름을 제거하십시오. TORONTO에서 INVENTQ를 변경하여 클러스터의 나머지 부분에서 액세스할 수 없도록 하십시오.

   ```
   ALTER QLOCAL(INVENTQ) CLUSTER(' ')
   ```
2. 큐가 더 이상 사용 가능하지 않은지 확인하십시오.

   전체 저장소 큐 관리자 LONDON 또는 NEWYORK에서 다음 명령을 발행하여 큐가 큐 관리자 TORONTO에 의해 더 이상 호스팅되지 않는지 확인하십시오.

   ```
   DIS QCLUSTER (INVENTQ)
   ```

   ALTER 명령이 성공적으로 완료된 경우 TORONTO가 결과에 나열되지 않습니다.
3. 큐를 사용 안함으로 설정하십시오.

   추가 메시지가 기록되지 않도록 TORONTO에서 INVENTQ 큐를 사용 안함으로 설정하십시오.

   ```
   ALTER QLOCAL(INVENTQ) PUT(DISABLED)
   ```

   이제 MQOO_BIND_ON_OPEN을 사용하여 이 큐에 전송 중인 메시지가 데드-레터 큐로 이동합니다. 모든 애플리케이션이 이 큐 관리자의 큐에 명시적으로 메시지를 넣지 못하도록 해야 합니다.
4. 큐가 비워질 때까지 큐를 모니터하십시오.

   DISPLAY QUEUE 명령을 사용하고 IPPROCS, OPPROCS 및 CURDEPTH 속성을 지정하여 큐를 모니터하거나 IBM i에서 **WRKMQMSTS** 명령을 사용하십시오. 입력 및 출력 프로세스의 수와 큐의 현재 용량이 0인 경우 이 큐는 비어 있는 큐입니다.
5. 채널을 모니터하여 인다우트(in-doubt) 메시지가 없는지 확인하십시오.

   채널 INVENTORY.TORONTO에 인다우트(in-doubt) 메시지가 없는지 확인하려면 다른 큐 관리자 각각에서 INVENTORY.TORONTO라는 클러스터 송신자 채널을 모니터하십시오. 각 큐 관리자에서 INDOUBT 매개변수를 지정하여 DISPLAY CHSTATUS 명령을 발행하십시오.

   ```
   DISPLAY CHSTATUS(INVENTORY.TORONTO) INDOUBT
   ```

   인다우트(in-doubt) 메시지가 있는 경우 진행하기 전에 이 메시지를 해결해야 합니다. 예를 들어, RESOLVE 채널 명령 발행 또는 채널 중지 및 재시작을 시도할 수 있습니다.
6. 로컬 큐를 삭제하십시오.

TORONTO에서 인벤토리 애플리케이션에 전달할 메시지가 더 이상 없는 것으로 확인되면 큐를 삭제할 수 있습니다.

```
DELETE QLOCAL(INVENTQ)
```

7. 이제 Toronto의 시스템에서 인벤토리 애플리케이션을 제거할 수 있습니다.

   이 애플리케이션을 제거하면 중복을 피하고 시스템 공간을 확보할 수 있습니다.

결과

이 태스크에서 설정되는 클러스터는 이전 태스크에서 설정된 클러스터와 유사합니다. 차이점은 큐 관리자 TORONTO에서 INVENTQ 큐가 더 이상 사용 가능하지 않다는 점입니다.

단계 1에서 큐를 서비스에서 제거할 때 TORONTO 큐 관리자가 메시지를 두 개의 전체 저장소 큐 관리자에 송신했습니다. 이 메시지는 큐 관리자에게 상태가 변경되었음을 알렸습니다. 전체 저장소 큐 관리자는 INVENTQ에 관한 정보 업데이트를 요청한 클러스터에 있는 다른 큐 관리자에게 이 정보를 전달합니다.

큐 관리자가 메시지를 INVENTQ 큐에 넣으면 업데이트된 부분 저장소가 INVENTQ 큐는 NEWYORK 큐 관리자에서만 사용 가능함을 표시합니다. 이 메시지는 NEWYORK 큐 관리자에게 송신됩니다.

다음에 수행할 작업

이 태스크에는 제거할 한 개의 큐와 이 큐를 제거할 한 개의 클러스터만 있습니다.

여러 클러스터 이름이 들어 있는 이름 목록을 참조하는 다수의 큐가 있는 것으로 가정하십시오. 예를 들어, TORONTO 큐 관리자는 INVENTQ 외에도 PAYROLLQ, SALESQ 및 PURCHASESQ를 호스팅할 수 있습니다. TORONTO는 모든 해당 클러스터 INVENTORY, PAYROLL, SALES 및 PURCHASES에서 이러한 큐를 사용 가능하게 합니다. TORONTO 큐 관리자에서 클러스터 이름의 이름 목록을 정의하십시오.

```
DEFINE NAMELIST(TOROLIST)
DESCR('List of clusters TORONTO is in')
NAMES(INVENTORY, PAYROLL, SALES, PURCHASES)
```

각 큐 정의에 이름 목록을 추가하십시오.

```
DEFINE QLOCAL(INVENTQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PAYROLLQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(SALESQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PURCHASESQ) CLUSNL(TOROLIST)
```

SALES 조작이 PURCHASES 조작에 의해 대체되기 때문에 SALES 클러스터에서 모든 큐를 제거하려고 하는 것으로 가정합니다. 수행해야 하는 유일한 작업은 TOROLIST 이름 목록에서 SALES 클러스터의 이름을 제거하기 위해 이 이름 목록을 대체하는 것입니다.

이름 목록에 있는 클러스터 중 하나에서 단일 큐를 제거하려는 경우 나머지 클러스터 이름 목록을 포함하는 이름 목록을 작성하십시오. 그런 다음 새 이름 목록을 사용하도록 큐 정의를 대체하십시오. INVENTORY 클러스터에서 PAYROLLQ를 제거하려면 다음을 수행하십시오.

1. 다음과 같이 이름 목록을 작성하십시오.

   ```
   DEFINE NAMELIST(TOROSHORTLIST)
   DESCR('List of clusters TORONTO is in other than INVENTORY')
   NAMES(PAYROLL, SALES, PURCHASES)
   ```

2. 다음과 같이 PAYROLLQ 큐 정의를 대체하십시오.

   ```
   ALTER QLOCAL(PAYROLLQ) CLUSNL(TOROSHORTLIST)
   ```

클러스터에서 큐 관리자 제거: 우수 사례

큐 관리자가 클러스터에서 일반적으로 하나 이상의 전체 저장소와 통신할 수 있는 시나리오에서 클러스터로부터 큐 관리자를 제거합니다.

시작하기 전에

이 메소드는 최소 하나 이상의 전체 저장소가 사용 가능하고 이러한 저장소에 제거할 큐 관리자가 접속할 수 있는 시나리오에 대한 우수 사례입니다. 이 메소드에는 최소한의 수동 개입이 포함되고 큐 관리자가 클러스터에서 제어된 방식으로 탈퇴하는 것을 협상할 수 있습니다. 제거할 큐 관리자가 전체 저장소에 접속할 수 없는 경우 284 페이지의 『클러스터에서 큐 관리자 제거: 대체 메소드』의 내용을 참조하십시오.

이 태스크 정보

이 태스크 예에서는 INVENTORY 클러스터에서 큐 관리자 LONDON을 제거합니다. INVENTORY 클러스터는 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되고 280 페이지의 『큐 관리자에서 클러스터 큐 제거』에서 설명된 대로 수정됩니다.

클러스터에서 큐 관리자를 제거하기 위한 프로세스는 큐 관리자를 추가하는 프로세스보다 복잡합니다.

큐 관리자가 클러스터에 조인하면 기존 클러스터 멤버는 새 큐 관리자에 대한 지식이 없기 때문에 새 큐 관리자와 상호작용하지 않습니다. 전체 저장소에 연결할 수 있도록 조인한 큐 관리자에 대해 새 송신자 및 수신자 채널이 작성되어야 합니다.

클러스터에서 큐 관리자가 제거되면 이 큐 관리자에 연결된 애플리케이션이 클러스터 외에서 호스팅되는 큐와 같은 오브젝트를 사용할 가능성이 높습니다. 또한 클러스터 내 다른 큐 관리자에 연결되는 애플리케이션은 대상 큐 관리자에서 호스팅되는 오브젝트를 사용할 수 있습니다. 이러한 애플리케이션의 결과로 현재 큐 관리자는 클러스터에 조인할 때 사용했던 전체 저장소 외 클러스터 멤버와의 통신을 설정하기 위해 추가 송신자 채널을 작성할 수 있습니다. 클러스터 내 모든 큐 관리자에는 다른 클러스터 멤버를 설명하는 데이터의 캐시된 사본이 있습니다. 여기에는 제거 중인 것도 포함될 수 있습니다.

프로시저

1. 클러스터에서 큐 관리자를 제거하기 전에 이 큐 관리자가 클러스터에 필요한 자원을 더 이상 호스팅하지 않음을 확인하십시오.
   - 큐 관리자가 전체 저장소를 호스팅하는 경우 248 페이지의 『다른 큐 관리자로 전체 저장소 이동』의 단계 1 - 6을 완료하십시오. 제거할 큐 관리자의 전체 저장소 기능을 다른 큐 관리자로 이동하지 않을 경우 단계 5와 6만 완료하면 됩니다.
   - 큐 관리자가 클러스터 큐를 호스팅하는 경우 280 페이지의 『큐 관리자에서 클러스터 큐 제거』의 단계 1 - 7을 완료하십시오.
   - 큐 관리자가 클러스터 토픽을 호스팅하는 경우 이 토픽을 삭제하거나(예를 들어, DELETE TOPIC 명령을 사용하여) 333 페이지의 『다른 큐 관리자로 클러스터 토픽 정의 이동』에 설명된 대로 다른 호스트로 이동하십시오.

     참고: 클러스터에서 큐 관리자를 제거하고 이 큐 관리자가 여전히 클러스터 토픽을 호스팅하는 경우 이 큐 관리자는 이 토픽이 삭제될 때까지 계속해서 클러스터에 남겨진 발행물을 큐 관리자에게 전달하려고 시도할 수 있습니다.
2. 수동으로 정의된 클러스터 수신자 채널을 대체하여 큐 관리자 LONDON에 있는 클러스터로부터 제거하십시오.

   ```
   ALTER CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) CLUSTER(' ')
   ```
3. 수동으로 정의된 클러스터 송신자 채널을 대체하여 큐 관리자 LONDON에 있는 클러스터로부터 제거하십시오.

   ```
   ALTER CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) CLUSTER(' ')
   ```

클러스터 내 다른 큐 관리자는 이 큐 관리자 및 해당 클러스터 자원이 더 이상 클러스터에 속하지 않음을 알게 됩니다.

4. 클러스터의 전체 저장소로 플로우되기 위해 대기 중인 메시지가 없을 때까지 큐 관리자 LONDON에서 클러스터 전송 큐를 모니터하십시오.

```
DISPLAY CHSTATUS(INVENTORY.PARIS) XQMSGSA
```

메시지가 전송 큐에 남아 있는 경우 계속하기 전에 메시지가 PARIS 및 NEWYORK 전체 저장소로 송신되지 않는 이유를 판별하십시오.

결과

큐 관리자 LONDON은 더 이상 클러스터에 속하지 않습니다. 그러나 이 큐 관리자는 여전히 독립적 큐 관리자로 작동할 수 있습니다.

다음에 수행할 작업

이러한 변경의 결과는 클러스터의 나머지 멤버에서 다음 명령을 발행하여 확인할 수 있습니다.

```
DISPLAY CLUSQMGR(LONDON)
```

큐 관리자에 대해 자동 정의된 클러스터 송신자 채널이 중지될 때까지 큐 관리자는 계속해서 표시됩니다. 이러한 중지가 발생할 때까지 기다리거나 다음 명령을 발행하여 활성 인스턴스에 대해 계속해서 모니터할 수 있습니다.

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

이 큐 관리자로 메시지가 더 이상 전달되지 않는 것이 확실하면 클러스터의 나머지 멤버에서 다음 명령을 발행하여 LONDON에 대한 클러스터 송신자 채널을 중지할 수 있습니다.

```
STOP CHANNEL(INVENTORY.LONDON) STATUS(INACTIVE)
```

변경이 클러스터 전체에 전파되고 이 큐 관리자에 더 이상 메시지가 전달되지 않으면 LONDON에서 CLUSRCVR 채널을 중지하고 삭제하십시오.

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
```

237 페이지의 『클러스터에 큐 관리자 추가』에 설명된 대로 제거된 큐 관리자는 나중에 해당 클러스터에 다시 추가할 수 있습니다. 제거된 큐 관리자는 최대 90일 동안 클러스터의 나머지 멤버에 대한 지식을 계속해서 캐시합니다. 이 캐시가 만료될 때까지 대기하는 것을 원치 않는 경우 287 페이지의 『큐 관리자를 해당 클러스터 전 상태로 복원』에서 설명된 대로 강제로 제거할 수 있습니다.

관련 태스크

클러스터에서 큐 관리자 제거 (IBM MQ Explorer사용)

관련 참조

ALTER CHANNEL(채널 설정 변경)

DISPLAY CHANNEL(채널 정의 표시)

DISPLAY CHSTATUS(채널 상태 표시)

DISPLAY CLUSQMGR(클러스터 큐 관리자의 채널 정보 표시)

STOP CHANNEL(채널 중지)

클러스터에서 큐 관리자 제거*:* 대체 메소드

중요한 시스템 또는 구성 문제로 인해 큐 관리자가 클러스터에서 전체 저장소와 통신할 수 없는 시나리오에서 클러스터로부터 큐 관리자를 제거합니다.

시작하기 전에

클러스터에서 큐 관리자를 수동으로 제거하는 이 대체 메소드는 제거된 큐 관리자를 클러스터에 링크하는 모든 클러스터 채널을 중지 및 삭제하고 클러스터에서 이 큐 관리자를 강제로 제거합니다. 이 메소드는 제거할 큐 관리자가 모든 전체 저장소와 통신할 수 없는 시나리오에서 사용됩니다. 이는 예를 들면, 큐 관리자가 작업을 중지했거나 큐 관리자와 클러스터 사이에 지속적 통신 실패가 있기 때문일 수 있습니다. 그렇지 않은 경우 가장 일반적인 메소드를 사용하십시오. 283 페이지의 『클러스터에서 큐 관리자 제거: 우수 사례』.

이 태스크 정보

이 태스크 예에서는 INVENTORY 클러스터에서 큐 관리자 LONDON을 제거합니다. INVENTORY 클러스터는 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되고 280 페이지의 『큐 관리자에서 클러스터 큐 제거』에서 설명된 대로 수정됩니다.

클러스터에서 큐 관리자를 제거하기 위한 프로세스는 큐 관리자를 추가하는 프로세스보다 복잡합니다.

큐 관리자가 클러스터에 조인하면 기존 클러스터 멤버는 새 큐 관리자에 대한 지식이 없기 때문에 새 큐 관리자와 상호작용하지 않습니다. 전체 저장소에 연결할 수 있도록 조인한 큐 관리자에 대해 새 송신자 및 수신자 채널이 작성되어야 합니다.

클러스터에서 큐 관리자가 제거되면 이 큐 관리자에 연결된 애플리케이션이 클러스터 외에서 호스팅되는 큐와 같은 오브젝트를 사용할 가능성이 높습니다. 또한 클러스터 내 다른 큐 관리자에 연결되는 애플리케이션은 대상 큐 관리자에서 호스팅되는 오브젝트를 사용할 수 있습니다. 이러한 애플리케이션의 결과로 현재 큐 관리자는 클러스터에 조인할 때 사용했던 전체 저장소 외 클러스터 멤버와의 통신을 설정하기 위해 추가 송신자 채널을 작성할 수 있습니다. 클러스터 내 모든 큐 관리자에는 다른 클러스터 멤버를 설명하는 데이터의 캐시된 사본이 있습니다. 여기에는 제거 중인 것도 포함될 수 있습니다.

이 프로시저는 큐 관리자가 별다른 문제 없이 클러스터에서 떠날 때까지 대기할 수 없는 비상 시에 적절할 수 있습니다.

프로시저

1. 클러스터에서 큐 관리자를 제거하기 전에 이 큐 관리자가 클러스터에 필요한 자원을 더 이상 호스팅하지 않음을 확인하십시오.
   - 큐 관리자가 전체 저장소를 호스팅하는 경우 248 페이지의 『다른 큐 관리자로 전체 저장소 이동』의 단계 1 - 6을 완료하십시오. 제거할 큐 관리자의 전체 저장소 기능을 다른 큐 관리자로 이동하지 않을 경우 단계 5와 6만 완료하면 됩니다.
   - 큐 관리자가 클러스터 큐를 호스팅하는 경우 280 페이지의 『큐 관리자에서 클러스터 큐 제거』의 단계 1 - 7을 완료하십시오.
   - 큐 관리자가 클러스터 토픽을 호스팅하는 경우 이 토픽을 삭제하거나(예를 들어, DELETE TOPIC 명령을 사용하여) 333 페이지의 『다른 큐 관리자로 클러스터 토픽 정의 이동』에 설명된 대로 다른 호스트로 이동하십시오.

   참고: 클러스터에서 큐 관리자를 제거하고 이 큐 관리자가 여전히 클러스터 토픽을 호스팅하는 경우 이 큐 관리자는 이 토픽이 삭제될 때까지 계속해서 클러스터에 남겨진 발행물을 큐 관리자에게 전달하려고 시도할 수 있습니다.
2. 클러스터 내 다른 큐 관리자와 통신하는 데 사용되는 모든 채널을 중지하십시오. MODE(FORCE)를 사용하여 큐 관리자 LONDON에서 CLUSRCVR 채널을 중지하십시오. 그렇지 않은 경우 송신자 큐 관리자가 채널을 중지할 때까지 대기해야 될 수도 있습니다.

```
STOP CHANNEL(INVENTORY.LONDON) MODE(FORCE)
STOP CHANNEL(INVENTORY.TORONTO)
STOP CHANNEL(INVENTORY.PARIS)
STOP CHANNEL(INVENTORY.NEWYORK)
```

3. 채널이 중지될 때까지 큐 관리자 LONDON에서 채널 상태를 모니터하십시오.

```
DISPLAY CHSTATUS(INVENTORY.LONDON)
DISPLAY CHSTATUS(INVENTORY.TORONTO)
```

```
DISPLAY CHSTATUS(INVENTORY.PARIS)
DISPLAY CHSTATUS(INVENTORY.NEWYORK)
```

채널이 중지된 후에는 클러스터 내 다른 큐 관리자와 애플리케이션 메시지를 주고 받지 않습니다.

4. 큐 관리자 LONDON에서 수동으로 정의된 클러스터 채널을 삭제하십시오.

```
DELETE CHANNEL(INVENTORY.NEWYORK)
DELETE CHANNEL(INVENTORY.TORONTO)
```

5. 클러스터 내 나머지 큐 관리자는 제거된 큐 관리자에 대한 지식을 계속 보유하고 제거된 큐 관리자에게 계속해서 메시지를 송신할 수도 있습니다. 나머지 큐 관리자에서 이러한 지식을 영구 제거하려면 전체 저장소 중 하나의 클러스터에서 제거된 큐 관리자를 재설정하십시오.

```
RESET CLUSTER(INVENTORY) ACTION(FORCEREMOVE) QMNAME(LONDON) QUEUES(YES)
```

제거된 큐 관리자와 같은 이름을 가진 다른 큐 관리자가 클러스터에 있을 가능성이 있는 경우 제거된 큐 관리자의 **QMID**를 지정하십시오.

결과

큐 관리자 LONDON은 더 이상 클러스터에 속하지 않습니다. 그러나 이 큐 관리자는 여전히 독립적 큐 관리자로 작동할 수 있습니다.

다음에 수행할 작업

이러한 변경의 결과는 클러스터의 나머지 멤버에서 다음 명령을 발행하여 확인할 수 있습니다.

```
DISPLAY CLUSQMGR(LONDON)
```

큐 관리자에 대해 자동 정의된 클러스터 송신자 채널이 중지될 때까지 큐 관리자는 계속해서 표시됩니다. 이러한 중지가 발생할 때까지 기다리거나 다음 명령을 발행하여 활성 인스턴스에 대해 계속해서 모니터할 수 있습니다.

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

변경이 클러스터 전체에 전파되고 이 큐 관리자에 더 이상 메시지가 전달되지 않으면 LONDON에서 CLUSRCVR 채널을 삭제하십시오.

```
DELETE CHANNEL(INVENTORY.LONDON)
```

237 페이지의 『클러스터에 큐 관리자 추가』에 설명된 대로 제거된 큐 관리자는 나중에 해당 클러스터에 다시 추가할 수 있습니다. 제거된 큐 관리자는 최대 90일 동안 클러스터의 나머지 멤버에 대한 지식을 계속해서 캐시합니다. 이 캐시가 만료될 때까지 대기하는 것을 원치 않는 경우 287 페이지의 『큐 관리자를 해당 클러스터 전 상태로 복원』에서 설명된 대로 강제로 제거할 수 있습니다.

관련 참조

DELETE CHANNEL
DISPLAY CHANNEL
DISPLAY CHSTATUS
DISPLAY CLUSQMGR
STOP CHANNEL
RESET CLUSTER

큐 관리자를 해당 클러스터 전 상태로 복원

큐 관리자가 클러스터에서 제거될 때 큐 관리자는 나머지 클러스터 멤버에 대한 지식을 보유합니다. 이 지식은 결국 만료되어 자동으로 삭제됩니다. 그러나 이러한 지식을 즉시 삭제하고 싶은 경우에는 이 주제의 단계를 사용할 수 있습니다.

시작하기 전에

큐 관리자가 클러스터에서 제거되었고 클러스터에서 더 이상 작업을 수행하지 않는 것으로 가정합니다. 예를 들어, 해당 큐는 클러스터에서 더 이상 메시지를 수신하지 않고 애플리케이션은 이러한 큐에 메시지가 도착하는 것을 대기하지 않습니다.

이 태스크 정보

큐 관리자가 클러스터에서 제거될 때 큐 관리자는 최대 90일 동안 나머지 클러스터 멤버에 대한 지식을 보유합니다. 이는 시스템적 이점을 가질 수 있으며 특히 큐 관리자가 빠르게 클러스터에 다시 조인하는 경우에 유리합니다. 이 지식은 만료되면 자동으로 삭제됩니다. 그러나 이 정보를 수동으로 삭제하는 것을 선호하는 이유가 있습니다. 예를 들면 다음과 같습니다.

- 이전에 클러스터 자원을 사용했던 모든 애플리케이션이 이 큐 관리자에서 중지되었음을 확인하려고 합니다. 나머지 클러스터 멤버에 대한 지식이 만료될 때까지 그러한 애플리케이션은 계속해서 전송 큐에 기록합니다. 클러스터 지식이 삭제된 후에 그러한 애플리케이션이 클러스터 자원을 사용하려고 시도하면 시스템이 오류 메시지를 생성합니다.
- 큐 관리자에 대한 상태 정보를 표시할 때 나머지 클러스터 멤버에 대한 만기 정보는 표시하지 않으려고 합니다.

이 태스크는 INVENTORY 클러스터를 예로 사용합니다. 283 페이지의 『클러스터에서 큐 관리자 제거: 우수 사례』에서 설명된 대로 INVENTORY 클러스터에서 LONDON 큐 관리자가 제거되었습니다. 클러스터의 나머지 멤버에 대한 지식을 삭제하려면 LONDON 큐 관리자에서 다음 명령을 실행하십시오.

프로시저

1. 이 큐 관리자에서 클러스터 내 다른 큐 관리자의 모든 메모리를 제거하십시오.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

2. 모든 클러스터 자원이 없어질 때까지 큐 관리자를 모니터하십시오.

```
DISPLAY CLUSQMGR(*) CLUSTER(INVENTORY)
DISPLAY QCLUSTER(*) CLUSTER(INVENTORY)
DISPLAY TOPIC(*) CLUSTER(INVENTORY)
```

관련 정보

클러스터
클러스터링과 분산 큐잉의 비교
클러스터 컴포넌트

큐 관리자 유지보수

유지보수를 수행하기 위해 클러스터에서 큐 관리자를 일시중단하고 재개합니다.

이 태스크 정보

클러스터의 일부인 큐 관리자에서 유지보수를 수행해야 되는 경우가 있습니다. 예를 들어, 데이터의 백업을 해당 큐에 넣거나 수정사항을 소프트웨어에 적용해야 합니다. 큐 관리자가 큐를 호스팅하는 경우 활동을 일시중단해야 합니다. 유지보수가 완료되면 그 활동이 재개될 수 있습니다.

프로시저

1. SUSPEND QMGR **runmqsc** 명령을 발행하여 큐 관리자를 일시중단하십시오.

   ```
   SUSPEND QMGR CLUSTER(SALES)
   ```

   SUSPEND **runmqsc** 명령은 SALES 클러스터의 큐 관리자에게 이 큐 관리자가 일시중단되었음을 알립니다.

   SUSPEND QMGR 명령의 목적은 가능하면 이 큐 관리자에게 메시지를 송신하는 것을 피하도록 다른 큐 관리자에게 권고하는 것입니다. 이는 큐 관리자가 사용 불가능함을 의미하지는 않습니다. 이 큐 관리자가 핸들링해야 하는 일부 메시지가 여전히 이 큐 관리자에 송신됩니다. 예를 들어, 이 큐 관리자가 클러스터된 큐의 유일한 호스트인 경우입니다.

   큐 관리자가 일시중단된 경우 워크로드 관리 루틴이 이 큐 관리자에 대한 메시지 송신을 피합니다. 해당 큐 관리자가 핸들링해야 하는 메시지에는 로컬 큐 관리자가 송신하는 메시지가 포함됩니다.

   IBM MQ는 워크로드 밸런싱 알고리즘을 사용하여 가능할 때마다 로컬 큐 관리자를 선택하기 보다는 적합한 대상을 판별합니다.

   a) SUSPEND QMGR 명령에 FORCE 옵션을 사용하여 큐 관리자의 일시중단을 강제 실행하십시오.

      ```
      SUSPEND QMGR CLUSTER(SALES) MODE(FORCE)
      ```

      MODE(FORCE)는 클러스터 내 다른 큐 관리자에서 모든 인바운드 채널을 강제로 중지합니다. MODE(FORCE)를 지정하지 않으면 기본 MODE(QUIESCE)가 적용됩니다.

2. 필요한 유지보수 태스크를 수행하십시오.
3. RESUME QMGR **runmqsc** 명령을 발행하여 큐 관리자를 재개하십시오.

   ```
   RESUME QMGR CLUSTER(SALES)
   ```

결과

RESUME **runmqsc** 명령은 큐 관리자가 다시 사용 가능함을 전체 저장소에 알립니다. 전체 저장소 큐 관리자는 이 큐 관리자에 관한 정보 업데이트를 요청한 다른 큐 관리자에게 이 정보를 전파합니다.

클러스터 전송 큐 유지보수

클러스터 전송 큐를 사용 가능한 상태로 유지하기 위해 최선을 다해야 합니다. 클러스터 전송 큐는 클러스터의 성능에 필수적입니다. z/OS z/OS에서는 클러스터 전송 큐의 INDXTYPE을 CORRELID로 설정하십시오.

시작하기 전에

- 클러스터 전송 큐가 가득 차지 않도록 하십시오.
- 클러스터 전송 큐를 가져오기 사용 안함 또는 넣기 사용 안함으로 우발적으로 설정하지 않도록 ALTER **runmqsc** 명령을 발행하지 않게 주의하십시오.
- z/OS (예: z/OS 페이지 세트)에서 클러스터 전송 큐가 저장되는 매체가 가득 차지 않도록 하십시오.

이 태스크 정보

z/OS

다음 프로시저는 z/OS에만 적용할 수 있습니다.

프로시저

클러스터 전송 큐의 INDXTYPE을 CORRELID로 설정하십시오.

클러스터 큐 관리자 새로 고치기

REFRESH CLUSTER 명령을 사용하여 로컬 저장소에서 자동 정의된 채널 또는 자동 정의된 클러스터 오브젝트를 제거할 수 있습니다. 메시지가 손실되지 않습니다.

시작하기 전에

IBM Support Center에서 이 명령을 사용하도록 요청을 받을 수 있습니다. 이 명령은 매우 주의해서 사용해야 합니다. 예를 들어, 대형 클러스터의 경우 **REFRESH CLUSTER** 명령의 사용은 진행되는 동안 및 클러스터 오브젝트가 모든 관련 큐 관리자에게 자동으로 상태 업데이트를 전송한 후 27일 간격으로 클러스터에 지장을 줄 수 있습니다. 클러스터링: REFRESH CLUSTER 우수 사례 사용을 참조하십시오.

이 태스크 정보

큐 관리자는 클러스터에서 새롭게 시작할 수 있습니다. 정상적인 환경에서는 REFRESH CLUSTER 명령을 사용할 필요가 없습니다.

프로시저

큐 관리자에서 REFRESH CLUSTER **MQSC** 명령을 실행하여 로컬 저장소에서 자동 정의된 클러스터 큐 관리자 및 큐 오브젝트를 제거하십시오.

이 명령은 큐 관리자를 참조하는 오브젝트만 제거하고 로컬 큐 관리자와 관련된 오브젝트는 제거하지 않습니다. 이 명령은 자동 정의된 채널도 제거합니다. 이 명령은 클러스터 전송 큐에 메시지가 없고 전제 저장소 큐 관리자에 연결되지 않은 채널을 제거합니다.

결과

효과적으로 REFRESH CLUSTER 명령을 사용하면 큐 관리자가 전체 저장소 컨텐츠와 관련하여 콜드 스타트할 수 있습니다. IBM MQ은(는) 큐에서 데이터가 손실되지 않도록 합니다.

관련 정보

클러스터링: REFRESH CLUSTER 사용 우수 사례

클러스터 큐 관리자 복구

REFRESH CLUSTER **runmqsc** 명령을 사용하여 큐 관리자에 대한 클러스터 정보를 최신 정보로 고칩니다. 특정 시점 백업에서 큐 관리자를 복구한 후에 이 프로시저를 따릅니다.

시작하기 전에

특정 시점 백업에서 클러스터 큐 관리자를 복원했습니다.

이 태스크 정보

클러스터에서 큐 관리자를 복구하려면 큐 관리자를 복구한 다음 REFRESH CLUSTER **runmqsc** 명령을 사용하여 클러스터 정보를 최신 정보로 고치십시오.

참고: 대형 클러스터의 경우, **REFRESH CLUSTER** 명령을 사용하면 진행 중에 클러스터에 혼란을 줄 수 있으며, 클러스터 오브젝트가 모든 관심 있는 큐 관리자에 자동으로 상태 업데이트를 보낸 이후 27일 간격으로 다시 수행됩니다. 대형 클러스터를 새로 고치면 클러스터의 성능 및 가용성에 영향을 줄 수 있음을 참조하십시오.

프로시저

큐 관리자가 참여하는 모든 클러스터의 복원된 큐 관리자에서 REFRESH CLUSTER 명령을 발행하십시오.

다음에 수행할 작업

다른 모든 큐 관리자에서 REFRESH CLUSTER 명령을 발행하지 않아도 됩니다.

관련 정보
클러스터링: REFRESH CLUSTER 사용 우수 사례

가용성을 위한 클러스터 채널 구성

우수 구성 사례에 따라 간헐적 네트워크 중지가 발생할 경우 클러스터 채널이 계속해서 원활하게 실행될 수 있도록 합니다.

시작하기 전에

클러스터는 채널 정의의 필요성을 줄여주지만 여전히 채널을 유지보수해야 합니다. 분산 큐잉에서 사용된 것과 같은 채널 기술이 클러스터 내 큐 관리자 간 통신에 사용됩니다. 클러스터 채널에 대해 이해하려면 다음과 같은 사항을 익혀야 합니다.

- 채널이 작동하는 방식
- 채널의 상태를 찾는 방법
- 채널 엑시트를 사용하는 방법

이 태스크 정보

다음과 같은 점을 특별히 고려하려고 합니다.

프로시저

클러스터 채널을 구성할 때 다음과 같은 점을 고려하십시오.

- 클러스터 송신자 채널 및 클러스터 수신자 채널에서 네트워크에 다량의 하트비트로 부담을 주지 않거나 활성 플로우를 유지하는 HBINT 또는 KAINT의 값을 선택하십시오. 때때로 네트워크가 느려지고 이 길이의 지연이 생기는 경우 간격이 약 10초 미만이면 False 실패가 발생합니다.
- 실패된 채널에서 인다우트(in-doubt)이기 때문에 고립된 메시지를 발생시키는 범위를 줄이려면 BATCHHB 값을 설정하십시오. 배치를 채우는 데 더 긴 시간이 제공되는 경우 실패한 채널의 인다우트(in-doubt) 배치가 발생할 가능성이 높아집니다. 채널의 메시지 트래픽이 메시지의 버스트 간 긴 시간 동안 산발적이면 실패한 배치가 발생할 가능성이 높습니다.
- 채널의 클라이언트 송신자 측이 실패한 후 하트비트 또는 활성 유지가 실패를 감지하기 전에 재시작을 시도할 경우 문제점이 발생합니다. 채널의 클러스터 수신자 측이 활성으로 유지되는 경우 채널 송신자 재시작이 거부됩니다. 실패를 피하려면 클러스터 송신자 채널이 재시작을 시도하면 클러스터 수신자 채널이 종료되고 재시작되도록 설정하십시오.

  z/OS **IBM MQ for z/OS**
  ALTER QMGR에서 ADOPTMCA 및 ADOPTCHK 매개변수를 사용하여 활성 상태인 채널의 클러스터 수신자 끝 문제점을 제어하십시오.

  Multi **멀티플랫폼**
  qm.ini 파일 또는 Windows NT 레지스트리의 AdoptNewMCA, AdoptNewMCATimeout및 AdoptNewMCACheck 속성을 사용하여 활성 상태인 채널의 클러스터 수신자 끝 문제점을 제어하십시오.

메시지를 클러스터로(부터) 라우팅

큐 알리어스, 큐 관리자 알리어스 및 리모트 큐 정의를 사용하여 클러스터를 외부 큐 관리자 및 다른 클러스터에 연결합니다.

메시지를 클러스터로(부터) 라우팅하는 것에 대한 자세한 내용은 다음 하위 주제를 참조하십시오.

관련 개념
304 페이지의 『큐 관리자 알리어스 및 클러스터』
큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

307 페이지의 『큐 알리어스 및 클러스터』

큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

306 페이지의 『응답 대상 큐 알리어스 및 클러스터』
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

관련 태스크

216 페이지의 『큐 관리자 클러스터 구성』
클러스터는 초기 구성 및 진행 중인 관리를 모두 단순화하는 방식으로 큐 관리자를 상호 연결하기 위한 메커니즘을 제공합니다. 클러스터 컴포넌트를 정의하고 클러스터를 작성 및 관리할 수 있습니다.

226 페이지의 『새 클러스터 설정』
다음 지시사항에 따라 클러스터 예를 설정합니다. 개별 지시사항에서는 TCP/IP, LU 6.2 및 단일 전송 큐 또는 다중 전송 큐에서의 클러스터 설정을 설명합니다. 한 큐 관리자에서 다른 큐 관리자로 메시지를 송신하여 클러스터가 작동하는지 테스트합니다.

관련 정보

클러스터

클러스터링과 분산 큐잉의 비교

클러스터의 컴포넌트

클러스터에 대한 요청/응답 구성

클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

시작하기 전에

292 페이지의 그림 54에서는 DEMO라고 하는 클러스터 외부에 있는 QM3라고 하는 큐 관리자를 보여줍니다. QM3는 클러스터를 지원하지 않는 IBM MQ 제품에서 큐 관리자일 수 있습니다. QM3는 Q3라고 하는 큐를 호스팅하며 다음과 같이 정의됩니다.

```
DEFINE QLOCAL(Q3)
```

클러스터 내부에는 QM1 및 QM2라고 하는 두 큐 관리자가 있습니다. QM2는 Q2라고 하는 클러스터 큐를 호스팅하며 다음과 같이 정의됩니다.

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO)
```

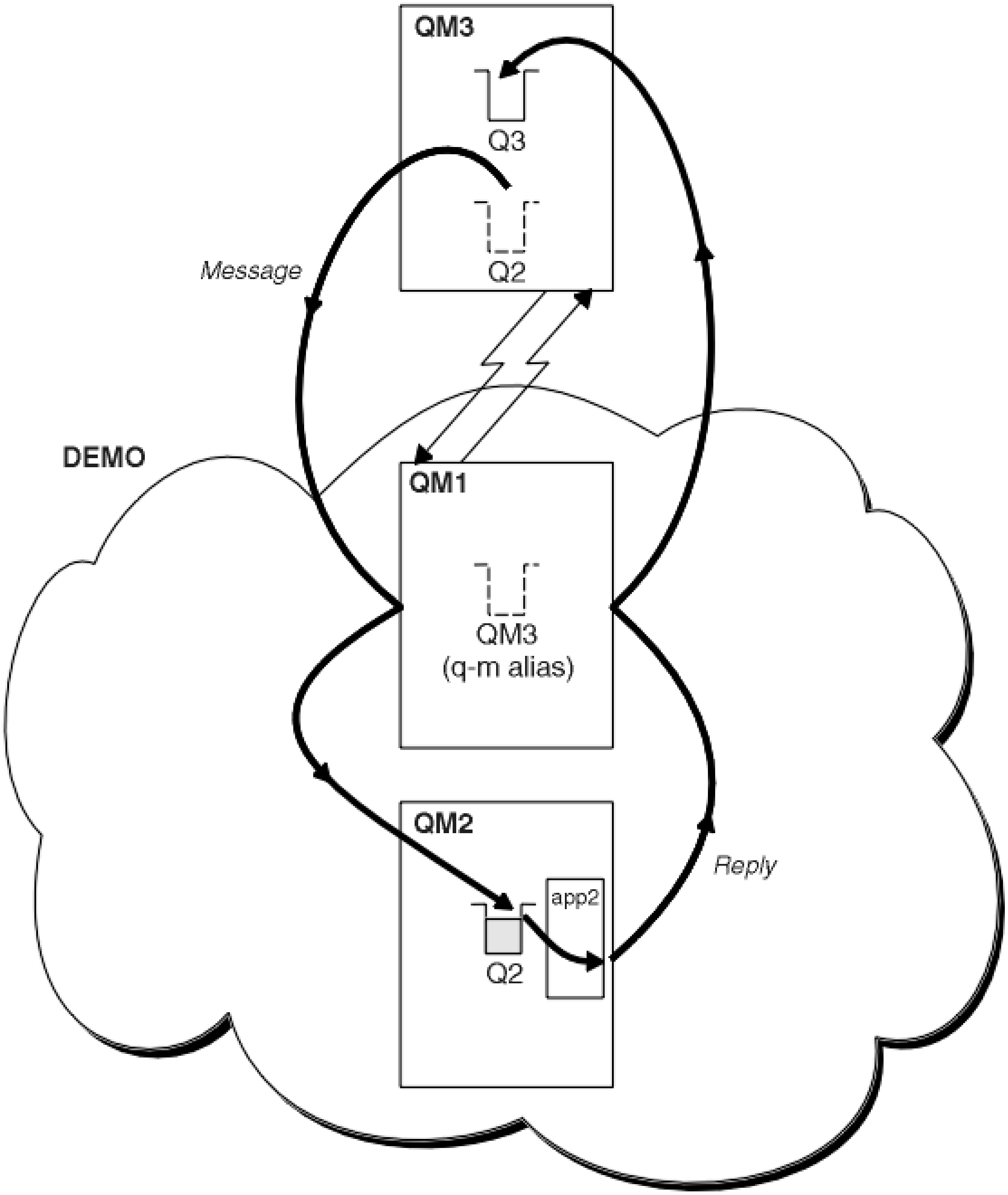


그림 *54.* 클러스터 외부 큐 관리자에서 넣기

이 태스크 정보

프로시저의 권고사항에 따라 요청 및 응답 메시지에 대한 경로를 설정하십시오.

프로시저

1. 요청 메시지를 클러스터에 송신하십시오.

   클러스터 외부에 있는 큐 관리자가 클러스터 내부에 있는 QM2의 큐 Q2에 메시지를 넣는 방법을 고려하십시오. 클러스터 외부에 있는 큐 관리자에는 이 큐 관리자가 메시지를 넣는 클러스터 내 각 큐에 대한 QREMOTE 정의가 있어야 합니다.

   a) QM3에서 Q2에 대한 리모트 큐를 정의하십시오.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(QM2) XMITQ(QM1)
```

QM3는 클러스터의 일부가 아니므로 분산 큐잉 기술을 사용하여 통신해야 합니다. 따라서 송신자 채널 및 QM1에 대한 전송 큐도 있어야 합니다. QM1에는 상응하는 수신자 채널이 필요합니다. 채널 및 전송 큐는 292 페이지의 그림 54에 명시적으로 표시되지 않습니다.

예에서 QM3에 있는 애플리케이션은 MQPUT 호출을 발행하여 메시지를 Q2에 넣습니다. QREMOTE 정의로 인해 메시지는 QM1 전송 큐에서 메시지를 가져오는 송신자 채널을 사용하여 QM2의 Q2에 라우트됩니다.

2. 클러스터에서 응답 메시지를 수신하십시오.

큐 관리자 알리어스를 사용하여 클러스터 외부 큐 관리자에 대한 응답의 리턴 경로를 작성하십시오. 게이트웨이 QM1은 클러스터 외부에 있는 큐 관리자 QM3의 큐 관리자 알리어스를 알립니다. 또한 클러스터 속성을 QM3에 대한 큐 관리자 알리어스 정의에 추가하여 클러스터 내부에 있는 큐 관리자에게 QM3를 알립니다. 큐 관리자 알리어스 정의는 리모트 큐 정의와 같지만 공백 RNAME이 있습니다.

a) QM1의 QM3에 대한 큐 관리자 알리어스를 정의하십시오.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

QM1에서 QM3로 응답을 다시 전달하는 데 사용되는 전송 큐의 이름 선택에 대해 고려해야 합니다. XMITQ 속성의 생략에 의해 QREMOTE 정의에서 암시적이며 전송 큐의 이름은 QM3입니다. 그러나 QM3은 큐 관리자 알리어스를 사용하여 나머지 클러스터에 광고할 것으로 예상되는 동일한 이름입니다. IBM MQ에서는 동일한 이름의 전송 큐와 큐 관리자 알리어스를 둘 다 제공할 수 없습니다. 한 가지 솔루션은 큐 관리자 알리어스에 다른 이름을 사용하여 QM3에 메시지를 전달하는 전송 큐를 작성하는 것입니다.

b) QREMOTE 정의에 전송 큐 이름을 제공하십시오.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO) XMITQ(QM3.XMIT)
```

새 큐 관리자 알리어스는 `QM3.XMIT`라고 하는 새 전송 큐와 QM3 큐 관리자 알리어스를 연결합니다. 이는 단순하고 정확한 솔루션이지만 완전히 만족스러운 솔루션은 아닙니다. 이 솔루션은 대상 큐 관리자와 같은 이름이 지정되는 전송 큐에 대한 이름 지정 규칙을 위반했습니다. 전송 큐 이름 지정 규칙을 준수하는 대체 솔루션이 있습니까?

QM3에서 송신되는 요청 메시지에 요청자가 응답 대상 큐 관리자 이름으로 QM3를 전달하도록 기본값 지정되었기 때문에 문제점이 발생합니다. QM2의 서버는 QM3 응답 대상 큐 관리자 이름을 사용하여 그 응답에서 QM3를 처리합니다. 이 솔루션에서 QM1은 QM3를 응답 메시지를 리턴할 큐 관리자 알리어스로 알려야 하고 QM1이 QM3를 전송 큐의 이름으로 사용하는 것을 방지해야 합니다.

QM3를 응답 대상 큐 관리자 이름으로 제공하는 것으로 기본값 설정하는 대신 QM3의 애플리케이션은 응답 대상 큐 관리자 알리어스를 응답 메시지에 대한 QM1에 전달해야 합니다. 게이트웨이 큐 관리자 QM1은 전송 큐 이름과의 충돌을 피하면서 큐 관리자 알리어스를 QM3 자체가 아닌 QM3에 대한 응답을 위해 알립니다.

c) QM1의 QM3에 대한 큐 관리자 알리어스를 정의하십시오.

```
DEFINE QREMOTE(QM3.ALIAS) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

구성 명령에 대한 두 가지 변경이 필요합니다.

i) QM1의 QREMOTE는 이제 큐 관리자 알리어스 `QM3.ALIAS`를 실제 큐 관리자 QM3의 이름에 연결하여 클러스터의 나머지 부분에 알립니다. QM3은 다시 QM3로 응답 큐를 전송하는 전송 큐의 이름입니다.

ii) 클라이언트 애플리케이션은 요청 메시지를 구성할 때 응답 대상 큐 관리자의 이름으로 `QM3.ALIAS`를 제공해야 합니다. 다음 두 가지 방식 중 하나로 `QM3.ALIAS`를 클라이언트 애플리케이션에 제공할 수 있습니다.

- MQMD에서 MQPUT에 의해 구성된 응답 대상 큐 관리자 이름 필드에서 `QM3.ALIAS`를 코드화하십시오. 응답에 동적 큐를 사용하는 경우 이 방법으로 이를 수행해야 합니다.

- 응답 대상 큐 이름을 제공할 때 응답 대상 큐가 아닌 응답 대상 큐 알리어스 Q3.ALIAS를 사용하십시오.

```
DEFINE QREMOTE(Q3.ALIAS) RNAME(Q3) RQMNAME(QM3.ALIAS)
```

다음에 수행할 작업

참고: AMQSREQ0와 함께 응답 대상 큐 알리어스를 사용하는 것을 보여줄 수 없습니다. 이는 매개변수 3 또는 기본 SYSTEM.SAMPLE.REPLY 모델 큐에서 제공된 큐 이름을 사용하여 응답 대상 큐를 엽니다. MQPUT에 대한 응답 대상 큐 관리자 알리어스의 이름을 지정하려면 응답 대상 큐 알리어스를 포함하는 다른 매개변수를 제공하여 샘플을 수정해야 합니다.

관련 개념

큐 관리자 알리어스 및 클러스터
큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

응답 대상 큐 알리어스 및 클러스터
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

큐 알리어스 및 클러스터
큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

관련 태스크

클러스터에서 요청/응답 구성
클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

클러스터 외부에서 워크로드 밸런싱 구성
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

클러스터 간 메시지 경로 구성
게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

294 페이지의 『클러스터 대상 큐 관리자의 이름 숨기기』
큐 관리자의 이름을 지정하지 않고 클러스터 내 임의의 큐 관리자에서 정의되는 클러스터 큐에 메시지를 라우트합니다.

클러스터 대상 큐 관리자의 이름 숨기기
큐 관리자의 이름을 지정하지 않고 클러스터 내 임의의 큐 관리자에서 정의되는 클러스터 큐에 메시지를 라우트합니다.

시작하기 전에

- 클러스터 내부에 있는 큐 관리자의 이름을 클러스터 외부에 있는 큐 관리자에 노출하지 않도록 하십시오.
  - 클러스터 내부에서 큐를 호스팅하는 큐 관리자에 대한 참조를 해석하면 워크로드 밸런싱을 수행하기 위한 유연성이 제거됩니다.
  - 또한 클러스터에서 큐를 호스팅하는 큐 관리자를 변경하는 것을 어렵게 합니다.
  - 대안은 RQMNAME을 클러스터 관리자가 제공한 큐 관리자 알리어스로 바꾸는 것입니다.
  - 294 페이지의 『클러스터 대상 큐 관리자의 이름 숨기기』에서는 클러스터 외부에 있는 큐 관리자를 클러스터 내부에 있는 큐 관리자의 관리로부터 결합 해제하기 위해 큐 관리자 알리어스를 사용하는 것에 대해 설명합니다.

- 그러나 전송 큐의 이름을 지정하기 위해 제안된 방식은 전송 큐에 대상 큐 관리자의 이름을 지정하는 것입니다. 전송 큐의 이름은 클러스터 내 큐 관리자의 이름을 노출시킵니다. 따를 규칙을 선택해야 합니다. 큐 관리자 이름 또는 클러스터 이름을 사용하여 전송 큐의 이름을 지정하도록 선택할 수 있습니다.

  게이트웨이 큐 관리자 이름을 사용하여 전송 큐의 이름 지정
  : 게이트웨이 큐 관리자의 이름을 클러스터 외부에 있는 큐 관리자에게 노출하는 것은 클러스터 큐 관리자 이름을 숨기는 규칙에 대한 합리적 예외입니다.

  클러스터의 이름을 사용하여 전송 큐의 이름 지정
  : 대상 큐 관리자의 이름으로 전송 큐의 이름을 지정하는 규칙을 준수하지 않는 경우 클러스터 이름을 사용하십시오.

이 태스크 정보

291 페이지의 『클러스터에 대한 요청/응답 구성』 태스크를 수정하여 클러스터 내부에 있는 대상 큐 관리자의 이름을 숨기십시오.

프로시저

예에서 296 페이지의 그림 55의 내용을 참조하여 DEMO라고 하는 게이트웨이 큐 관리자 QM1에서 큐 관리자 알리어스를 정의하십시오.

```
DEFINE QREMOTE(DEMO) RNAME(' ') RQMNAME(' ')
```

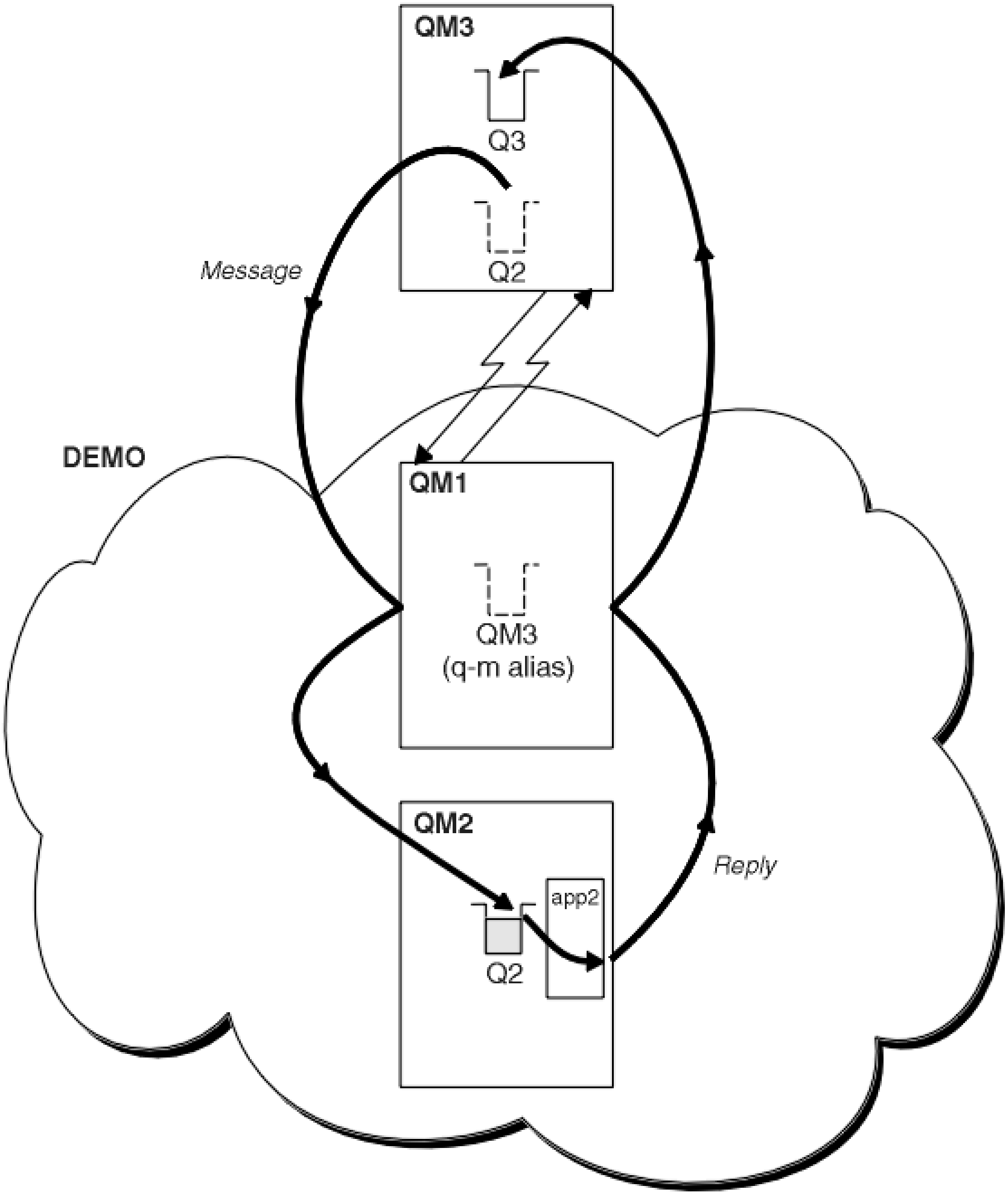


그림 55. 클러스터 외부 큐 관리자에서 넣기

QM1의 QREMOTE 정의는 큐 관리자 알리어스 DEMO를 게이트웨이 큐 관리자에게 알려지게 합니다. 클러스터 외부에 있는 큐 관리자 QM3는 실제 큐 관리자 이름을 사용할 필요 없이 큐 관리자 알리어스 DEMO를 사용하여 메시지를 DEMO의 클러스터 큐에 송신할 수 있습니다.

클러스터에 연결하는 전송 큐의 이름 지정에 클러스터 이름을 사용하는 규칙을 채택하면 Q2에 대한 리모트 큐 정의는 다음이 됩니다.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(DEMO) XMIT(DEMO)
```

결과

DEMO의 Q2를 목적지로 하는 메시지는 DEMO 전송 큐에 배치됩니다. 전송 큐에서 이러한 메시지는 송신자 채널에 의해 게이트웨이 큐 관리자 QM1으로 전송됩니다. 게이트웨이 큐 관리자는 클러스터 큐 Q2를 호스팅하는 클러스터에 있는 임의의 큐 관리자에게 메시지를 라우트합니다.

클러스터에서 요청/응답 구성

클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

시작하기 전에

298 페이지의 그림 56에서는 클러스터 DEMO 내부에 있는 큐 관리자 QM2를 보여줍니다. 이 큐 관리자는 클러스터 외부에 있는 큐 관리자에서 호스팅되는 큐 Q3에 요청을 송신합니다. 응답이 클러스터 내부에 있는 QM2의 Q2에 리턴됩니다.

클러스터 외부에 있는 큐 관리자와 통신하기 위해 클러스터 내부에서 하나 이상의 큐 관리자가 게이트웨이 역할을 합니다. 게이트웨이 큐 관리자에는 클러스터 외부에 있는 큐 관리자에 대한 통신 경로가 있습니다. 이 예에서 QM1은 게이트웨이입니다.

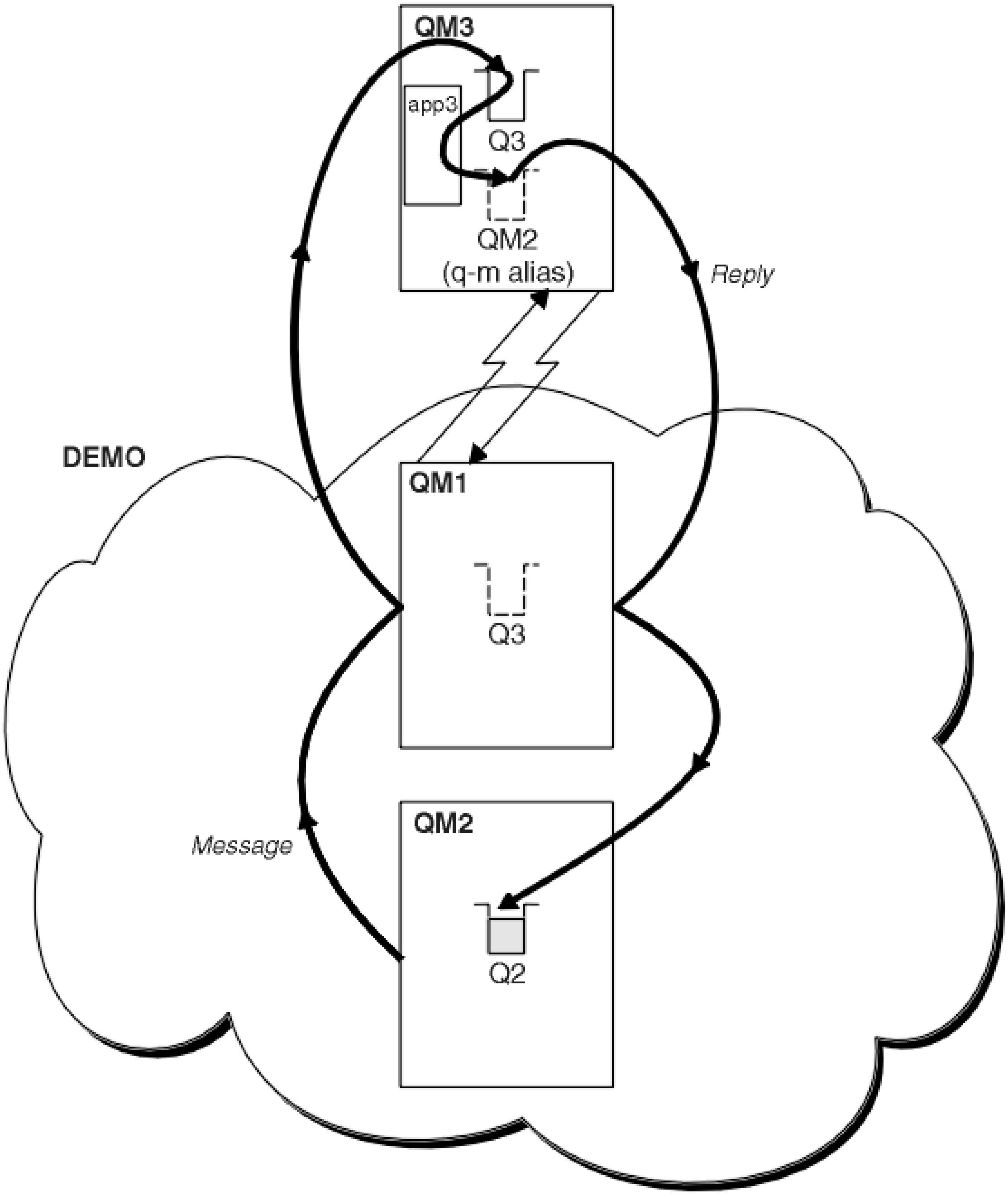


그림 56. 클러스터 외부에 있는 큐 관리자에 넣기

이 태스크 정보

지시사항에 따라 요청 및 응답 메시지에 대한 경로를 설정하십시오.

프로시저

1. 클러스터에서 요청 메시지를 송신하십시오.

   클러스터 내부에 있는 큐 관리자 QM2가 클러스터 외부에 있는 QM3의 큐 Q3에 메시지를 넣는 방법을 고려하십시오.

   a) QM1에서 리모트 큐 Q3를 클러스터에 알리는 QREMOTE 정의를 작성하십시오.

   ```
   DEFINE QREMOTE(Q3) RNAME(Q3) RQMNAME(QM3) CLUSTER(DEMO)
   ```

또한 클러스터 외부에 있는 큐 관리자에 대한 송신자 채널 및 전송 큐가 있습니다. QM3에는 해당 수신자 채널이 있습니다. 채널은 298 페이지의 그림 56에 표시되지 않습니다.

QM2의 애플리케이션이 대상 큐 및 응답이 송신될 큐를 지정하는 MQPUT 호출을 발행합니다. 대상 큐는 Q3이고 응답 대상 큐는 Q2입니다.

이 메시지는 해당 리모트 큐 정의를 사용하여 큐 이름을 QM3의 Q3로 해석하는 QM1으로 송신됩니다.

2. 클러스터 외부에 있는 큐 관리자에서 응답 메시지를 수신하십시오.

   클러스터 외부에 있는 큐 관리자에는 메시지를 송신하는 클러스터에 있는 각 큐 관리자에 대한 큐 관리자 알리어스가 있어야 합니다. 큐 관리자 알리어스는 게이트웨이 큐 관리자에 대한 전송 큐의 이름도 지정해야 합니다. 이 예에서 QM3에는 QM2에 대한 큐 관리자 알리어스 정의가 필요합니다.

   a) QM3에서 큐 관리자 알리어스 QM2를 작성하십시오.

```
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) XMITQ(QM1)
```

   QM3에는 QM1에 대한 송신자 채널 및 전송 큐도 필요하고 QM1에는 해당 수신자 채널이 필요합니다.

   그러면 QM3의 애플리케이션 **app3**는 MQPUT 호출을 발행하고 큐 이름 Q2 및 큐 관리자 이름 QM2를 지정하여 QM2에 응답을 송신할 수 있습니다.

다음에 수행할 작업

클러스터 외부에 둘 이상의 라우트를 정의할 수 있습니다.

관련 개념

큐 관리자 알리어스 및 클러스터
큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

응답 대상 큐 알리어스 및 클러스터
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

큐 알리어스 및 클러스터
큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

관련 태스크

클러스터에 대한 요청/응답 구성
클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

클러스터 외부에서 워크로드 밸런싱 구성
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

클러스터 간 메시지 경로 구성
게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

클러스터 외부에서 워크로드 밸런싱 구성

클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

시작하기 전에

291 페이지의 『클러스터에 대한 요청/응답 구성』의 292 페이지의 그림 54에 표시된 대로 예를 구성하십시오.

이 태스크 정보

이 시나리오에서 301 페이지의 그림 57의 클러스터 외부에 있는 큐 관리자 QM3는 큐 Q2에 요청을 송신합니다. Q2는 클러스터 DEMO 내 두 큐 관리자 QM2 및 QM4에서 호스팅됩니다. 두 큐 관리자 모두 워크로드 밸런싱을 사용하기 위해 NOTFIXED의 기본 바인드 옵션으로 구성됩니다. 클러스터 외부에 있는 큐 관리자인 QM3의 요청이 QM1을 통해 Q2의 인스턴스 중 하나로 송신됩니다.

QM3는 클러스터의 일부가 아니고 분산 큐잉 기술을 사용하여 통신합니다. 이 큐 관리자에는 QM1에 대한 송신자 채널 및 전송 큐가 있어야 합니다. QM1에는 해당 수신자 채널이 필요합니다. 채널 및 전송 큐는 301 페이지의 그림 57에 명시적으로 표시되지 않습니다.

이 프로시저는 291 페이지의 『클러스터에 대한 요청/응답 구성』의 292 페이지의 그림 54에 있는 예를 확장합니다.

프로시저

1. QM3에서 Q2에 대한 QREMOTE 정의를 작성하십시오.

   ```
   DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(Q3) XMITQ(QM1)
   ```

   QM3가 메시지를 넣는 클러스터의 각 큐에 대해 QREMOTE 정의를 작성하십시오.
2. QM1에서 큐 관리자 알리어스 Q3를 작성하십시오.

   ```
   DEFINE QREMOTE(Q3) RNAME(' ') RQMNAME(' ')
   ```

   Q3는 실제 큐 관리자 이름이 아닙니다. 이것은 큐 관리자 알리어스 이름 Q3와 공백 ' '가 같음을 표시하는 클러스터 내 큐 관리자 알리어스 정의의 이름입니다.
3. QM2 및 QM4 각각에서 Q2라고 하는 로컬 큐를 정의하십시오.

   ```
   DEFINE QLOCAL(Q2) CLUSTER(DEMO) DEFBIND(NOTFIXED)
   ```
4. 게이트웨이 큐 관리자인 QM1에는 특별한 정의가 없습니다.

결과

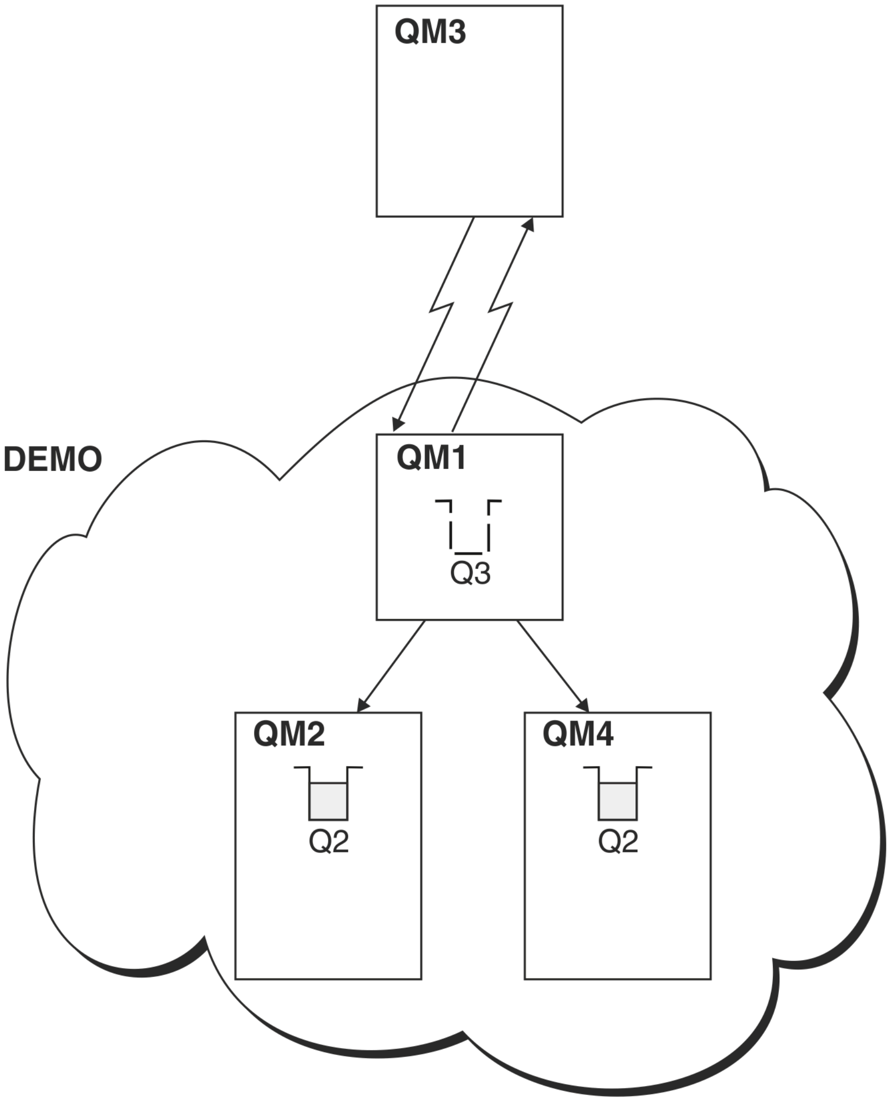


그림 57. 클러스터 외부 큐 관리자에서 넣기

QM3 애플리케이션이 메시지를 Q2에 넣기 위해 MQPUT 호출을 발행하는 경우 QM3의 QREMOTE 정의로 인해 메시지는 게이트웨이 큐 관리자 QM1을 통해 라우트됩니다. QM1이 메시지를 수신하면 메시지가 여전히 Q2라는 이름의 큐를 대상으로 하고 이름 해석을 수행함을 인식합니다. QM1은 그 로컬 정의를 검사하고 Q2를 찾지 않습니다. 그런 다음 QM1은 그 클러스터 구성을 검사하고 클러스터 DEMO에 있는 Q2의 두 인스턴스를 인식함을 알아 냅니다. QM1은 이제 워크로드 밸런싱을 이용하여 메시지를 QM2 및 QM4에 상주하는 Q2의 인스턴스에 분배할 수 있습니다.

관련 개념

큐 관리자 알리어스 및 클러스터
큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

응답 대상 큐 알리어스 및 클러스터
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

큐 알리어스 및 클러스터
큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

관련 태스크

클러스터에 대한 요청/응답 구성
클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

클러스터에서 요청/응답 구성
클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

클러스터 간 메시지 경로 구성
게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

관련 정보

큐 이름 해석

이름 해석

클러스터 간 메시지 경로 구성

게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

이 태스크 정보

하나의 대형 클러스터에 모든 큐 관리자를 그룹화하는 대신 여러 작은 클러스터로 구성할 수 있습니다. 각 클러스터에는 브릿지 역할을 함에 있어 하나 이상의 큐 관리자가 있습니다. 이를 사용하면 클러스터 전체에서 큐 및 큐 관리자 이름 표시를 제한할 수 있습니다. 중첩 클러스터를 참조하십시오. 이름 충돌을 피하거나 로컬 이름 지정 규칙을 준수하려면 알리어스를 사용하여 큐 및 큐의 이름을 변경하십시오.

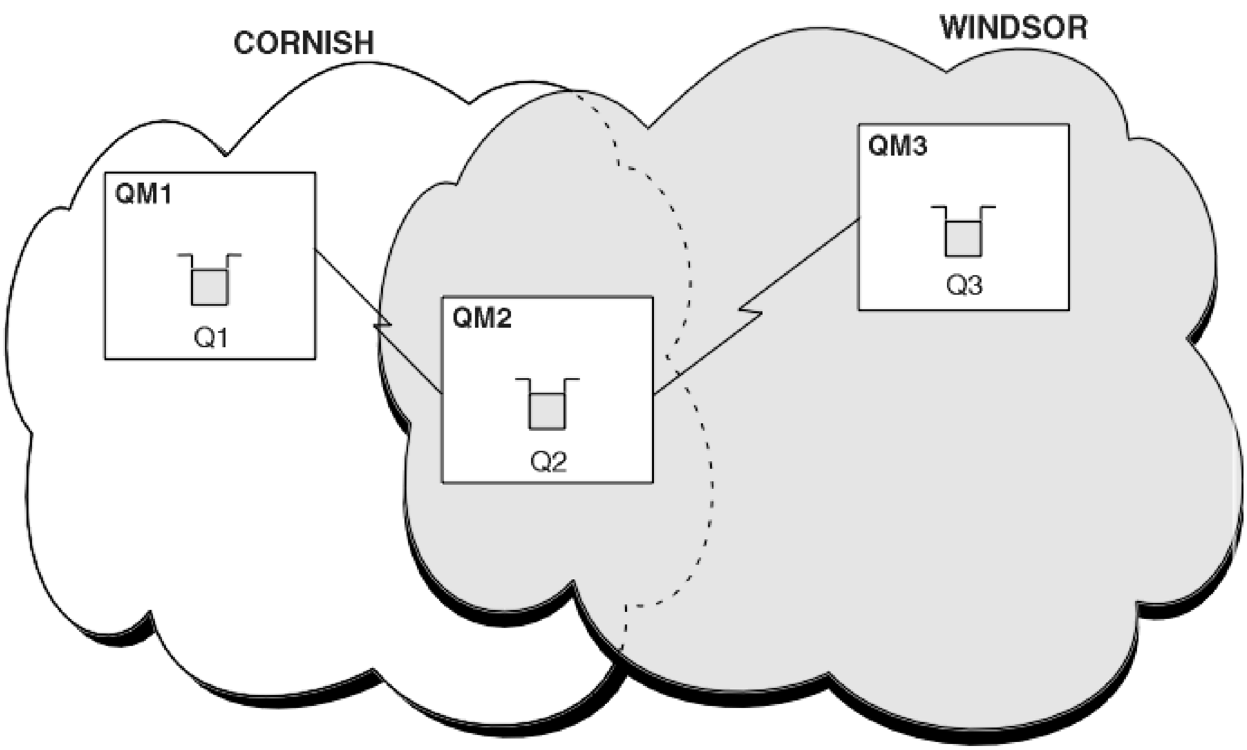


그림 *58.* 클러스터 전체에서 브릿징

302 페이지의 그림 58에서는 사이에 브릿지가 있는 두 개의 클러스터를 보여줍니다. 둘 이상의 브릿지가 있을 수 있습니다.

다음 프로시저를 사용하여 클러스터를 구성하십시오.

프로시저

1. QM1에서 클러스터 큐 Q1을 정의하십시오.

```
DEFINE QLOCAL(Q1) CLUSTER(CORNISH)
```

2. QM3에서 클러스터 큐 Q3를 정의하십시오.

```
DEFINE QLOCAL(Q3) CLUSTER(WINDSOR)
```

3. QM2에서 두 클러스터의 이름을 모두 포함하는 CORNISHWINDSOR이라고 하는 이름 목록을 작성하십시오.

```
DEFINE NAMELIST(CORNISHWINDSOR) DESCR('CornishWindsor namelist')
NAMES(CORNISH, WINDSOR)
```

4. QM2에서 클러스터 큐 Q2를 정의하십시오.

```
DEFINE QLOCAL(Q2) CLUSNL(CORNISHWINDSOR)
```

다음에 수행할 작업

QM2는 두 클러스터의 멤버이고 두 클러스터 간 브릿지입니다. 브릿지 전체에 표시되게 하려는 각 큐의 경우 브릿지에 대한 QALIAS 정의가 필요합니다. 예를 들어, 302 페이지의 그림 58에서 QM2에는 다음이 필요합니다.

```
DEFINE QALIAS(MYQ3) TARGQ(Q3) CLUSTER(CORNISH) DEFBIND(NOTFIXED)
```

큐 알리어스를 사용하여 CORNISH에서 큐 관리자에 연결된 애플리케이션(예: QM1)은 메시지를 Q3에 넣을 수 있습니다. 이 애플리케이션은 Q3를 MYQ3로 참조합니다. 이 메시지는 QM3의 Q3로 라우트됩니다.

큐를 열 때 DEFBIND를 NOTFIXED 또는 QDEF로 설정해야 합니다. DEFBIND를 기본값 OPEN으로 두면 큐 관리자는 호스팅하는 브릿지 큐 관리자로 알리어스 정의를 해석합니다. 이 브릿지는 메시지를 전달하지 않습니다.

표시할 각 큐 관리자에 대해 큐 관리자 알리어스 정의가 필요합니다. 예를 들어, QM2에서 다음이 필요합니다.

```
DEFINE QREMOTE(QM1) RNAME(' ') RQMNAME(QM1) CLUSTER(WINDSOR)
```

WINDSOR에서 큐 관리자(예: QM3)에 연결되는 애플리케이션은 MQOPEN 호출에서 QM1을 명시적으로 이름 지정하여 QM1에 있는 임의의 큐에 메시지를 넣을 수 있습니다.

관련 개념

큐 관리자 알리어스 및 클러스터
큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

응답 대상 큐 알리어스 및 클러스터
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

큐 알리어스 및 클러스터
큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

관련 태스크

클러스터에 대한 요청/응답 구성
클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

클러스터에서 요청/응답 구성

클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

클러스터 외부에서 워크로드 밸런싱 구성
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

큐 관리자 알리어스 및 클러스터

큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

공백 RNAME이 포함된 리모트 큐 정의를 사용하여 작성되는 큐 관리자 알리어스에는 다음과 같은 다섯 가지 용도가 있습니다.

메시지를 송신할 때 큐 관리자 이름 재맵핑

큐 관리자 알리어스는 MQOPEN 호출에 지정된 큐 관리자 이름을 다른 큐 관리자로 재맵핑하는 데 사용할 수 있습니다. 이는 클러스터 큐 관리자일 수 있습니다. 예를 들어, 큐 관리자에는 큐 관리자 알리어스 정의가 있을 수 있습니다.

```
DEFINE QREMOTE(YORK) RNAME(' ') RQMNAME(CLUSQM)
```

YORK는 CLUSQM이라고 하는 큐 관리자의 알리어스로 사용될 수 있습니다. 이 정의를 작성한 큐 관리자의 애플리케이션이 메시지를 큐 관리자 YORK에 넣으면 로컬 큐 관리자가 이 이름을 CLUSQM로 해석합니다. 로컬 큐 관리자를 CLUSQM이라고 하지 않을 경우 메시지를 CLUSQM으로 이동할 클러스터 전송 큐에 넣습니다. 또한 전송 헤더를 YORK가 아닌 CLUSQM을 표시하도록 변경합니다.

참고: 이 정의는 이 정의를 작성하는 큐 관리자에서만 적용됩니다. 알리어스를 전체 클러스터에 알리려면 CLUSTER 속성을 리모트 큐 정의에 추가해야 합니다. 그러면 YORK를 대상으로 했던 다른 큐 관리자의 메시지가 CLUSQM으로 송신됩니다.

메시지 송신 시 전송 큐 대체 또는 지정

알리어스는 클러스터를 비클러스터 시스템에 조인하는 데 사용할 수 있습니다. 예를 들어, 클러스터 ITALY의 큐 관리자는 클러스터 외부에 있는 PALERMO라고 하는 큐 관리자와 통신할 수 있습니다. 통신하려면 클러스터 내 큐 관리자 중 하나가 게이트웨이 역할을 해야 합니다. 게이트웨이 큐 관리자에서 다음 명령을 실행하십시오.

```
DEFINE QREMOTE(ROME) RNAME(' ') RQMNAME(PALERMO) XMITQ(X) CLUSTER(ITALY)
```

이 명령은 큐 관리자 알리어스 정의입니다. 이 명령은 ROME을 큐 관리자로 정의하고 알리며 이 큐 관리자를 통해 클러스터 ITALY 내 모든 큐 관리자의 메시지가 PALERMO에 있는 해당 대상에 도달하기 위해 멀티호핑할 수 있습니다. ROME으로 설정된 큐 관리자 이름으로 열리는 큐에 넣은 메시지는 큐 관리자 알리어스 정의가 있는 게이트웨이 큐 관리자로 송신됩니다. 여기에서 큐 관리자 메시지는 전송 큐 X에 들어가고 비클러스터 채널에 의해 큐 관리자 PALERMO로 이동합니다.

이 예에서 이름 ROME을 선택하는 것은 중요하지 않습니다. QREMOTE 및 RQMNAME에 대한 값은 둘 다 같을 수 있습니다.

메시지를 수신할 때 대상 판별

큐 관리자는 메시지를 수신하면 전송 헤더에서 목적지 큐와 큐 관리자의 이름을 추출합니다. 큐 관리자는 전송 헤더에서 큐 관리자와 같은 이름을 가지는 큐 관리자 알리어스 정의를 찾습니다. 이러한 정의를 찾으면 전송 헤더의 큐 관리자 이름에 대한 큐 관리자 알리어스 정의에서 RQMNAME을 대체합니다.

이 방식으로 큐 관리자 알리어스를 사용하는 두 가지 이유가 있습니다.

- 메시지를 다른 큐 관리자로 경로 지정
- 큐 관리자 이름을 로컬 큐 관리자와 같도록 변경

게이트웨이 큐 관리자에서 큐 관리자 알리어스를 사용하여 서로 다른 클러스터에 있는 큐 관리자 사이에 메시지를 라우트합니다.

애플리케이션은 큐 관리자 알리어스를 사용하여 다른 클러스터에 있는 큐에 메시지를 송신할 수 있습니다. 큐는 꼭 클러스터 큐가 아니어도 됩니다. 큐는 한 개의 클러스터에서 정의됩니다. 애플리케이션이 다른 클러스터에 있는 큐 관리자에 연결됩니다. 게이트웨이 큐 관리자가 두 클러스터를 연결합니다. 큐가 클러스터된 것으로 정의되지 않은 경우 올바른 라우팅이 발생하도록 애플리케이션은 큐 이름 및 클러스터된 큐 관리자 알리어스 이름을 사용하여 큐를 열어야 합니다. 구성 예는 261 페이지의 『게이트웨이 큐 관리자로 두 개의 중첩 클러스터 작성』의 내용을 참조하십시오. 여기에서 그림 1에 설명된 응답 메시지 플로우가 선택됩니다.

이 다이어그램은 RQ라고 하는 응답 메시지가 다시 임시 동적 큐로 돌아가기 위해 따라가는 경로를 표시합니다. QM3에 연결되는 서버 애플리케이션은 큐 관리자 이름 QM2를 사용하여 응답 큐를 엽니다. 큐 관리자 이름 QM2는 QM1에서 클러스터된 큐 관리자 알리어스로 정의됩니다. QM3는 응답 메시지를 QM1으로 라우트합니다. QM1은 그 메시지를 QM2로 라우트합니다.

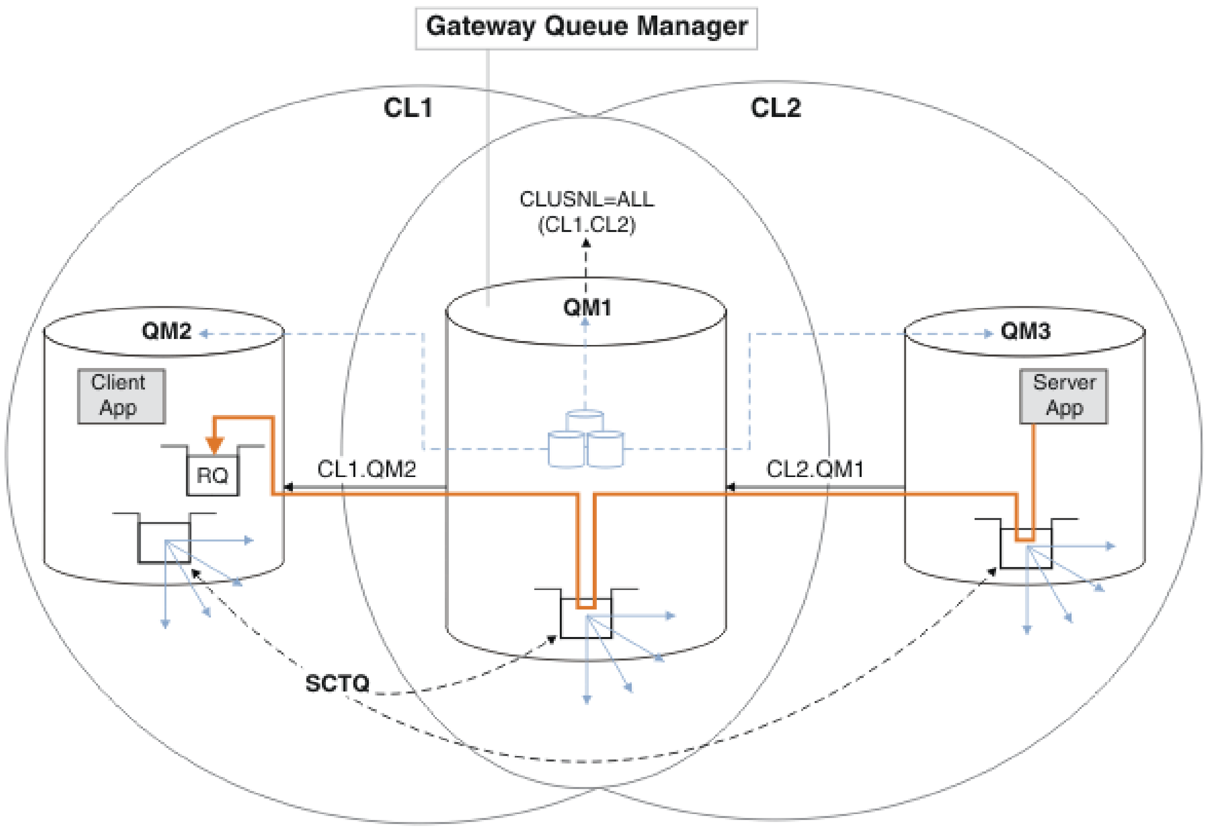


그림 59. 큐 관리자 알리어스를 사용하여 응답 메시지를 다른 클러스터로 리턴

라우팅이 작동하는 방식은 다음과 같습니다. 각 클러스터의 모든 큐 관리자에게는 QM1에 큐 관리자 알리어스 정의가 있습니다. 이러한 알리어스는 모든 클러스터에서 클러스터됩니다. 각 알리어스에서 큐 관리자로의 회색 파선 화살표는 각 큐 관리자 알리어스가 최소 하나 이상의 클러스터에 있는 실제 큐 관리자로 해석됨을 표시합니다. 이 경우 QM2 알리어스는 두 클러스터 CL1 및 CL2 모두에서 클러스터되어 CL1에 있는 실제 큐 관리자 QM2로 해석됩니다. 서버 애플리케이션은 큐 이름 RQ에 대한 응답 및 큐 관리자 이름 QM2에 대한 응답을 사용하여 응답 메시지를 작성합니다. 이 메시지는 QM1으로 라우트됩니다. 큐 관리자 알리어스 정의 QM2가 CL2에 있는 QM1에서 정의되고 큐 관리자 QM2는 클러스터 CL2에 없기 때문입니다. 메시지를 대상 큐 관리자로 송신할 수 없으므로 이 메시지는 알리어스 정의가 있는 큐 관리자에게 송신됩니다.

QM1은 QM2로 전송할 메시지를 QM1의 클러스터 전송 큐에 배치합니다. QM1은 메시지를 QM2로 라우트합니다. QM2에 대한 QM1의 큐 관리자 알리어스 정의는 QM2를 실제 대상 큐 관리자로 정의하기 때문입니다. 정의는 순환되지 않습니다. 알리어스 정의는 실제 정의만 참조할 수 있고 알리어스는 그 자체를 가리킬 수 없기 때문입니다. 실제 정의는 QM1에 의해 해석됩니다. QM1 및 QM2 둘 다 같은 클러스터 CL1에 있기 때문입니다. QM1은 CL1의 저장소에서 QM2에 대한 연결 정보를 찾아내고 메시지를 QM2에 라우트합니다. 메시지가 QM1에 의해 재라우트되도록 하려면 서버 애플리케이션은 DEFBIND 옵션 MQBND_BIND_NOT_FIXED로 설정하

여 응답 큐를 열어야 합니다. 서버 애플리케이션이 MQBND_BIND_ON_OPEN 옵션을 사용하여 응답 큐를 연 경우 메시지는 재라우트되지 않고 데드-레터 큐에 있게 됩니다.

큐 관리자를 클러스터로의 게이트웨이로 사용하여 클러스터 외부에서 수신되는 메시지를 워크로드 밸런싱합니다.

클러스터에 있는 둘 이상의 큐 관리자에서 EDINBURGH라고 하는 큐를 정의합니다. 클러스터링 메커니즘이 클러스터 외부에서 수신되는 메시지에 대해 워크로드 밸런싱을 수행하게 하려고 합니다.

클러스터 외부의 큐 관리자에게는 클러스터 내 한 큐 관리자에 대한 송신자 채널 및 전송 큐가 필요합니다. 이 큐를 게이트웨이 큐 관리자라고 합니다. 기본 워크로드 밸런싱 메커니즘을 이용하려면 다음 규칙 중 하나를 적용해야 합니다.

- 게이트웨이 큐 관리자는 EDINBURGH 큐의 인스턴스를 포함해서는 안됩니다.
- 게이트웨이 큐 관리자는 ALTER QMGR에서 CLWLUSEQ(ANY)를 지정합니다.

클러스터 외부에서의 워크로드 밸런싱 예는 299 페이지의 『클러스터 외부에서 워크로드 밸런싱 구성』의 내용을 참조하십시오.

관련 개념

응답 대상 큐 알리어스 및 클러스터
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

큐 알리어스 및 클러스터
큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

관련 태스크

클러스터에 대한 요청/응답 구성
클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

클러스터에서 요청/응답 구성
클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

클러스터 외부에서 워크로드 밸런싱 구성
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

클러스터 간 메시지 경로 구성
게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

응답 대상 큐 알리어스 및 클러스터

응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

예를 들면 다음과 같습니다.

- 큐 관리자 VENICE의 애플리케이션은 MQPUT 호출을 사용하여 큐 관리자 PISA에 메시지를 송신합니다. 이 애플리케이션은 메시지 디스크립터에서 다음 응답 대상 큐 정보를 제공합니다.

```
ReplyToQ='QUEUE'
ReplyToQMgr=''
```

- QUEUE에 송신된 응답을 PISA의 OTHERQ에서 수신할 수 있도록 VENICE에서 응답 대상 큐 알리어스로 사용되는 리모트 큐 정의를 작성하십시오. 이 알리어스는 알리어스가 작성된 시스템에서만 유효합니다.

```
DEFINE QREMOTE(QUEUE) RNAME(OTHERQ) RQMNAME(PISA)
```

RQMNAME 및 QREMOTE는 RQMNAME이 그 자체로 클러스터 큐 관리자인 경우라도 같은 이름을 지정할 수 있습니다.

관련 개념

큐 관리자 알리어스 및 클러스터
큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

큐 알리어스 및 클러스터
큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

관련 태스크

클러스터에 대한 요청/응답 구성
클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

클러스터에서 요청/응답 구성
클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

클러스터 외부에서 워크로드 밸런싱 구성
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

클러스터 간 메시지 경로 구성
게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

큐 알리어스 및 클러스터

큐 알리어스를 사용하여 클러스터 큐의 이름을 숨기거나, 큐를 클러스터링하거나, 다른 속성을 채택하거나 다른 액세스 제어를 채택할 수 있습니다.

QALIAS 정의는 큐를 나타내는 알리어스를 작성하는 데 사용됩니다. 다음과 같은 여러 이유로 알리어스를 작성할 수 있습니다.

- 다른 큐 사용을 시작하려고 하지만 애플리케이션은 변경하지 않으려고 합니다.
- 애플리케이션이 메시지를 넣는 큐의 실제 이름을 애플리케이션에 알리지 않으려고 합니다.
- 큐가 정의된 것과는 다른 이름 지정 규칙이 있을 수 있습니다.
- 애플리케이션이 실제 이름을 사용할 경우 큐에 대한 액세스 권한이 부여되지 않고 그 알리어스를 사용하는 경우에만 액세스 권한이 부여됩니다.

DEFINE QALIAS 명령을 사용하여 큐 관리자에서 QALIAS 정의를 작성하십시오. 예를 들어, 다음 명령을 실행하십시오.

```
DEFINE QALIAS(PUBLIC) TARGQ(LOCAL) CLUSTER(C)
```

명령은 PUBLIC이라는 큐를 클러스터 C의 큐 관리자에 알립니다. PUBLIC은 LOCAL이라는 큐로 해석되는 별명입니다. PUBLIC으로 송신되는 메시지는 LOCAL이라고 하는 큐에 라우트됩니다.

큐 알리어스 정의를 사용하여 큐 이름을 클러스터 큐로 해석할 수도 있습니다. 예를 들어, 다음 명령을 실행하십시오.

```
DEFINE QALIAS(PRIVATE) TARGQ(PUBLIC)
```

이 명령을 사용하면 큐 관리자가 이름 PRIVATE을 사용하여 PUBLIC이라는 이름으로 클러스터 다른 위치에 알려진 큐에 액세스할 수 있습니다. 이 정의에는 CLUSTER 속성이 포함되지 않으므로 이를 작성하는 큐 관리자에만 적용됩니다.

관련 개념

큐 관리자 알리어스 및 클러스터

큐 관리자 알리어스를 사용하여 메시지를 클러스터 내외로 송신할 때 큐 관리자의 이름을 숨기고 클러스터에 송신된 메시지를 워크로드 밸런싱합니다.

응답 대상 큐 알리어스 및 클러스터
응답 대상 큐 알리어스 정의는 응답 정보에 대한 대체 이름을 지정하는 데 사용됩니다. 응답 대상 큐 알리어스 정의는 분산 큐잉 환경에서와 같이 클러스터에서 사용할 수 있습니다.

관련 태스크

클러스터에 대한 요청/응답 구성
클러스터 외부에 있는 큐 관리자로부터 요청/응답 메시지를 구성합니다. 게이트웨이 큐 관리자를 클러스터에 대한 통신 경로로 사용하여 클러스터의 내부 세부사항을 숨깁니다.

클러스터에서 요청/응답 구성
클러스터에서 클러스터 외부에 있는 큐 관리자로 요청/응답 메시지 경로를 구성합니다. 게이트웨이 큐 관리자를 사용하여 클러스터 내부 큐 관리자가 클러스터 외부와 통신하는 방법에 대한 세부사항을 숨깁니다.

클러스터 외부에서 워크로드 밸런싱 구성
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

클러스터 간 메시지 경로 구성
게이트웨이 큐 관리자를 사용하여 클러스터를 서로 연결합니다. 게이트웨이 큐 관리자에서 클러스터 큐 또는 클러스터 큐 관리자 알리어스를 정의하여 큐 또는 큐 관리자가 모든 클러스터에 표시되도록 합니다.

워크로드 관리에 클러스터 사용

클러스터에 있는 여러 큐 관리자에서 한 큐의 다중 인스턴스를 정의하여 다중 서버에 큐 서비스 작업을 분산시킬 수 있습니다. 실패가 발생할 경우 메시지가 다른 큐 관리자에 다시 큐잉되지 않게 방지할 수 있는 여러 요소가 있습니다.

시스템 관리를 줄이게 위해 클러스터를 설정할 뿐 아니라 둘 이상의 큐 관리자가 같은 큐의 인스턴스를 호스팅하는 클러스터를 작성할 수 있습니다.

클러스터 안에 있는 큐 관리자가 서로의 복제본이 되도록 클러스터를 구성할 수 있습니다. 각 큐 관리자는 동일한 애플리케이션을 실행하고 동일한 큐의 로컬 정의를 가질 수 있습니다. z/OS 예를 들어, z/OS 병렬 sysplex에서 복제된 애플리케이션은 공유 Db2 또는 VSAM(Virtual Storage Access Method) 데이터베이스의 데이터에 액세스할 수 있습니다. 애플리케이션의 여러 인스턴스를 가짐으로써 워크로드를 큐 관리자에 사이에 분산시킬 수 있습니다. 애플리케이션의 각 인스턴스는 다른 인스턴스와는 독립적으로 메시지를 수신하고 실행합니다.

이 방식으로 클러스터를 사용하는 장점은 다음과 같습니다.

- 큐 및 애플리케이션의 증가된 가용성
- 더 빨라진 메시지 처리량
- 네트워크에서 워크로드의 더 고른 분배

특정 큐의 인스턴스를 호스팅하는 큐 관리자 중 어느 하나는 해당 큐를 목적지로 하는 메시지를 핸들링할 수 있고 애플리케이션은 메시지를 송신할 때 큐 관리자의 이름을 지정하지 않습니다. 클러스터에 같은 큐에 대한 둘 이상의 인스턴스가 포함된 경우 IBM MQ는 메시지를 라우트할 큐 관리자를 선택합니다. 적당한 목적지는 큐 관리자 및 큐의 가용성과 큐 관리자, 큐 및 채널과 연관된 다수의 클러스터 워크로드별 속성을 기준으로 선택됩니다. 클러스터에서 워크로드 밸런싱을 참조하십시오.

z/OS IBM MQ for z/OS에서 큐 공유 그룹에 있는 큐 관리자는 클러스터 큐를 공유 큐로 호스팅할 수 있습니다. 공유 클러스터 큐는 같은 큐 공유 그룹에 있는 모든 큐 관리자가 사용할 수 있습니다. 예를 들어, 같은 큐의 다중 인스턴스를 포함하는 클러스터에서 큐 관리자 QM2 및 QM4 중 하나 또는 둘 다가 공유 큐 관리자일 수 있습니다. 각각에는 큐 Q3에 대한 정의가 있습니다. QM4와 같은 큐 공유 그룹에 있는 큐 관리자는 공유 큐 Q3에 넣은 메시지를 읽을 수 있습니다. 각 큐 공유 그룹은 각각 같은 데이터에 대한 액세스를 가지는 최대 32개의 큐 관리자를 포함할 수 있습니다. 큐 공유는 메시지 처리량을 상당히 증가시킵니다.

워크로드 관리를 위한 클러스터 구성의 자세한 정보는 다음 하위 주제를 참조하십시오.

관련 개념

290 페이지의 『메시지를 클러스터로(부터) 라우팅』

큐 알리어스, 큐 관리자 알리어스 및 리모트 큐 정의를 사용하여 클러스터를 외부 큐 관리자 및 다른 클러스터에 연결합니다.

관련 태스크

216 페이지의 『큐 관리자 클러스터 구성』
클러스터는 초기 구성 및 진행 중인 관리를 모두 단순화하는 방식으로 큐 관리자를 상호 연결하기 위한 메커니즘을 제공합니다. 클러스터 컴포넌트를 정의하고 클러스터를 작성 및 관리할 수 있습니다.

226 페이지의 『새 클러스터 설정』
다음 지시사항에 따라 클러스터 예를 설정합니다. 개별 지시사항에서는 TCP/IP, LU 6.2 및 단일 전송 큐 또는 다중 전송 큐에서의 클러스터 설정을 설명합니다. 한 큐 관리자에서 다른 큐 관리자로 메시지를 송신하여 클러스터가 작동하는지 테스트합니다.

299 페이지의 『클러스터 외부에서 워크로드 밸런싱 구성』
클러스터 외부에 있는 큐 관리자에서 클러스터 큐 사본에 대한 메시지 경로를 구성합니다. 결과는 클러스터 외부 요청을 클러스터 큐의 각 인스턴스로 워크로드 밸런싱하는 것입니다.

관련 정보

클러스터링과 분산 큐잉의 비교
분산 큐잉 및 클러스터
클러스터의 컴포넌트
클러스터 채널
클러스터 큐가 MQPUT에 대해 사용 안함으로 설정되는 경우 발생하는 상황
클러스터-송신자 채널에서 설정된 워크로드 밸런싱이 작동하지 않음
클러스터 큐 모니터링 샘플 프로그램(AMQSCLM)
클러스터 워크로드 엑시트 작성 및 컴파일

둘 이상의 큐 인스턴스가 있는 클러스터의 예

둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

310 페이지의 그림 60에서는 큐 Q3에 대해 둘 이상의 정의가 있는 클러스터를 보여줍니다. QM1의 애플리케이션이 메시지를 Q3에 넣는 경우 해당 메시지를 처리할 Q3의 인스턴스를 꼭 알아야 하는 것은 아닙니다. 애플리케이션이 Q3의 로컬 인스턴스가 있는 QM2 또는 QM4에서 실행되는 경우 기본적으로 Q3의 로컬 인스턴스가 열립니다. CLWLUSEQ 큐 속성을 설정하면 큐의 로컬 인스턴스를 큐의 원격 인스턴스와 동일하게 처리할 수 있습니다.

MQOPEN 옵션 DefBind는 MQOPEN 호출이 발행될 때 또는 메시지가 전송 큐에서 전송될 때 대상 큐 관리자가 선택되는지 여부를 제어합니다.

DefBind를 MQBND_BIND_NOT_FIXED로 설정하면 메시지가 전송될 때 사용 가능한 큐의 인스턴스에 메시지가 송신될 수 있습니다. 이 경우 다음과 같은 문제점을 피할 수 있습니다.

- 메시지가 대상 큐 관리자에 도착할 때 대상 큐가 사용 불가능합니다.
- 큐의 상태가 변경되었습니다.
- 메시지를 클러스터 큐 알리어스를 사용하여 넣었고 클러스터 큐 알리어스의 인스턴스가 정의되는 큐 관리자에 대상 큐의 인스턴스가 존재하지 않습니다.

전송 시 이러한 문제점 중 어느 하나라도 발견되면 대상 큐의 사용 가능한 다른 인스턴스를 찾아 메시지를 다시 라우팅합니다. 사용 가능한 큐의 인스턴스가 없는 경우 메시지가 데드-레터 큐에 배치됩니다.

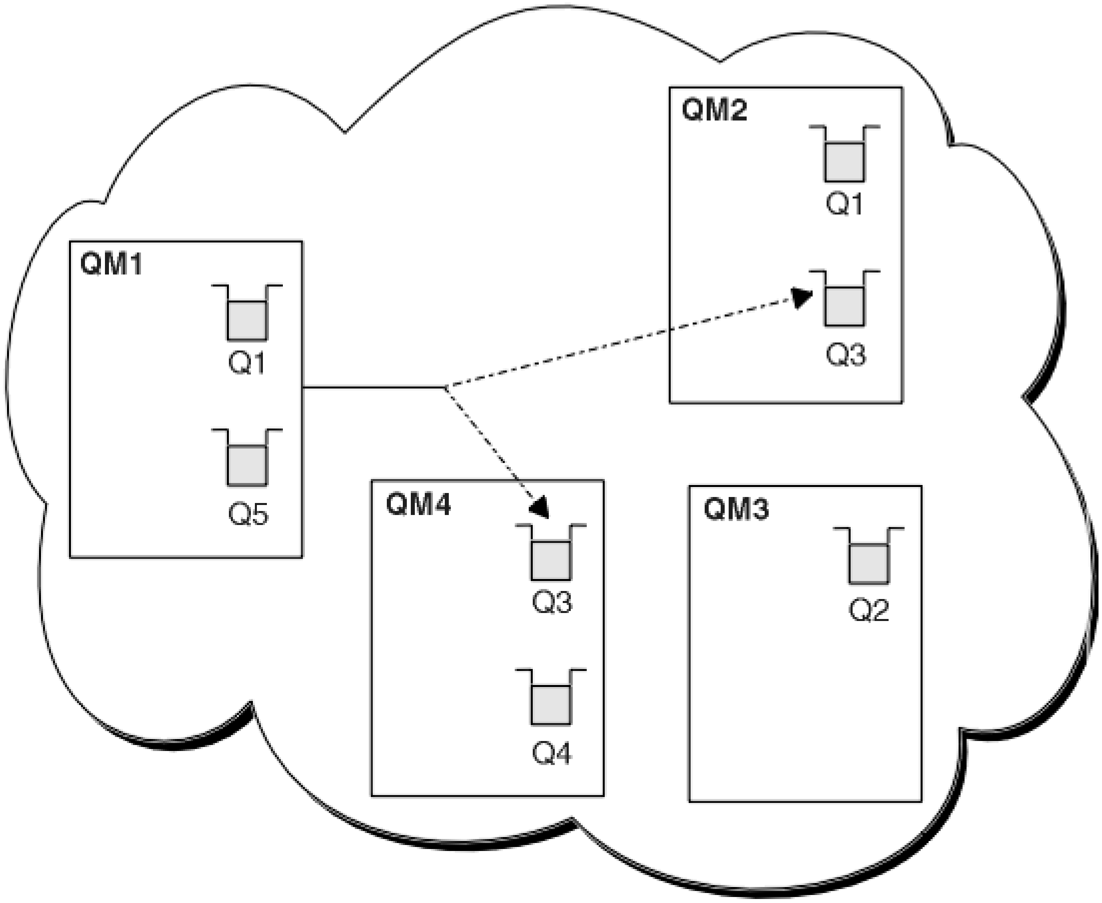


그림 *60.* 동일한 큐의 다중 인스턴스를 포함하는 클러스터

메시지가 다시 라우팅되는 것을 막을 수 있는 한 가지 요소는 MQBND_BIND_ON_OPEN을 사용하여 메시지를 고정 큐 관리자 또는 채널로 지정하는 경우입니다. MQOPEN에 바인드된 메시지는 다른 채널에 재할당되지 않습니다. 또한 메시지 재할당은 클러스터 채널이 실제로 실패할 때만 발생한다는 점도 유의하십시오. 재할당은 채널이 이미 실패한 경우에는 발생하지 않습니다.

목적지 큐 관리자가 작동하지 않으면 시스템이 메시지를 다시 라우팅하려고 시도합니다. 이러한 시도에서 시스템은 메시지를 손실할 수 있는 위험을 감수하거나 복제본을 작성하여 메시지의 무결성에 영향을 미치지 않습니다. 큐 관리자가 실패하고 인다우트(in-doubt) 상태로 메시지를 둘 경우 해당 메시지는 다시 라우팅되지 않습니다.

z/OS IBM MQ for z/OS에서 채널은 메시지 재할당 프로세스가 완료될 때까지 완전히 중지되지 않습니다. 모드를 FORCE 또는 TERMINATE로 설정한 상태로 채널을 중지하면 프로세스가 인터럽트되므로, 이를 수행하는 경우 일부 BIND_NOT_FIXED 메시지는 이미 다른 채널로 재할당되거나 메시지가 순서대로 정렬되지 않았을 수 있습니다.

참고: z/OS

1. 동일한 큐의 다중 인스턴스가 있는 클러스터를 설정하기 전에 메시지에 서로에 대한 종속 항목이 없는지 확인하십시오. 예를 들어, 특정 시퀀스로 또는 같은 큐 관리자에 의해 처리되어야 하는 경우입니다.
2. 같은 큐의 다른 인스턴스에 대한 정의를 동일하게 작성하십시오. 그렇지 않으면 다른 MQINQ 호출에서 다른 결과를 얻게 됩니다.

관련 개념

애플리케이션 프로그래밍 및 클러스터
동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 두 개의 네트워크 사용
다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

큐를 로컬로 호스팅하는 큐 관리자 추가

다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 클러스터에 새 큐 관리자 추가에 설명된 대로 설정되었습니다. 이 클러스터는 세 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS는 부분 저장소를 보유합니다. 인벤토리 애플리케이션은 NEWYORK 큐 관리자에 연결된 New York의 시스템에서 실행됩니다. 애플리케이션은 INVENTQ 큐에 메시지가 도착하면 구동됩니다.
- INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공하려고 합니다.

이 태스크 정보

다음 단계에 따라 큐를 로컬로 호스팅하는 큐 관리자를 추가하십시오.

프로시저

1. PARIS 큐 관리자를 대체하십시오.

   Paris에 있는 애플리케이션이 Paris에 있는 INVENTQ와 New York에 있는 INVENTQ를 사용하게 하려면 큐 관리자에게 알려야 합니다. PARIS에서 다음 명령을 실행하십시오.

   ```
   ALTER QMGR CLWLUSEQ(ANY)
   ```
2. 메시지 연관관계에 대해 인벤토리 애플리케이션을 검토하십시오.

   계속하기 전에 메시지 처리 순서에 따라 인벤토리 애플리케이션에 종속 항목이 없는지 확인하십시오. 자세한 정보는 메시지 연관관계 핸들링을 참조하십시오.
3. Paris에 있는 시스템에 인벤토리 애플리케이션을 설치하십시오.
4. 클러스터 큐 INVENTQ를 정의하십시오.

이미 NEWYORK 큐 관리자가 호스팅하고 있는 INVENTQ 큐도 PARIS가 호스팅합니다. 다음과 같이 이 큐를 PARIS 큐 관리자에서 정의하십시오.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

모든 정의를 마쳤으므로 아직 수행하지 않은 경우 IBM MQ for z/OS에서 채널 시작기를 시작하십시오. 모든 플랫폼의 큐 관리자 PARIS에서 리스너 프로그램을 시작하십시오. 리스너는 수신되는 네트워크 요청을 청취하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

결과

312 페이지의 그림 61에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

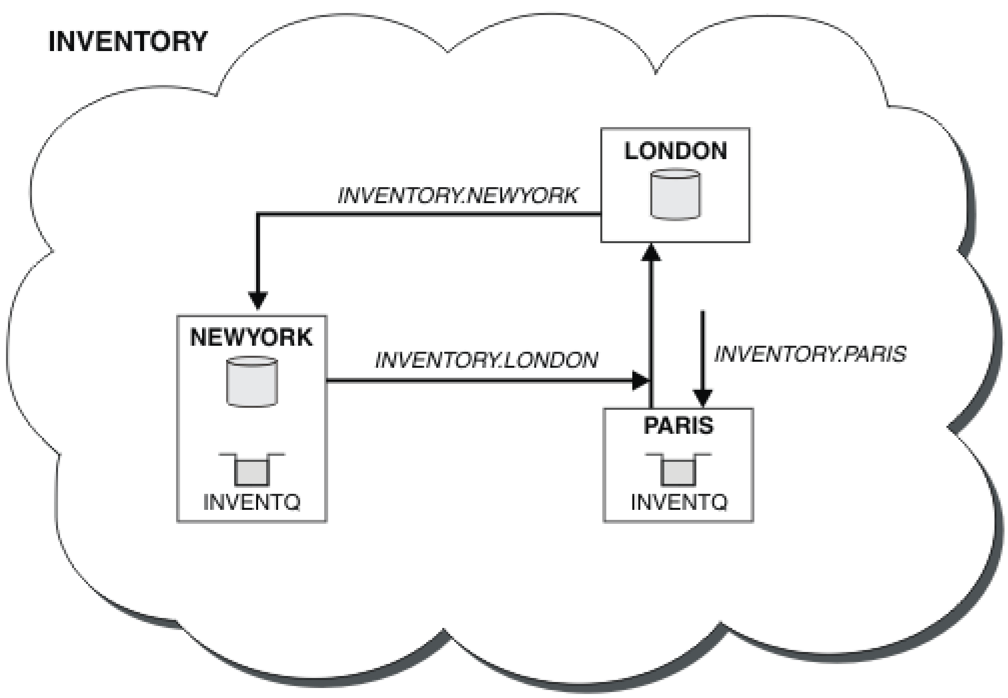


그림 *61.* 세 개의 큐 관리자가 있는 *INVENTORY* 클러스터

이 클러스터에 대한 수정은 큐 관리자 NEWYORK 또는 LONDON을 변경하지 않고 수행되었습니다. 이러한 큐 관리자의 전체 저장소는 메시지를 PARIS의 INVENTQ에 송신하기 위해 필요한 정보로 자동 업데이트됩니다.

다음에 수행할 작업

INVENTQ 큐 및 인벤토리 애플리케이션은 이제 클러스터에 있는 두 개의 큐 관리자에서 호스팅됩니다. 이 경우 두 큐 관리자의 가용성이 증가되고 메시지 처리 속도가 빨라지며 두 큐 관리자 사이에 워크로드를 분산시킬 수 있습니다. 큐 관리자 LONDON, NEWYORK 및 PARIS 중 어느 하나가 INVENTQ에 넣은 메시지는 워크로드 밸런싱을 위해 PARIS 또는 NEWYORK에 교대로 라우트됩니다.

관련 개념

둘 이상의 큐 인스턴스가 있는 클러스터의 예
둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

애플리케이션 프로그래밍 및 클러스터
동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

클러스터에서 두 개의 네트워크 사용

다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

클러스터에서 두 개의 네트워크 사용

다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 "클러스터에 큐 관리자 추가"에 설명된 대로 설정되었습니다. 이 클러스터는 세 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS는 부분 저장소를 보유합니다. 인벤토리 애플리케이션은 NEWYORK 큐 관리자에 연결된 New York의 시스템에서 실행됩니다. 애플리케이션은 INVENTQ 큐에 메시지가 도착하면 구동됩니다.
- 새 저장소가 두 개의 다른 네트워크가 있는 TOKYO에 추가됩니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

이 태스크 정보

다음 단계에 따라 클러스터에서 두 네트워크를 사용하십시오.

프로시저

1. TOKYO가 처음 참조할 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 클러스터에 대한 정보를 수집하기 위해 전체 저장소 중 어느 하나를 참조해야 합니다. 그러면 자체 부분 저장소를 빌드합니다. 저장소 선택이 특별히 중요한 것은 아닙니다. 이 예에서는 NEWYORK을 선택합니다. 새 큐 관리자는 클러스터에 조인하고 나면 두 저장소와 모두 통신합니다.
2. CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자를 정의해야 합니다. 이 큐 관리자는 각 네트워크에서 통신할 수 있어야 합니다.

   ```
   DEFINE CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME('TOKYO.NETB.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
   channel using network B for TOKYO')
   ```

   ```
   DEFINE CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME('TOKYO.NETA.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
   channel using network A for TOKYO')
   ```
3. 큐 관리자 TOKYO에서 CLUSSDR 채널을 정의하십시오.

클러스터 내 모든 큐 관리자는 메시지를 해당 첫 번째 전체 저장소에 송신할 수 있는 한 개의 클러스터 송신자 채널을 정의해야 합니다. 이 경우 NEWYORK을 선택했으므로 TOKYO에는 다음 정의가 필요합니다.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
channel from TOKYO to repository at NEWYORK')
```

모든 정의를 마쳤으므로 아직 수행하지 않은 경우 IBM MQ for z/OS에서 채널 시작기를 시작하십시오. 모든 플랫폼의 큐 관리자 PARIS에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 청취하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

결과

314 페이지의 그림 62에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

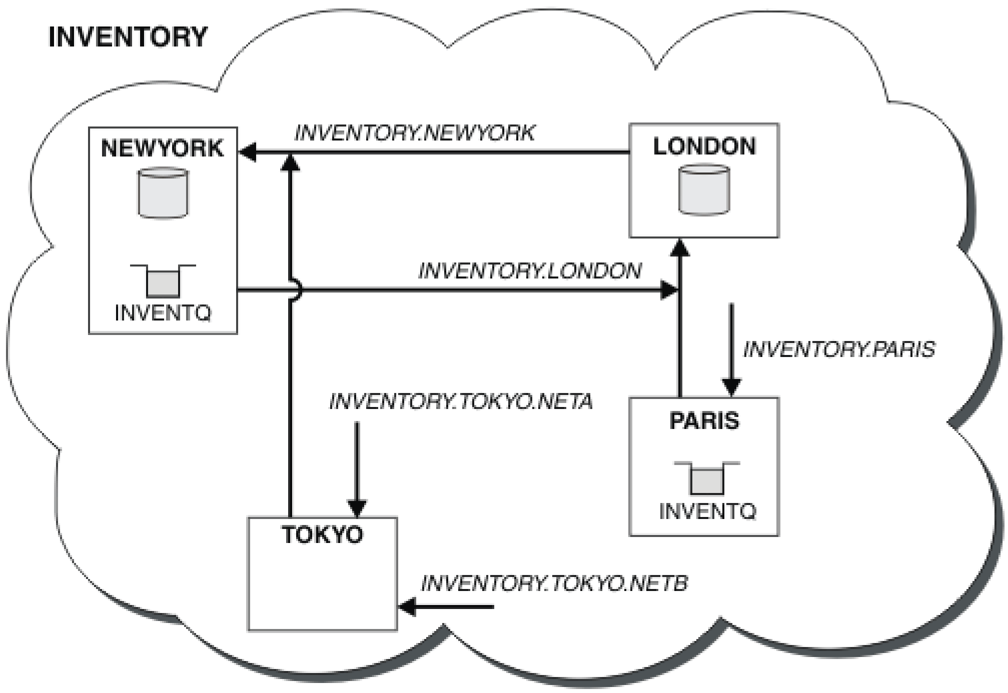


그림 *62.* 네 개의 큐 관리자가 있는 *INVENTORY* 클러스터

세 개의 정의만 작성하여 큐 관리자 TOKYO를 사용 가능한 두 개의 다른 네트워크 루트가 있는 클러스터에 추가했습니다.

관련 개념

둘 이상의 큐 인스턴스가 있는 클러스터의 예
둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

애플리케이션 프로그래밍 및 클러스터
동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

237 페이지의 『클러스터에 큐 관리자 추가』
다음 지시사항에 따라 사용자가 작성한 클러스터에 큐 관리자를 추가합니다. 클러스터 큐 및 토픽에 대한 메시지는 단일 클러스터 전송 큐 SYSTEM.CLUSTER.TRANSMIT.QUEUE를 사용하여 전송됩니다.

클러스터에서 *1*차 및 보조 네트워크 사용

다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 313 페이지의 『클러스터에서 두 개의 네트워크 사용』에서 설명된 대로 설정되었습니다. 이 클러스터는 네 개의 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS 및 TOKYO는 부분 저장소를 보유합니다. 인벤토리 애플리케이션은 큐 관리자 NEWYORK에 연결된 New York의 시스템에서 실행됩니다. TOKYO 큐 관리자에는 통신에 사용할 수 있는 두 개의 다른 네트워크가 있습니다.
- 이러한 네트워크 중 하나는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성하려고 합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용할 계획입니다.

이 태스크 정보

NETPRTY 속성을 사용하여 클러스터에서 1차 및 보조 네트워크를 구성하십시오.

프로시저

TOKYO에서 기존 CLUSRCVR 채널을 대체하십시오.

네트워크 A 채널이 1차 채널이고 네트워크 B가 보조 채널임을 표시하려면 다음 명령을 사용하십시오.

a) `ALTER CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) NETPRTY(2) DESCR('Main cluster-receiver channel for TOKYO')`

b) `ALTER CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) NETPRTY(1) DESCR('Backup cluster-receiver channel for TOKYO')`

다음에 수행할 작업

서로 다른 네트워크 우선순위로 채널을 구성함으로써 클러스터에 대해 1차 네트워크 및 보조 네트워크가 있음을 구성했습니다. 이러한 채널을 사용하는 클러스터의 큐 관리자는 사용 가능한 경우 자동으로 1차 네트워크를 사용합니다. 1차 네트워크가 사용 불가능한 경우 큐 관리자는 보조 네트워크를 사용하기 위해 장애를 복구합니다.
관련 개념
둘 이상의 큐 인스턴스가 있는 클러스터의 예
둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

애플리케이션 프로그래밍 및 클러스터

동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 두 개의 네트워크 사용
다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

백업 역할을 하는 큐 추가

다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되었습니다. 이 클러스터는 세 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS는 부분 저장소를 보유합니다. 인벤토리 애플리케이션은 NEWYORK 큐 관리자에 연결된 New York의 시스템에서 실행됩니다. 애플리케이션은 INVENTQ 큐에 메시지가 도착하면 구동됩니다.
- New York에서 실행되는 인벤토리 시스템에 대한 백업을 제공하기 위해 Chicago에서 새 저장소가 설정됩니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

이 태스크 정보

다음 단계에 따라 백업 역할을 할 큐를 추가하십시오.

프로시저

1. CHICAGO가 처음 참조할 전체 저장소를 결정하십시오.

   클러스터 내 모든 큐 관리자는 클러스터에 대한 정보를 수집하기 위해 전체 저장소 중 어느 하나를 참조해야 합니다. 그러면 자체 부분 저장소를 빌드합니다. 특정 큐 관리자에 대해 선택하는 저장소가 특별히 중요하지는 않습니다. 이 예에서는 NEWYORK을 선택합니다. 새 큐 관리자는 클러스터에 조인하고 나면 두 저장소와 모두 통신합니다.
2. CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자를 정의해야 합니다. CHICAGO에서 다음을 정의하십시오.

```
DEFINE CHANNEL(INVENTORY.CHICAGO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(CHICAGO.CMSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel for CHICAGO')
```

3. 큐 관리자 CHICAGO에서 CLUSSDR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 해당 첫 번째 전체 저장소에 송신할 수 있는 한 개의 클러스터 송신자 채널을 정의해야 합니다. 이 경우 NEWYORK을 선택했으므로 CHICAGO에는 다음 정의가 필요합니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
   channel from CHICAGO to repository at NEWYORK')
   ```

4. 기존 클러스터 큐 INVENTQ를 대체하십시오.

   이미 NEWYORK 큐 관리자가 호스팅하고 있는 INVENTQ가 큐의 기본 인스턴스입니다.

   ```
   ALTER QLOCAL(INVENTQ) CLWLPRTY(2)
   ```

5. 메시지 연관관계에 대해 인벤토리 애플리케이션을 검토하십시오.

   계속하기 전에 메시지 처리 순서에 따라 인벤토리 애플리케이션에 종속 항목이 없는지 확인하십시오.

6. CHICAGO에 있는 시스템에 인벤토리 애플리케이션을 설치하십시오.
7. 백업 클러스터 큐 INVENTQ를 정의하십시오.

   이미 NEWYORK 큐 관리자가 호스팅하고 있는 INVENTQ 큐도 CHICAGO에 의해 백업으로 호스팅됩니다. 다음과 같이 이 큐를 CHICAGO 큐 관리자에서 정의하십시오.

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) CLWLPRTY(1)
   ```

   모든 정의를 마쳤으므로 아직 수행하지 않은 경우 IBM MQ for z/OS에서 채널 시작기를 시작하십시오. 모든 플랫폼의 큐 관리자 CHICAGO에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 청취하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

결과

317 페이지의 그림 63에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

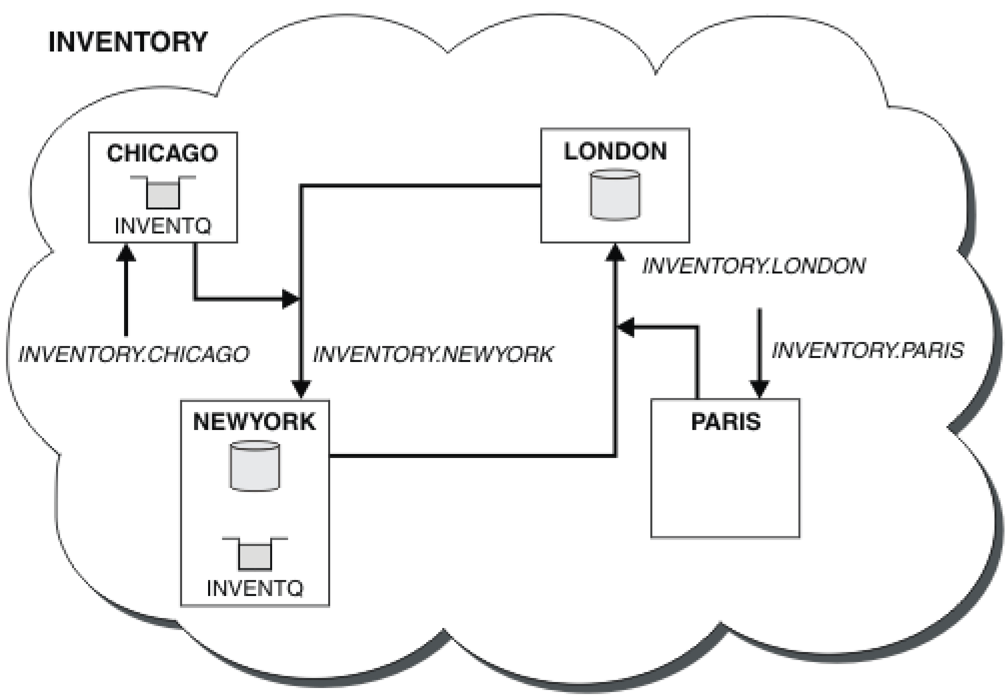


그림 *63.* 네 개의 큐 관리자가 있는 *INVENTORY* 클러스터

INVENTQ 큐 및 인벤토리 애플리케이션은 이제 클러스터에 있는 두 개의 큐 관리자에서 호스팅됩니다. CHICAGO 큐 관리자는 백업입니다. INVENTQ에 넣은 메시지는 CHICAGO로 대신 송신될 때 사용 불가능하지 않으면 NEWYORK에 라우트됩니다.

참고:

리모트 큐 관리자의 가용성은 해당 큐 관리자에 대한 채널의 상태를 기준으로 합니다. 채널이 시작될 때 해당 상태는 여러 번 변경되며, 일부 상태는 클러스터 워크로드 관리 알고리즘에 대한 우선권이 낮습니다. 실제로 이는 상위 우선순위(1차) 목적지에 대한 채널이 시작되는 동안 하위 우선순위(백업) 목적지를 선택할 수 있음을 의미합니다.

백업 목적지로 메시지가 이동하지 않도록 해야 하는 경우 CLWLPRTY를 사용하지 마십시오. 1차에서 백업으로의 수동 전환에 개별 큐 또는 CLWLRANK 사용을 고려하십시오.

관련 개념

둘 이상의 큐 인스턴스가 있는 클러스터의 예
둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

애플리케이션 프로그래밍 및 클러스터
동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 두 개의 네트워크 사용
다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

사용되는 채널 수 제한

다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- 가격 검사 애플리케이션은 다양한 큐 관리자에 설치됩니다. 사용되는 채널의 수를 적은 수로 유지하기 위해 각 서버가 실행하는 활성 채널의 수가 제한됩니다. 애플리케이션은 PRICEQ 큐에 메시지가 도착하면 구동됩니다.
- 네 개의 서버 큐 관리자가 가격 검사 애플리케이션을 호스팅합니다. 두 조회 큐 관리자가 가격을 조회하기 위한 메시지를 PRICEQ에 송신합니다. 두 개의 추가 큐 관리자가 전체 저장소로 구성됩니다.

이 태스크 정보

다음 단계에 따라 사용되는 채널의 수를 제한하십시오.

프로시저

1. 두 개의 전체 저장소를 선택하십시오.

   가격 검사 클러스터의 전체 저장소가 될 두 개의 큐 관리자를 선택하십시오. 이를 REPOS1 및 REPOS2라고 합니다.

   다음 명령을 실행하십시오.

   ```
   ALTER QMGR REPOS(PRICECHECK)
   ```
2. 각 큐 관리자에서 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 각 큐 관리자에서 클러스터 수신자 채널 및 클러스터 송신자 채널을 정의하십시오. 어떤 채널이 처음 정의되는지는 중요하지 않습니다.

   ```
   DEFINE CHANNEL(PRICECHECK.SERVE1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(SERVER1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-receiver channel')
   ```
3. 각 큐 관리자에서 CLUSSDR 채널을 정의하십시오.

   각 큐 관리자에서 CLUSSDR 정의를 작성하여 해당 큐 관리자를 전체 저장소 큐 관리자 중 어느 하나에 링크하십시오.

   ```
   DEFINE CHANNEL(PRICECHECK.REPOS1) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(REPOS1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-sender channel to
   repository queue manager')
   ```
4. 가격 검사 애플리케이션을 설치하십시오.
5. 모든 서버 큐 관리자에서 PRICEQ 큐를 정의하십시오.

   각각에서 다음 명령을 실행하십시오.

   ```
   DEFINE QLOCAL(PRICEQ) CLUSTER(PRICECHECK)
   ```
6. 조회에 사용되는 채널의 수를 제한하십시오.

   조회 큐 관리자에서 각각에 다음 명령을 발행하여 사용되는 활성 채널의 수를 제한합니다.

   ```
   ALTER QMGR CLWLMRUC(2)
   ```
7. 아직 수행하지 않은 경우 IBM MQ for z/OS에서 채널 시작기를 시작하십시오. 모든 플랫폼에서 리스너 프로그램을 시작하십시오.

   리스너 프로그램은 수신되는 네트워크 요청을 청취하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

결과

320 페이지의 그림 64에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

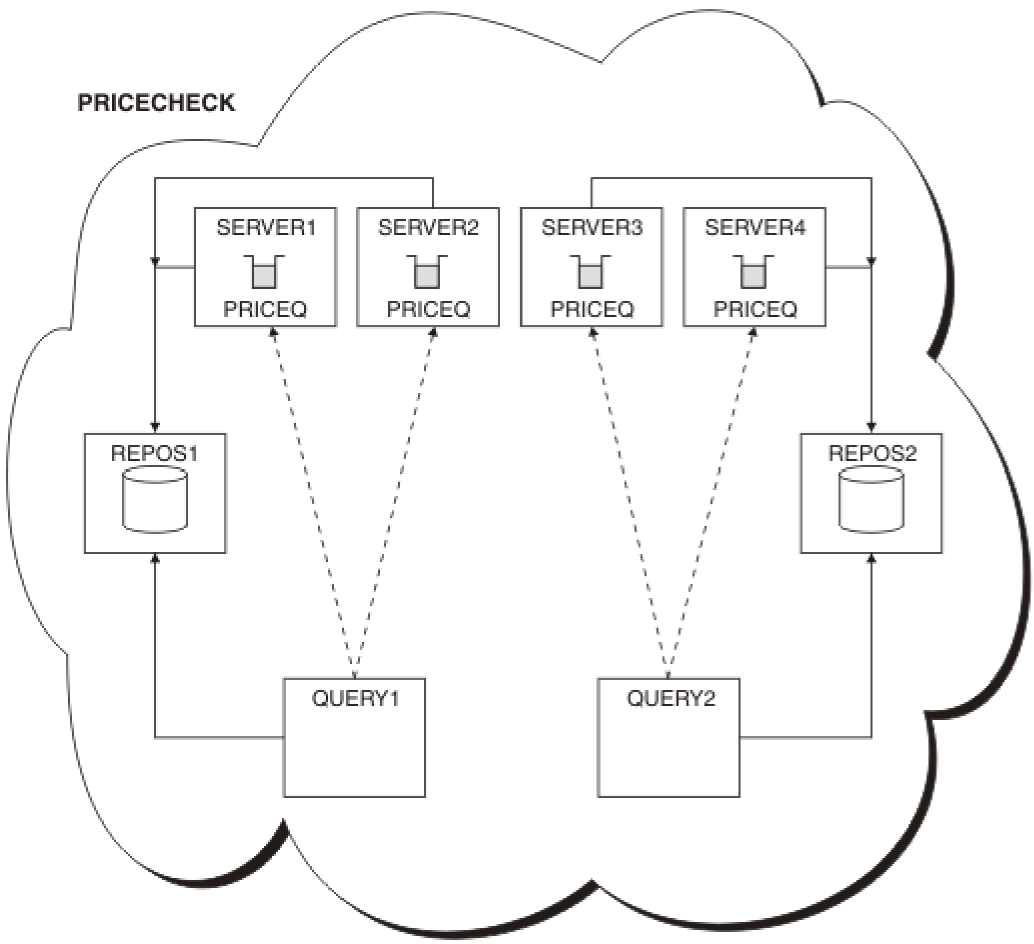


그림 *64.* 네 개의 서버 큐 관리자, 두 개의 저장소 및 두 개의 조회 큐 관리자가 포함된 *PRICECHECK* 클러스터

PRICECHECK 클러스터에 사용 가능한 네 개의 PRICEQ 큐 인스턴스가 있지만 각 조회 큐 관리자는 그 중 두 개의 인스턴스만 사용합니다. 예를 들어, QUERY1 큐 관리자에만 SERVER1 및 SERVER2 큐 관리자에 대한 활성 채널이 있습니다. SERVER1이 사용 불가능한 상태가 되면 QUERY1 큐 관리자는 다른 큐 관리자(예: SERVER3)를 사용하기 시작합니다.

관련 개념

둘 이상의 큐 인스턴스가 있는 클러스터의 예
둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

애플리케이션 프로그래밍 및 클러스터
동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 두 개의 네트워크 사용
다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가

다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

시작하기 전에

참고: 클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

시나리오:

- INVENTORY 클러스터가 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되었습니다. 이 클러스터는 세 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS는 부분 저장소를 보유하고 INVENTQ에서 메시지를 넣습니다. 인벤토리 애플리케이션은 NEWYORK 큐 관리자에 연결된 New York의 시스템에서 실행됩니다. 애플리케이션은 INVENTQ 큐에 메시지가 도착하면 구동됩니다.
- 새 저장소가 Los Angeles에서 설정됩니다. 추가 용량을 제공하기 위해 New York뿐만 아니라 Los Angeles에서도 인벤토리 시스템을 실행하려고 합니다. 새 큐 관리자는 New York에 비해 두 배 많은 메시지를 처리할 수 있습니다.

이 태스크 정보

다음 단계에 따라 큐를 호스팅하는 보다 강력한 큐 관리자를 추가하십시오.

프로시저

1. LOSANGELES가 처음 참조할 전체 저장소를 결정하십시오.
2. 클러스터 내 모든 큐 관리자는 클러스터에 대한 정보를 수집하기 위해 전체 저장소 중 어느 하나를 참조해야 합니다. 그러면 자체 부분 저장소를 빌드합니다. 저장소 선택이 특별히 중요한 것은 아닙니다. 이 예에서는 NEWYORK을 선택합니다. 새 큐 관리자는 클러스터에 조인하고 나면 두 저장소와 모두 통신합니다.

   ```
   DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
   CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-sender channel from LOSANGELES to repository at NEWYORK')
   ```

3. 큐 관리자 LOSANGELES에서 CLUSRCVR 채널을 정의하십시오.

   클러스터 내 모든 큐 관리자는 메시지를 수신할 수 있는 클러스터 수신자 채널을 정의해야 합니다. LOSANGELES에서 다음을 정의하십시오.

   ```
   DEFINE CHANNEL(INVENTORY.LOSANGELES) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
   CONNAME(LOSANGELES.CHSTORE.COM) CLUSTER(INVENTORY)
   DESCR('Cluster-receiver channel for queue manager LOSANGELES')
   CLWLWGHT(2)
   ```

   클러스터 수신자 채널은 클러스터 INVENTORY에 있는 다른 큐 관리자로부터 메시지를 수신할 큐 관리자의 가용성을 알립니다. CLWLWGHT를 2로 설정하면 Los Angeles 큐 관리자가 New York에 비해 두 배 많은 인벤토리 메시지를 가져오도록 보장합니다(NEWYORK에 대한 채널이 1로 설정된 경우).
4. 큐 관리자 NEWYORK에서 CLUSRCVR 채널을 대체하십시오.

   Los Angeles 큐 관리자가 New York에 비해 두 배 많은 인벤토리 메시지를 가져오도록 하십시오. 클러스터 수신자 채널의 정의를 대체하십시오.

```
ALTER CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) CLWLWGHT(1)
```

5. 메시지 연관관계에 대해 인벤토리 애플리케이션을 검토하십시오.

   계속하기 전에 메시지 처리 순서에 따라 인벤토리 애플리케이션에 종속 항목이 없는지 확인하십시오.
6. Los Angeles에 있는 시스템에 인벤토리 애플리케이션을 설치하십시오.
7. 클러스터 큐 INVENTQ를 정의하십시오.

   이미 NEWYORK 큐 관리자가 호스팅하고 있는 INVENTQ 큐도 LOSANGELES가 호스팅합니다. 다음과 같이 이 큐를 LOSANGELES 큐 관리자에서 정의하십시오.

   ```
   DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
   ```

   모든 정의를 마쳤으므로 아직 수행하지 않은 경우 IBM MQ for z/OS에서 채널 시작기를 시작하십시오. 모든 플랫폼의 큐 관리자 LOSANGELES에서 리스너 프로그램을 시작하십시오. 리스너 프로그램은 수신되는 네트워크 요청을 청취하고 필요한 경우 클러스터 수신자 채널을 시작합니다.

결과

321 페이지의 『큐를 호스팅하는 더 강력한 큐 관리자 추가』에서는 이 태스크에 의해 설정되는 클러스터를 보여줍니다.

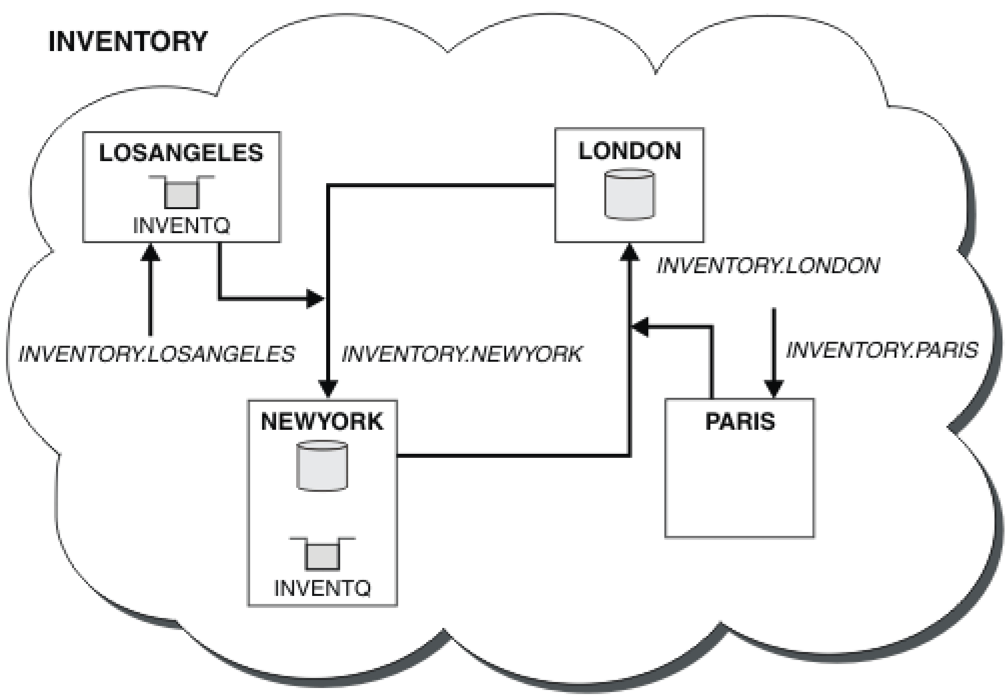


그림 *65.* 네 개의 큐 관리자가 있는 *INVENTORY* 클러스터

클러스터에 대한 이러한 수정은 큐 관리자 LONDON 및 PARIS를 변경하지 않고도 수행할 수 있습니다. 이러한 큐 관리자의 저장소는 메시지를 LOSANGELES의 INVENTQ에 송신하기 위해 필요한 정보로 자동 업데이트됩니다.

다음에 수행할 작업

INVENTQ 큐 및 인벤토리 애플리케이션은 클러스터에 있는 두 개의 큐 관리자에서 호스팅됩니다. 이 구성은 두 큐 관리자의 가용성을 증가시키고 메시지 처리 속도를 높이고 두 큐 관리자 사이에 워크로드를 분산시킬 수 있습니다. LOSANGELES 또는 NEWYORK에 의해 INVENTQ에 넣어진 메시지는 가능할 때마다 로컬 큐 관리자에서 인스턴스에 의해 처리됩니다. LONDON 또는 PARIS에 의해 넣어진 메시지가 LOSANGELES 또는 NEWYORK으로 라우트되며 두 배 많은 메시지가 LOSANGELES로 송신됩니다.

관련 개념

둘 이상의 큐 인스턴스가 있는 클러스터의 예
둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

애플리케이션 프로그래밍 및 클러스터
동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 `INVENTQ`의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 두 개의 네트워크 사용
다음 지시사항에 따라 `TOKYO`에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

애플리케이션 프로그래밍 및 클러스터

동일한 큐의 다중 인스턴스를 이용하기 위해 프로그래밍 변경을 작성하지 않아도 됩니다. 그러나 일부 프로그램은 메시지의 시퀀스가 큐의 동일 인스턴스로 송신되지 않으면 올바르게 작동하지 않습니다.

애플리케이션은 `MQOPEN` 호출을 사용하여 큐를 열 수 있습니다. 애플리케이션은 `MQPUT` 호출을 사용하여 메시지를 열린 큐에 넣습니다. 애플리케이션은 `MQPUT1` 호출을 사용하여 아직 열리지 않은 큐에 단일 메시지를 넣을 수 있습니다.

동일 큐의 다중 인스턴스가 있는 클러스터를 설정하는 경우 특정 애플리케이션 프로그래밍 고려사항이 없습니다. 그러나 클러스터링의 워크로드 관리 측면에서 이득을 얻기 위해 애플리케이션을 수정해야 할 수도 있습니다. 동일한 큐의 다중 정의가 있는 네트워크를 설정할 경우 메시지 연관관계에 대해 애플리케이션을 검토하십시오.

예를 들어, 질문과 응답 양식으로 그 사이에서 플로우하는 일련의 메시지에 의존하는 두 애플리케이션이 있다고 가정하십시오. 질문을 송신한 큐 관리자에게 응답이 돌아가게 하려고 할 수 있습니다. 워크로드 관리 루틴이 응답 큐의 사본을 호스팅하는 어떤 큐 관리자에게도 메시지를 송신하지 않는 것이 중요합니다.

메시지가 순서대로 처리하도록 요구하는 애플리케이션(예: 순서대로 검색되어야 하는 메시지의 배치를 송신하는 데이터베이스 복제 애플리케이션)이 있을 수 있습니다. 세그먼트화된 메시지의 사용은 연관관계 문제점의 원인이 될 수도 있습니다.

로컬 또는 리모트 버전의 대상 큐 열기

큐 관리자가 대상 큐의 로컬 버전을 사용할지 또는 리모트 버전을 사용할지 선택하는 방식을 알아야 합니다.

1. 큐 관리자는 로컬 버전 대상 큐를 열어 메시지를 읽거나 큐의 속성을 설정합니다.
2. 큐 관리자는 다음 조건 중 최소한 하나 이상이 true이면 대상 큐의 인스턴스를 열어 메시지를 기록합니다.
   - 로컬 버전 대상 큐는 존재하지 않습니다.
   - 큐 관리자는 `ALTER QMGR`에서 `CLWLUSEQ(ANY)`를 지정합니다.
   - 큐 관리자의 큐가 `CLWLUSEQ(ANY)`를 지정합니다.

관련 개념

둘 이상의 큐 인스턴스가 있는 클러스터의 예

둘 이상의 큐 인스턴스가 있는 이 클러스터 예에서 메시지는 큐의 다른 인스턴스로 라우트됩니다. 큐의 특정 인스턴스에 대해 메시지를 강제 실행하고 메시지 시퀀스를 큐 관리자 중 하나에 송신하도록 선택할 수 있습니다.

관련 태스크

큐를 로컬로 호스팅하는 큐 관리자 추가
다음 지시사항에 따라 INVENTQ의 인스턴스를 추가하여 Paris와 New York에서 인벤토리 애플리케이션 시스템을 실행하기 위한 추가 용량을 제공합니다.

클러스터에서 두 개의 네트워크 사용
다음 지시사항에 따라 TOKYO에 두 개의 다른 네트워크가 있는 새 저장소를 추가합니다. 두 네트워크 모두 Tokyo의 큐 관리자와 통신하는 데 사용 가능해야 합니다.

클러스터에서 1차 및 보조 네트워크 사용
다음 지시사항에 따라 한 네트워크는 1차 네트워크로 하고 다른 네트워크는 백업 네트워크로 구성합니다. 1차 네트워크에 문제점이 있을 경우 백업 네트워크를 사용합니다.

백업 역할을 하는 큐 추가
다음 지시사항에 따라 New York에서 실행되는 인벤토리 시스템에 대한 백업을 Chicago에 제공합니다. 이 Chicago 시스템은 New York 시스템에 문제점이 있는 경우에만 사용됩니다.

사용되는 채널 수 제한
다음 지시사항에 따라 다양한 큐 관리자에 가격 검사 애플리케이션이 설치될 때 각 서버가 실행하는 활성 채널의 수를 제한합니다.

큐를 호스팅하는 더 강력한 큐 관리자 추가
다음 지시사항에 따라 인벤토리 시스템을 New York뿐만 아니라 Los Angeles에서도 실행하여 추가 용량을 제공합니다. 여기서 Los Angeles는 New York에 비해 메시지를 두 배로 처리할 수 있습니다.

메시지 연관관계 핸들링
메시지 연관관계는 우수한 프로그래밍 설계에는 거의 포함되지 않습니다. 클러스터링을 완전하게 사용하려면 메시지 연관관계를 제거해야 합니다. 메시지 연관관계를 제거할 수 없는 경우 같은 채널을 사용하여 관련 메시지가 같은 큐 관리자에 전달되도록 강제 실행할 수 있습니다.

메시지 연관관계가 포함된 애플리케이션이 있는 경우 클러스터 사용을 시작하기 전에 이 연관관계를 제거하십시오.

메시지 연관관계를 제거하면 애플리케이션의 가용성을 향상시킵니다. 애플리케이션이 메시지 연관관계가 있는 메시지의 배치를 큐 관리자에게 송신합니다. 큐 관리자가 배치의 일부만 수신한 경우에는 실패합니다. 송신 큐 관리자가 추가 메시지를 송신하려면 먼저 불완전한 메시지 배치가 복구되어 처리될 때까지 대기해야 합니다.

메시지 연관관계를 제거하면 애플리케이션의 확장성도 향상시킵니다. 연관관계가 있는 메시지 배치는 후속 메시지를 대기하는 동안 목적지 큐 관리자에서 자원을 잠글 수 있습니다. 이러한 자원은 다른 애플리케이션이 해당 작업을 수행하지 못하도록 장기간 동안 잠금 상태로 남아 있을 수 있습니다.

또한 메시지 연관관계는 클러스터 워크로드 관리 루틴이 최적의 큐 관리자를 선택하지 못하도록 합니다.

연관관계를 제거하려면 다음과 같은 가능성을 고려하십시오.

- 메시지로 상태 정보 운반
- 모든 큐 관리자가 액세스할 수 있는 비휘발성 스토리지(예: Db2 데이터베이스)에서 상태 정보 유지보수
- 둘 이상의 큐 관리자가 액세스할 수 있도록 읽기 전용 데이터 복제

메시지 연관관계를 제거하기 위해 애플리케이션을 수정하는 것이 적절하지 않은 경우 이 문제점에 대해 가능한 솔루션이 다수 있습니다.

MQOPEN 호출에서 특정 목적지의 이름 지정

MQOPEN 호출 시마다 리모트 큐 이름 및 큐 관리자 이름을 지정하면 해당 오브젝트 핸들을 사용하여 큐에 넣어진 모든 메시지가 로컬 큐 관리자일 수도 있는 같은 큐 관리자로 이동합니다.

MQOPEN 호출 시마다 리모트 큐 이름 및 큐 관리자 이름을 지정하는 것에는 다음과 같은 단점이 있습니다.

- 워크로드 밸런싱이 수행되지 않습니다. 클러스터 워크로드 밸런싱의 이점을 이용하지 않습니다.
- 대상 큐 관리자가 리모트이고 이 큐 관리자에 대해 둘 이상의 채널이 있을 경우 메시지는 다른 라우트를 선택할 수 있고 메시지 시퀀스는 보존되지 않습니다.

- 큐 관리자에 목적지 큐 관리자와 같은 이름을 가지는 전송 큐에 대한 정의가 있는 경우 메시지는 클러스터 전송 큐가 아닌 해당 전송 큐로 이동합니다.

응답 대상 큐 관리자 필드에 큐 관리자 이름 리턴

배치의 첫 번째 메시지를 수신하는 큐 관리자가 해당 이름을 해당 응답으로 리턴할 수 있게 합니다. 이는 메시지 디스크립터의 ReplyToQMgr 필드를 사용하여 수행됩니다. 그러면 송신 측 큐 관리자가 응답 대상 큐 관리자 이름을 추출하고 이를 모든 후속 메시지에 지정할 수 있습니다.

응답의 ReplyToQMgr 정보를 사용하는 것에는 다음과 같은 단점이 있습니다.

- 요청 큐 관리자는 첫 번째 메시지에 대한 응답을 대기해야 합니다.
- 후속 메시지를 송신하기 전에 ReplyToQMgr 정보를 찾고 사용하기 위한 추가 코드를 작성해야 합니다.
- 큐 관리자에 대한 라우트가 둘 이상 있는 경우 메시지 시퀀스가 보존되지 않을 수 있습니다.

MQOPEN 호출에서 MQOO_BIND_ON_OPEN 옵션을 설정하십시오.

MQOPEN 호출의 MQOO_BIND_ON_OPEN 옵션을 사용하여 모든 메시지를 동일한 대상에 강제 실행하십시오. 클러스터와 함께 메시지 그룹 을 사용하여 그룹의 모든 메시지가 동일한 목적지에서 처리되도록 하려면 MQOO_BIND_ON_OPEN 또는 MQOO_BIND_ON_GROUP 를 지정해야 합니다.

큐를 열고 MQOO_BIND_ON_OPEN을 지정하여 이 큐로 송신되는 모든 메시지가 동일한 큐 인스턴스로 송신되도록 강제 실행합니다. MQOO_BIND_ON_OPEN은 모든 메시지를 같은 큐 관리자와 같은 라우트에 바인드합니다. 예를 들어, 동일한 목적지에 대한 IP 라우트 및 NetBIOS 라우트가 있는 경우 큐가 열릴 때 이들 중 하나가 선택되며 이러한 선택은 확보된 오브젝트 핸들을 사용하여 같은 큐에 들어가는 모든 메시지에 대해 준수됩니다.

MQOO_BIND_ON_OPEN을 지정하여 모든 메시지가 같은 목적지로 라우트되도록 강제 실행합니다. 따라서 메시지 연관관계가 있는 애플리케이션은 중단되지 않습니다. 목적지가 사용 불가능한 경우에는 사용 가능해질 때까지 메시지가 전송 큐에 남게 됩니다.

MQOO_BIND_ON_OPEN은 큐를 열 때 큐 관리자 이름이 오브젝트 디스크립터에서 지정되는 경우에도 적용됩니다. 이름 지정된 큐 관리자에 대한 라우트가 둘 이상 있을 수 있습니다. 예를 들어, 다중 네트워크 경로가 있거나 다른 큐 관리자가 알리어스를 정의했을 수 있습니다. MQOO_BIND_ON_OPEN을 지정하는 경우 큐가 열리면 라우트가 선택됩니다.

참고: 이 방법이 권장되는 기술입니다. 그러나 이 방법은 큐 관리자가 클러스터 큐에 대한 알리어스를 알리는 멀티홉 구성에서는 작동하지 않습니다. 또한 애플리케이션이 여러 메시지 그룹에 대해 같은 큐 관리자에서 다른 큐를 사용하는 상황에서도 도움이 되지 않습니다.

MQOPEN 호출에서 MQOO_BIND_ON_OPEN을 지정하는 것에 대한 대안은 큐 정의를 수정하는 것입니다. 큐 정의에서 DEFBIND(OPEN)을 지정하고 MQOPEN 호출의 DefBind 옵션이 MQOO_BIND_AS_Q_DEF로 기본값 설정되도록 하십시오.

MQOPEN 호출에서 MQOO_BIND_ON_GROUP 옵션을 설정하십시오.

Force all your messages in a group to be put to the same destination using the MQOO_BIND_ON_GROUP option on the MQOPEN call. 클러스터와 함께 메시지 그룹 을 사용하여 그룹의 모든 메시지가 동일한 목적지에서 처리되도록 하려면 MQOO_BIND_ON_OPEN 또는 MQOO_BIND_ON_GROUP 를 지정해야 합니다.

큐를 열고 MQOO_BIND_ON_GROUP을 지정하여 이 큐로 송신되는 모든 메시지가 동일한 큐 인스턴스로 송신되도록 강제 실행합니다. MQOO_BIND_ON_GROUP은 그룹 내 모든 메시지를 같은 큐 관리자와 같은 라우트에 바인드합니다. 예를 들어, 동일한 목적지에 대한 IP 라우트 및 NetBIOS 라우트가 있는 경우 큐가 열릴 때 이들 중 하나가 선택되며 이러한 선택은 확보된 오브젝트 핸들을 사용하여 같은 큐에 들어가는 그룹 내 모든 메시지에 대해 준수됩니다.

MQOO_BIND_ON_GROUP을 지정하여 그룹 내 모든 메시지가 같은 목적지로 라우트되도록 강제 실행합니다. 따라서 메시지 연관관계가 있는 애플리케이션은 중단되지 않습니다. 목적지가 사용 불가능한 경우에는 사용 가능해질 때까지 메시지가 전송 큐에 남게 됩니다.

MQOO_BIND_ON_GROUP은 큐를 열 때 큐 관리자 이름이 오브젝트 디스크립터에서 지정되는 경우에도 적용됩니다. 이름 지정된 큐 관리자에 대한 라우트가 둘 이상 있을 수 있습니다. 예를 들어, 다중 네트워크 경로가 있거나

다른 큐 관리자가 알리어스를 정의했을 수 있습니다. MQOO_BIND_ON_GROUP을 지정하는 경우 큐가 열리면 라우트가 선택됩니다.

MQOO_BIND_ON_GROUP이 유효하려면 MQPUT에서 MQPMO_LOGICAL_ORDER Put 옵션을 포함해야 합니다. 메시지의 MQMD에서 **GroupId**를 MQGI_NONE으로 설정할 수 있으며, 메시지의 MQMD **MsgFlags** 필드 내에 다음을 메시지 플래그를 포함해야 합니다.

- 그룹의 마지막 메시지: MQMF_LAST_MSG_IN_GROUP
- 그룹의 기타 모든 메시지: MQMF_MSG_IN_GROUP

MQOO_BIND_ON_GROUP이 지정되지만 메시지가 그룹화되지 않는 경우 이 작동은 MQOO_BIND_NOT_FIXED와 동일합니다.

참고: 이는 그룹 내 메시지가 같은 목적지에 송신되도록 보장하기 위해 권장되는 기술입니다. 그러나 큐 관리자가 클러스터 큐에 대한 별명을 알리는 다중 홉 구성에서는 작동하지 않습니다.

MQOPEN 호출에서 MQOO_BIND_ON_GROUP을 지정하는 것에 대한 대안은 큐 정의를 수정하는 것입니다. 큐 정의에서 DEFBIND(GROUP)을 지정하고 MQOPEN 호출의 DefBind 옵션이 MQOO_BIND_AS_Q_DEF로 기본값 설정되도록 하십시오.

사용자 정의된 클러스터 워크로드 엑시트 프로그램 작성

애플리케이션을 수정하는 대신 클러스터 워크로드 엑시트 프로그램을 작성하여 메시지 연관관계 문제점을 피할 수 있습니다. 클러스터 워크로드 엑시트 프로그램을 작성하는 것은 쉽지 않고 권장되는 솔루션이 아닙니다. 프로그램은 메시지의 컨텐츠를 검사하여 연관관계를 인식하도록 설계되어야 합니다. 연관관계를 인식하면 프로그램이 모든 관련 메시지를 동일한 큐 관리자에 라우트하기 위한 워크로드 관리 유틸리티를 강제 실행합니다.

발행/구독 메시징 구성

큐 대기된 발행/구독의 상태를 시작, 중지 또는 표시할 수 있습니다. 스트림을 추가하거나 제거할 수 있고 브로커 계층에서 큐 관리자를 추가하고 삭제할 수도 있습니다.

프로시저

- 큐된 발행/구독 제어에 대한 자세한 정보는 다음 하위 주제를 참조하십시오.
  - 326 페이지의 『큐된 발행/구독 메시지 속성 설정』
  - 328 페이지의 『큐된 발행/구독 시작』
  - 328 페이지의 『큐된 발행/구독 중지』
  - 329 페이지의 『스트림 추가』
  - 330 페이지의 『스트림 삭제』
  - 330 페이지의 『구독 지점 추가』
  - 337 페이지의 『발행/구독 네트워크에서 토픽 공간 결합』

큐된 발행/구독 메시지 속성 설정

큐 관리자 속성을 사용하여 일부 발행/구독 메시지 속성의 작동을 제어합니다.다른 속성은 qm.ini 파일의 *Broker* 스탠자에서 제어하는 속성입니다.

이 태스크 정보

다음 발행/구독 속성을 설정할 수 있습니다. 자세한 내용은 큐 관리자 매개변수를 참조하십시오.

표 *25.* 발행/구독 구성 매개변수

설명	MQSC 매개변수 이름
명령 메시지 재시도 수	**PSRTYCNT**

표 25. 발행/구독 구성 매개변수 (계속)	
설명	**MQSC 매개변수 이름**
전달할 수 없는 명령 입력 메시지 제거	**PSNPMSG**
전달할 수 없는 명령 응답 메시지 다음의 작동	**PSNPRES**
동기점에서 명령 메시지 처리	**PSSYNCPT**

브로커 스탠자는 다음 구성 설정을 관리하는 데 사용됩니다.

- `PersistentPublishRetry=yes | force`

  Yes를 지정한 다음 큐된 발행/구독 인터페이스를 통한 지속 메시지의 발행이 실패하고 부정적인 응답이 요청되지 않은 경우 발행 조작이 재시도됩니다.

  부정적 응답 메시지를 요청한 경우 부정적 응답이 송신되고 추가 재시도는 발생하지 않습니다.

  Force를 지정한 다음 큐된 발행/구독 인터페이스를 통한 지속 메시지 발행이 실패하는 경우 성공적으로 처리될 때까지 발행 조작이 재시도됩니다. 부정적인 응답이 송신되지 않습니다.

- `NonPersistentPublishRetry=yes | force`

  Yes를 지정한 다음 큐된 발행/구독 인터페이스를 통한 비지속 메시지의 발행이 실패하고 부정적인 응답이 요청되지 않은 경우 발행 조작이 재시도됩니다.

  부정적 응답 메시지를 요청한 경우 부정적 응답이 송신되고 추가 재시도는 발생하지 않습니다.

  Force를 지정한 다음 큐된 발행/구독 인터페이스를 통한 비지속 메시지의 발행이 실패하는 경우 성공적으로 처리될 때까지 발행 조작이 재시도됩니다. 부정적인 응답이 송신되지 않습니다.

  참고: 비지속 메시지에 대해 이 기능을 사용하려는 경우 `NonPersistentPublishRetry` 값을 설정하는 것 외에도 큐 관리자 속성 **PSSYNCPT**가 Yes로 설정되는지 확인해야 합니다.

  이를 수행하면 STREAM 큐의 **MQGET**이 이제 동기점 아래에서 발생하므로 비지속적 발행물 처리 성능에 영향을 줄 수도 있습니다.

- `PublishBatchSize=`*number*

  브로커는 일반적으로 동기점 내 발행 메시지를 처리합니다. 각 발행물을 개별적으로 커미트하는 것은 비효율적일 수 있고 일부 환경에서 브로커는 단일 작업 단위로 여러 발행 메시지를 처리할 수 있습니다. 이 매개변수는 단일 작업 단위로 처리할 수 있는 최대 발행 메시지 수를 지정합니다.

  `PublishBatchSize`의 기본값은 5입니다.

- `PublishBatchInterval=`*number*

  브로커는 일반적으로 동기점 내 발행 메시지를 처리합니다. 각 발행물을 개별적으로 커미트하는 것은 비효율적일 수 있고 일부 환경에서 브로커는 단일 작업 단위로 여러 발행 메시지를 처리할 수 있습니다. 이 매개변수는 배치 내 첫 번째 메시지와 같은 배치에 포함된 후속 발행물 간 최대 시간(밀리초 단위)을 지정합니다.

  배치 간격이 0이면 메시지가 즉시 사용 가능하다는 전제 하에 최대 `PublishBatchSize`개의 메시지를 처리할 수 있음을 나타내는 것입니다.

  `PublishBatchInterval`의 기본값은 0입니다.

프로시저

IBM MQ 탐색기, 프로그래밍 가능한 명령 또는 **runmqsc** 명령을 사용하여 발행/구독의 작동을 제어하는 큐 관리자 속성을 대체하십시오.

예

ALTER QMGR PSNPRES(SAFE)

큐된 발행/구독 시작

큐 관리자의 PSMODE 속성을 설정하여 큐된 발행/구독을 시작하십시오.

시작하기 전에

PSMODE에 대한 설명을 읽고 발행/구독의 다음 세 가지 모드에 대해 이해하십시오.

- `COMPAT`
- `DISABLED`
- `ENABLED`

이 태스크 정보

큐된 발행/구독 인터페이스(브로커라고도 함) 또는 발행/구독 엔진(버전 7 발행/구독이라고도 함)또는 이 둘 다를 시작하려면 QMGR PSMODE 속성을 설정하십시오. 큐된 발행/구독을 시작하려면 PSMODE를 `ENABLED`로 설정해야 합니다. 기본값은 `ENABLED`입니다.

프로시저

IBM MQ Explorer 또는 **runmqsc** 명령을 사용하여 아직 사용으로 설정되지 않은 큐된 발행/구독 인터페이스를 사용으로 설정하십시오.

예

ALTER QMGR `PSMODE (ENABLED)`

다음에 수행할 작업

IBM MQ는 큐된 발행/구독 명령 및 발행/구독 메시지 큐 인터페이스(MQI) 호출을 처리합니다.

큐된 발행/구독 중지

큐 관리자의 PSMODE 속성을 설정하여 큐된 발행/구독을 중지하십시오.

시작하기 전에

PSMODE에 대한 설명을 읽고 발행/구독의 다음 세 가지 모드에 대해 이해하십시오.

- COMPAT
- DISABLED
- ENABLED

이 태스크 정보

큐된 발행/구독 인터페이스(브로커라고도 함) 또는 발행/구독 엔진(버전 7 발행/구독이라고도 함)을 중지하거나 이 둘 다를 중지하려면 QMGR PSMODE 속성을 설정하십시오. 큐된 발행/구독을 중지하려면 PSMODE를 `COMPAT`로 설정해야 합니다. 발행/구독 엔진 전체를 중지하려면 PSMODE를 `DISABLED`로 설정하십시오.

프로시저

IBM MQ Explorer 또는 **runmqsc** 명령을 사용하여 큐된 발행/구독 인터페이스를 사용 안함으로 설정하십시오.

예

ALTER QMGR `PSMODE (COMPAT)`

스트림 추가

애플리케이션 간 데이터 격리 또는 IBM WebSphere MQ 6 발행/구독 계층과의 상호 운용을 가능하게 하기 위해 수동으로 스트림을 추가할 수 있습니다.

시작하기 전에

발행/구독 스트림이 작동하는 방식을 익히십시오. 스트림과 토픽을 참조하십시오.

이 태스크 정보

PCF 명령 **runmqsc** 또는 IBM MQ Explorer를 사용하여 다음 단계를 수행하십시오.

참고: 순서에 관계없이 단계 1과 단계 2를 수행할 수 있습니다. 단계 1 및 단계 2를 둘 다 완료한 후에는 단계 3만 수행하십시오.

프로시저

1. IBM WebSphere MQ 6 스트림과 같은 이름을 사용하여 로컬 큐를 정의하십시오.
2. IBM WebSphere MQ 6 스트림과 같은 이름을 사용하여 로컬 토픽을 정의하십시오.
3. 큐 이름을 이름 목록 `SYSTEM.QPUBSUB.QUEUE.NAMELIST`에 추가하십시오.
4. 발행/구독 계층에서 IBM WebSphere MQ 7.1 이상의 모든 큐 관리자에 대해 반복하십시오.

`'Sport'` 추가

스트림 `'Sport'`, IBM WebSphere MQ 6 및 7.1 큐 관리자를 공유하는 예제에서는 동일한 발행/구독 계층에서 작업이 수행됩니다. IBM WebSphere MQ 6 큐 관리자는 `'Sport'`라고 하는 스트림을 공유합니다. 이 예는 `'Sport'`라고 하는 IBM WebSphere MQ 7.1 큐 관리자에서 IBM WebSphere MQ 6 스트림 `'Sport'`와 공유하는 토픽 문자열 `'Sport'`를 사용하여 큐 및 토픽을 작성하는 방법을 보여줍니다.

토픽 문자열 `'Soccer/Results'`를 사용하여 토픽 `'Sport'`에 발행하는 IBM WebSphere MQ 7.1 발행 애플리케이션은 결과 토픽 문자열 `'Sport/Soccer/Results'`를 작성합니다. IBM WebSphere MQ 7.1 큐 관리자에서 토픽 문자열 `'Soccer/Results'`를 사용하여 토픽 `'Sport'`에 대한 구독자는 발행물을 수신합니다.

IBM WebSphere MQ 6 큐 관리자에서 스트림 `'Sport'`에 대한 구독자는 토픽 문자열 `'Soccer/Results'`를 사용하여 발행물을 수신합니다.

```
runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
define qlocal('Sport')
     1 : define qlocal('Sport')
AMQ8006: IBM MQ queue created.
define topic('Sport') topicstr('Sport')
     2 : define topic('Sport') topicstr('Sport')
AMQ8690: IBM MQ topic created.
alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
     3 : alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
AMQ8551: IBM MQ namelist changed.
```

참고: 이름 목록 오브젝트에 있는 기존 이름 외에도 추가 중인 새 이름을 **alter namelist** 명령에 제공해야 합니다.

다음에 수행할 작업

스트림에 대한 정보는 계층에 있는 다른 브로커에게 전달됩니다.

브로커가 버전 6인 경우, 브로커를 IBM WebSphere MQ 6 브로커로 관리하십시오. 즉, 스트림 큐를 수동으로 작성하거나 필요한 경우 브로커가 스트림 큐를 동적으로 작성하게 하는 것 중에 선택할 수 있습니다. 이 큐는 모델 큐 정의 `SYSTEM.BROKER.MODEL.STREAM`을 기반으로 합니다.

브로커가 7.1버전인 경우, 계층 구조에 있는 각 IBM WebSphere MQ 7.1 큐 관리자를 수동으로 구성해야 합니다.

스트림 삭제

IBM WebSphere MQ 7.1 이상 큐 관리자에서 스트림을 삭제할 수 있습니다.

시작하기 전에

스트림을 삭제하기 전에 스트림에 대해 남아 있는 구독이 없는지 확인하고 스트림을 사용하는 모든 애플리케이션을 일시정지해야 합니다. 발행물이 계속해서 삭제된 스트림으로 플로우되는 경우 시스템을 깨끗하게 작동되는 상태로 복원하려면 많은 관리 노력이 필요합니다.

프로시저

1. 이 스트림을 호스팅하는 모든 연결된 브로커를 찾으십시오.
2. 모든 브로커에서 스트림에 대한 모든 구독을 취소하십시오.
3. 이름 목록 `SYSTEM.QPUBSUB.QUEUE.NAMELIST`에서 큐(스트림과 같은 이름을 가진 큐)를 제거하십시오.
4. 스트림과 이름이 같은 큐에서 모든 메시지를 삭제하거나 영구 제거하십시오.
5. 스트림과 이름이 같은 큐를 삭제하십시오.
6. 연관된 토픽 오브젝트를 삭제하십시오.

다음에 수행할 작업

스트림을 호스팅하는 다른 모든 연결된 IBM WebSphere MQ 7.1 이상 큐 관리자에서 단계 3 - 5를 반복하십시오.

구독 지점 추가

새 구독 지점을 사용하여 IBM Integration Bus의 이전 버전에서 마이그레이션한 기존 큐된 발행/구독 애플리케이션을 확장하는 방법입니다.

시작하기 전에

1. 등록 위치가 `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST`에 아직 정의되지 않았는지 확인하십시오.
2. 구독 지점과 이름이 같은 토픽 오브젝트 또는 토픽 문자열이 있는지 검사하십시오.

이 태스크 정보

IBM WebSphere MQ 7.1 이상 애플리케이션에서는 구독 지점을 사용하지 않지만 구독 지점 마이그레이션 메커니즘을 사용하여 구독 지점을 사용하는 기존 애플리케이션과 상호 운용할 수 있습니다.

중요사항: 구독 지점 마이그레이션 메커니즘은 IBM MQ 8.0에서 제거되었습니다. 기존 애플리케이션을 마이그레이션해야 하는 경우 최신 버전으로 마이그레이션하기 전에 사용자 제품 버전에 해당하는 문서에 설명된 프로시저를 수행해야 합니다.

구독 포인트는 IBM WebSphere MQ 6 또는 그 이전 버전에서 마이그레이션된 `MQRFH1` 헤더를 사용하는 대기 중인 발행/구독 프로그램에서 작동하지 않습니다.

IBM WebSphere MQ 7.1 이상을 위해 작성된 통합 발행/구독 애플리케이션을 사용하기 위해 구독 지점을 추가할 필요는 없습니다.

프로시저

1. 등록 위치 이름을 `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST`에 추가하십시오.
   - z/OS에서 **NLTYPE**은 기본값 `NONE`입니다.
   - 같은 발행/구독 토폴로지에 연결되어 있는 모든 큐 관리자에서 이 단계를 반복하십시오.

2. 구독 지점의 이름과 일치하는 토픽 문자열을 포함하는 토픽 오브젝트를 구독 지점의 이름을 부여하여 추가하십시오.
    - 구독 지점이 클러스터에 있는 경우 토픽 오브젝트를 클러스터 토픽 호스트의 클러스터 토픽으로 추가하십시오.
    - 구독 지점의 이름과 같은 토픽 문자열을 포함하는 토픽 오브젝트가 있을 경우 기존 토픽 오브젝트를 사용하십시오. 기존 토픽을 재사용하는 구독 지점의 결과에 대해 이해해야 합니다. 기존 토픽이 기존 애플리케이션의 일부인 경우 두 개의 동일하게 이름 지정된 토픽 간 충돌을 해결해야 합니다.
    - 구독 지점과 같은 이름, 다른 토픽 문자열을 갖는 토픽 오브젝트가 있을 경우 다른 이름을 갖는 토픽을 작성하십시오.
3. **`Topic`** 속성 `WILDCARD`를 값 `BLOCK`으로 설정하십시오.

   # 또는 *는 구독 지점에 대한 와일드카드 구독을 차단합니다(와일드카드 및 서브스크립션 포인트참조).
4. 토픽 오브젝트에서 필요한 모든 속성을 설정하십시오.

예

이 예제는 두 개의 등록 지점(USD 및 GBP)을 추가하는 **`runmqsc`** 명령 파일을 표시합니다.

```
DEFINE TOPIC(USD) TOPICSTR(USD)
DEFINE TOPIC(GBP) TOPICSTR(GBP) WILDCARD(BLOCK)
ALTER NL(SYSTEM.QPUBSUB.SUBPOINT.NAMELIST) NAMES(SYSTEM.BROKER.DEFAULT.SUBPOINT, USD, GBP)
```

참고:

1. **ALTER** 명령을 사용하여 추가되는 구독 지점의 목록에 기본 구독 지점을 포함시키십시오. **ALTER**는 이름 목록에서 기존 이름을 삭제합니다.
2. 이름 목록을 대체하기 전에 토픽을 정의하십시오. 큐 관리자는 큐 관리자가 시작할 때 및 이름 목록이 대체될 때에만 이름 목록을 검사합니다.

분산 발행/구독 네트워크 구성

분산 발행/구독 토폴로지에 함께 연결되는 큐 관리자는 공용 연합 토픽 공간을 공유합니다. 하나의 큐 관리자에서 작성된 구독은 토폴로지의 다른 큐 관리자에 연결된 애플리케이션이 발행한 메시지를 수신할 수 있습니다.

클러스터 또는 계층에서 큐 관리자를 모두 연결하여 작성된 토픽 공간의 범위를 제어할 수 있습니다. 발행/구독 클러스터에서는 클러스터에 분산된 토픽 공간의 각 분기마다 토픽 오브젝트가 "클러스터"되어야 합니다. 계층의 경우 계층에서 해당 '상위'를 식별하기 위해 각 큐 관리자가 구성되어야 합니다.

각 발행 및 구독이 로컬 또는 글로벌인지 여부를 선택하여 토폴로지 내에서 발행 및 구독 플로우를 보다 세부적으로 제어할 수 있습니다. 로컬 발행과 구독은 발행자 또는 구독자가 연결된 큐 관리자를 넘어 전파되지 않습니다.

관련 정보

분산 발행/구독 네트워크

발행 범위

구독 범위

토픽 공간

클러스터 토픽 정의

발행/구독 클러스터 구성

큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

시작하기 전에

일부 클러스터 구성은 직접 라우트된 발행/구독의 오버헤드를 수용할 수 없습니다. 이 구성을 사용하기 전에 발행/구독 클러스터 구성에 자세하게 설명된 고려사항 및 옵션을 탐색하십시오.

클러스터에 대한 변경이 클러스터 전체에 전파되도록 하려면 최소한 하나 이상의 전체 저장소가 항상 사용 가능해야 합니다. 이 태스크를 시작하기 전에 저장소가 사용 가능한지 확인하십시오.

발행/구독 클러스터에 대한 라우팅: 작동에 대한 참고도 참조하십시오.

시나리오:

- INVENTORY 클러스터가 237 페이지의 『클러스터에 큐 관리자 추가』에서 설명된 대로 설정되었습니다. 이 클러스터는 세 큐 관리자를 포함하며 LONDON 및 NEWYORK은 둘 다 전체 저장소를 보유하고 PARIS는 부분 저장소를 보유합니다.

이 태스크 정보

클러스터에 있는 큐 관리자에서 토픽을 정의할 때 이 토픽이 클러스터 토픽인지 여부와 (클러스터 토픽인 경우) 이 토픽의 발행물 및 구독에 대해 클러스터 내 라우팅을 지정해야 합니다. 토픽을 클러스터 토픽으로 작성하려면 클러스터의 이름을 사용하여 TOPIC 오브젝트에서 **CLUSTER** 특성을 구성합니다. 클러스터에서 큐 관리자에 대한 클러스터 토픽을 정의하면 토픽을 전체 클러스터에서 사용할 수 있게 됩니다. 클러스터에서 사용할 메시지 라우팅을 선택하려면 TOPIC 오브젝트의 **CLROUTE** 특성을 다음 값 중 하나로 설정합니다.

- **DIRECT**
- **TOPICHOST**

기본적으로 주제 라우팅은 **DIRECT**입니다. IBM MQ 8.0 이전에는 이 옵션이 유일한 옵션이었습니다. 큐 관리자에서 직접 라우트된 클러스터 토픽을 구성할 경우, 클러스터에 있는 모든 큐 관리자가 클러스터에 있는 다른 모든 큐 관리자를 인식하게 됩니다. 따라서 발행 및 구독 조작을 수행할 경우 큐 관리자가 각각 클러스터에 있는 다른 큐 관리자에 직접 연결될 수 있습니다. 직접 라우트된 발행/구독 클러스터의 내용을 참조하십시오.

IBM MQ 8.0에서 대신 주제 라우팅을 **TOPICHOST**으로 구성할 수 있습니다. 토픽 호스트 라우팅을 사용할 경우, 클러스터의 모든 큐 관리자가 라우팅되는 토픽 정의를 호스팅하는 클러스터 큐 관리자(토픽 오브젝트를 정의한 큐 관리자)를 인식하게 됩니다. 발행 및 구독 조작을 수행할 경우, 클러스터의 큐 관리자는 서로 직접 연결되지 않고 이러한 토픽 호스트 큐 관리자에만 연결됩니다. 토픽 호스트 큐 관리자는 구독이 일치하는 큐 관리자에 발행물을 발행하는 큐 관리자에서 발행물을 라우팅하는 작업을 담당합니다. 토픽 호스트 라우트된 발행/구독 클러스터의 내용을 참조하십시오.

참고: 주제 오브젝트가 클러스터된 후 (**CLUSTER** 특성 설정을 통해) **CLROUTE** 특성의 값을 변경할 수 없습니다. 값을 변경하려면 먼저 클러스터화를 해제해야 합니다(**CLUSTER**를 ' '로 설정). 토픽의 클러스터화를 해제하면 토픽 정의가 로컬 토픽으로 변환되어 이 기간 동안에는 발행물이 리모트 큐 관리자의 구독에 제공되지 않으므로, 이 변경을 수행할 때 이 사항을 고려해야 합니다. 다른 큐 관리자의 클러스터 토픽과 동일한 이름의 비클러스터 토픽을 정의할 경우 발생하는 결과를 참조하십시오. 클러스터되는 동안 **CLROUTE** 특성의 값을 변경하려는 경우, 시스템은 MQRCCF_CLROUTE_NOT_ALTERABLE 예외를 생성합니다.

프로시저

1. 토픽을 호스팅할 큐 관리자를 선택하십시오.

   클러스터 큐 관리자는 토픽을 호스팅할 수 있습니다. 세 큐 관리자(LONDON, NEWYORK 또는 PARIS) 중 하나를 선택하고 TOPIC 오브젝트의 특성을 구성하십시오. 직접 라우팅을 사용할 계획이면 큐 관리자 선택은 가동에 영향을 주지 않습니다. 토픽 호스트 라우팅을 사용할 계획이면 선택한 큐 관리자가 발행물 라우팅에 대해 추가적 책임을 집니다. 따라서 토픽 호스트 라우팅에 대해 보다 강력한 시스템 중 하나에서 호스팅되고 우수한 네트워크 연결성을 가지는 큐 관리자를 선택하십시오.
2. 큐 관리자에서 토픽 정의를 참조하십시오.

   토픽을 클러스터 토픽으로 작성하려면 토픽을 정의할 때 클러스터 이름을 포함시키고 이 토픽의 발행물 및 구독에 사용할 라우팅을 설정하십시오. 예를 들어, LONDON 큐 관리자에서 직접 라우팅 클러스터 토픽을 작성하려면 다음과 같이 토픽을 작성하십시오.

   ```
   DEFINE TOPIC(INVENTORY) TOPICSTR('/INVENTORY') CLUSTER(INVENTORY) CLROUTE(DIRECT)
   ```

   클러스터에서 큐 관리자에 대한 클러스터 토픽을 정의하면 토픽을 전체 클러스터에서 사용할 수 있게 됩니다.

CLROUTE 사용에 대한 자세한 정보는 DEFINE TOPIC (CLROUTE) 및 발행/구독 클러스터에 대한 라우팅: 작동에 대한 참고를 참조하십시오.

결과

클러스터가 토픽에 대한 발행물 및 구독을 수신할 준비가 되었습니다.

다음에 수행할 작업

토픽 호스트 라우트된 발행/구독 클러스터를 구성한 경우 이 토픽에 두 번째 토픽 호스트를 추가하려고 할 수 있습니다. 335 페이지의 『토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가』의 내용을 참조하십시오.

여러 발행/구독 클러스터가 있는 경우(예를 들어, 조직이 지리적으로 분산되어 있는 경우) 일부 클러스터 토픽을 모든 클러스터에 전파할 수 있습니다. 계층에서 클러스터를 연결하여 이를 수행할 수 있습니다. 340 페이지의 『다중 클러스터의 토픽 공간 결합』의 내용을 참조하십시오. 한 클러스터에서 다른 클러스터로 플로우되는 발행물을 제어할 수도 있습니다. 342 페이지의 『다중 클러스터에서 토픽 공간 결합 및 격리』의 내용을 참조하십시오.

관련 개념

발행 및 구독 범위 결합
IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

발행/구독 네트워크에서 토픽 공간 결합
발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

관련 태스크

다른 큐 관리자로 클러스터 토픽 정의 이동
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가
토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

큐 관리자를 발행/구독 계층에 연결
계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

발행/구독 계층에서 큐 관리자 연결 끊기
발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

관련 정보

발행/구독 클러스터 디자인
분배된 발행/구독 문제점 해결
클러스터된 발행/구독 금지

다른 큐 관리자로 클러스터 토픽 정의 이동

토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

이 태스크 정보

동일한 클러스터 토픽 오브젝트의 여러 정의가 클러스터에 존재할 수 있습니다. 이는 토픽 호스트 라우트 클러스터의 경우에는 정상 상태이고 직접 라우트 클러스터의 경우에는 비정상적인 상태입니다. 자세한 정보는 동일한 이름의 다중 클러스터 토픽 정의를 참조하십시오.

발행 플로우를 방해하지 않고 클러스터의 다른 큐 관리자로 클러스터 토픽 정의를 이동하려면 다음 단계를 완료하십시오. 프로시저는 큐 관리자 QM1에서 큐 관리자 QM2로 정의를 이동합니다.

프로시저

1. QM2에서 클러스터 토픽 정의의 복제본을 작성하십시오.

   직접 라우팅의 경우, 모든 속성이 QM1의 정의와 일치하도록 설정하십시오.

   토픽 호스트 라우팅의 경우, 처음에 새 토픽 호스트를 PUB(DISABLED)로 정의하십시오. 이렇게 하면 QM2가 클러스터 내 구독을 인식할 수는 있지만 라우팅 발행을 시작할 수는 없습니다.
2. 클러스터 전체에 정보가 전파될 때까지 기다리십시오.

   전체 저장소 큐 관리자가 클러스터 내 모든 큐 관리자로 새 클러스터 토픽 정의를 전파할 때까지 기다리십시오. **DISPLAY CLUSTER** 명령을 사용하여 각 클러스터 멤버에서 클러스터 토픽을 표시하고 QM2에서 가져온 정의를 확인하십시오.

   토픽 호스트 라우팅의 경우, QM2의 새 토픽 호스트가 모든 구독을 인식할 때까지 기다리십시오. QM2에 알려진 프록시 구독과 QM1에 알려진 프록시 구독을 비교하십시오. 큐 관리자의 프록시 구독을 볼 수 있는 한 가지 방법은 다음 **runmqsc** 명령을 실행하는 것입니다.

   ```
   DISPLAY SUB(*) SUBTYPE(PROXY)
   ```
3. 토픽 호스트 라우팅의 경우, QM2의 토픽 호스트를 PUB(ENABLED)로 재정의한 다음 QM1의 토픽 호스트를 PUB(DISABLED)로 재정의하십시오.

   QM2의 새 토픽 호스트가 다른 큐 관리자의 모든 구독을 인식했으므로 토픽 호스트가 발행물 라우팅을 시작할 수 있습니다.

   PUB(DISABLED) 설정을 사용하여 QM1을 통한 메시지 트래픽을 일시정지함으로써 클러스터 토픽 정의를 삭제할 때 QM1을 통해 진행되는 발행이 없는지 확인할 수 있습니다.
4. QM1에서 클러스터 토픽 정의를 삭제하십시오.

   QM1에서 정의를 삭제하려면 큐 관리자를 사용할 수 있어야 합니다. 그렇지 않으면 QM1이 재시작되거나 강제로 제거될 때까지 기존 두 정의를 모두 실행해야 합니다.

   QM1을 오랫동안 사용할 수 없고 그 기간 동안 QM2에서 클러스터된 토픽 정의를 수정해야 하는 경우에는 QM2 정의가 QM1 정의보다 최신 상태이므로 일반적으로 더 우세합니다.

   이 기간 동안 QM1과 QM2의 정의에 차이가 발생하면 두 큐 관리자 모두의 오류 로그에 오류가 기록되어 충돌 클러스터 토픽 정의를 경고합니다.

   QM1이 절대로 클러스터로 돌아가지 않는 경우(예를 들어, 하드웨어 실패 후 예상치 못한 압축 해제로 인해) 마지막 수단으로 RESET CLUSTER 명령을 사용하여 큐 관리자를 강제로 제거할 수 있습니다. **RESET CLUSTER**는 대상 큐 관리자에서 호스팅하는 모든 토픽 오브젝트를 자동으로 삭제합니다.

관련 개념

발행 및 구독 범위 결합
IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

발행/구독 네트워크에서 토픽 공간 결합
발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

관련 태스크

발행/구독 클러스터 구성
큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가

토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

큐 관리자를 발행/구독 계층에 연결
계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

발행/구독 계층에서 큐 관리자 연결 끊기
발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가

토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

시작하기 전에

여러 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의하는 것은 토픽 호스트 라우트 클러스터에만 기능적으로 유용합니다. 직접 라우트 클러스터에서 일치하는 여러 토픽을 정의해도 해당 작동은 변경되지 않습니다. 이 태스크는 토픽 호스트 라우트 클러스터에만 적용됩니다.

이 태스크는 동일한 이름의 다중 클러스터 토픽 정의 문서, 특히 다음 절의 내용을 읽은 것으로 가정합니다.

- 토픽 호스트 라우트 클러스터의 다중 클러스터 토픽 정의
- PUB 매개변수의 특수 처리

이 태스크 정보

큐 관리자가 라우트 토픽 호스트가 되면 먼저 클러스터에서 구독된 모든 관련 토픽의 존재를 인식해야 합니다. 추가 호스트 토픽이 추가되는 시점에 발행물이 해당 토픽에 발행되고 해당 호스트가 클러스터에서 다른 큐 관리자의 구독 존재를 인식하기 전에 새 호스트로 발행물이 라우트되는 경우에는 새 호스트가 해당 발행물을 해당 구독으로 전달하지 않습니다. 이 경우 구독에서 발행물이 누락됩니다.

발행물은 명시적으로 클러스터 토픽 오브젝트 **PUB** 매개변수를 사용 불가능으로 설정한 토픽 호스트 큐 관리자를 통해 라우트되지 않으므로, 이 설정을 사용하여 추가 토픽 호스트를 추가하는 프로세스 중에 구독이 누락되지 않도록 할 수 있습니다.

참고: 큐 관리자가 `PUB(DISABLED)`(으)로 정의된 클러스터 토픽을 호스트하는 동안, 해당 큐 관리자에 연결된 발행자는 메시지를 발행할 수 없으며, 해당 큐 관리자의 일치하는 구독은 클러스터의 다른 큐 관리자에 발행된 발행물을 수신하지 않습니다. 따라서 구독이 존재하고 발행 애플리케이션이 연결되는 경우 큐 관리자에 토픽 호스트 라우트 토픽을 정의하는 데 주의를 기울여야 합니다.

프로시저

1. 새 주제 호스트를 구성하고 초기에 새 주제 호스트를 `PUB(DISABLED)`(으)로 정의하십시오.

   이렇게 하면 새 토픽 호스트가 클러스터 내 구독을 인식할 수는 있지만 라우팅 발행을 시작할 수는 없습니다.

   토픽 호스트 구성에 대한 정보는 331 페이지의 『발행/구독 클러스터 구성』의 내용을 참조하십시오.
2. 새 토픽 호스트가 모든 구독을 인식한 시점을 판별하십시오.

   이를 수행하려면 새 토픽 호스트에 알려진 프록시 구독과 기존 토픽 호스트에 알려진 프록시 구독을 비교하십시오. 프록시 구독을 보는 한 가지 방법은 다음 **runmqsc** 명령을 발행하는 것입니다. `DISPLAY SUB(*) SUBTYPE(PROXY)`
3. 새 토픽 호스트를 `PUB(ENABLED)`(으)로 재정의하십시오.

   새 토픽 호스트가 다른 큐 관리자의 모든 구독을 인식한 후에는 토픽이 발행물 라우팅을 시작할 수 있습니다.

관련 개념

발행 및 구독 범위 결합
IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

발행/구독 네트워크에서 토픽 공간 결합
발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

관련 태스크

발행/구독 클러스터 구성
큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

다른 큐 관리자로 클러스터 토픽 정의 이동
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

큐 관리자를 발행/구독 계층에 연결
계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

발행/구독 계층에서 큐 관리자 연결 끊기
발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

발행 및 구독 범위 결합

IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

발행물은 발행/구독 토폴로지에서 연결된 모든 큐 관리자 또는 로컬 큐 관리자로만 이동할 수 있습니다. 이는 프록시 구독의 경우에도 유사합니다. 구독과 일치하는 발행물은 이 두 플로우의 조합으로 결정됩니다.

발행물과 구독물의 범위는 둘 다 QMGR 또는 ALL에 해당될 수 있습니다. 발행자와 구독자가 둘 다 동일한 큐 관리자에 연결된 경우에는 범위 설정이 구독자가 해당 발행자로부터 수신하는 발행물에 영향을 주지 않습니다.

발행자와 구독자가 다른 큐 관리자에 연결되어 있는 경우 원격 발행을 수신하려면 두 설정 모두 ALL이어야 합니다.

발행자가 여러 큐 관리자에 연결된 것으로 가정하십시오. 구독자가 다른 발행자로부터 발행물을 수신하도록 하려면 구독 범위를 ALL로 설정하십시오. 그런 다음 각 발행자마다 발행물 범위를 발행자의 로컬 구독자로 제한할지 여부를 결정할 수 있습니다.

구독자가 여러 큐 관리자에 연결된 것으로 가정하십시오. 발행자의 발행물을 모든 구독자로 보내려면 발행 범위를 ALL로 설정하십시오. 구독자가 동일한 큐 관리자에 연결된 발행자의 발행물만 수신하도록 하려면 구독 범위를 QMGR로 설정하십시오.

예: 축구 경기 결과 서비스

축구 리그에 소속된 팀인 것으로 가정하십시오. 각 팀마다 발행/구독 클러스터의 다른 팀 모두에 큐 관리자가 연결되어 있습니다.

팀은 Football/result/*Home team name*/*Away team name*(이)라는 주제를 사용하여 홈에서 플레이한 모든 게임의 결과를 공개합니다. 이탤릭체 문자열은 가변적인 토픽 이름이고 발행물은 경기 결과입니다.

또한 각 클럽은 Football/myteam/*Home team name*/*Away team name* 토픽 문자열을 사용하여 클럽에 대한 결과를 다시 공개합니다.

두 토픽 모두 전체 클러스터에 발행됩니다.

다음 구독은 모든 팀의 팬이 세 가지 흥미로운 방법으로 결과를 구독할 수 있도록 리그에서 설정되었습니다.

SUBSCOPE(QMGR)(으)로 클러스터 토픽을 설정할 수 있습니다. 토픽 정의는 클러스터의 각 멤버로 전파되지만 구독 범위는 로컬 큐 관리자뿐입니다. 따라서 각 큐 관리자의 구독자는 동일한 구독에서 다른 발행물을 수신합니다.

모든 결과 수신

```
DEFINE TOPIC(A) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(ALL)
```

모든 홈 결과 수신

```
DEFINE TOPIC(B) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(QMGR)
```

구독의 범위는 QMGR이므로 홈 그라운드에서 발행된 결과만 일치 여부를 확인합니다.

내 팀 결과 모두 수신

```
DEFINE TOPIC(C) TOPICSTR('Football/myteam/') CLUSTER SUBSCOPE(QMGR)
```

구독의 범위는 QMGR이므로 로컬로 재발행된 로컬 팀 결과의 일치 여부만을 확인합니다.

관련 개념

발행/구독 네트워크에서 토픽 공간 결합
발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

관련 태스크

발행/구독 클러스터 구성
큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

다른 큐 관리자로 클러스터 토픽 정의 이동
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가
토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

큐 관리자를 발행/구독 계층에 연결
계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

발행/구독 계층에서 큐 관리자 연결 끊기
발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

관련 정보

분산 발행/구독 네트워크
발행 범위
구독 범위

발행/구독 네트워크에서 토픽 공간 결합

발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

CLUSTER, **PUBSCOPE** 및 **SUBSCOPE** 속성, 발행/구독 클러스터 및 발행/구독 계층의 빌딩 블록을 사용하여 다른 발행/구독 토픽 공간을 작성할 수 있습니다.

다음 시나리오는 단일 큐 관리자에서 발행/구독 클러스터로의 확장 예부터 다양한 발행/구독 토폴로지를 보여줍니다.

관련 개념

발행 및 구독 범위 결합

IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

관련 태스크

발행/구독 클러스터 구성
큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

다른 큐 관리자로 클러스터 토픽 정의 이동
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가
토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

큐 관리자를 발행/구독 계층에 연결
계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

발행/구독 계층에서 큐 관리자 연결 끊기
발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

관련 정보

분산 발행/구독 네트워크
토픽 공간
클러스터 토픽 정의

발행/구독 클러스터에서 단일 토픽 공간 작성

발행/구독 시스템이 여러 큐 관리자에서 실행될 수 있도록 확장합니다. 발행/구독 클러스터를 사용하여 각 발행자 및 구독자에게 동일한 단일 토픽 공간을 제공하십시오.

시작하기 전에

단일 버전 7 큐 관리자에서 발행/구독 시스템을 구현했습니다.

항상 `SYSTEM.BASE.TOPIC`의 속성을 상속하는 것에 의존하지 않고 고유한 루트 주제로 토픽 공간을 작성하십시오. 발행/구독 시스템을 클러스터로 확장하는 경우 클러스터 토픽 호스트에서 루트 토픽을 클러스터 토픽으로 정의하면 클러스터 전체에서 모든 토픽을 공유합니다.

이 태스크 정보

이제 시스템이 보다 많은 발행자와 구독자를 지원하고 클러스터 전체에서 모든 토픽을 인식할 수 있도록 시스템을 확장할 수 있습니다.

프로시저

1. 발행/구독 시스템에 사용할 클러스터를 작성하십시오.

   전통적인 기존 클러스터가 있는 경우에는 성능상의 이유로 새 발행/구독 시스템에 새 클러스터를 설정하는 것이 좋습니다. 두 클러스터의 클러스터 저장소에 동일한 서버를 사용할 수 있습니다.
2. 클러스터 토픽 호스트가 될 하나의 큐 관리자(예를 들어, 저장소 중 하나)를 선택하십시오.
3. 발행/구독 클러스터 전체에서 인식되는 모든 토픽이 관리 토픽 오브젝트로 해석되어야 합니다.
   발행/구독 클러스터의 이름을 지정하는 **CLUSTER** 속성을 설정하십시오.

다음에 수행할 작업

발행자 및 구독자 애플리케이션을 클러스터 내 큐 관리자에 연결하십시오.

CLUSTER 속성이 있는 관리 토픽 오브젝트를 작성하십시오. 토픽은 또한 클러스터 전체에서 전파됩니다. 발행자 및 구독자 프로그램은 관리 토픽을 사용하므로 클러스터 내 다른 큐 관리자에 연결되어도 해당 작동은 변경되지 않습니다.

모든 큐 관리자에서 클러스터 토픽처럼 동작하려면 SYSTEM.BASE.TOPIC이(가) 필요한 경우 모든 큐 관리자에서 이를 수정해야 합니다.

관련 태스크

기존 IBM WebSphere MQ 6 토픽 공간에 버전 7 이상 큐 관리자 추가
기존 IBM WebSphere MQ 6 발행/구독 시스템이 동일한 큐 토픽 공간을 공유하는 버전 7 이상 큐 관리자와 상호 운용되도록 확장합니다.

다중 클러스터의 토픽 공간 결합
여러 클러스터에 분산되는 토픽 공간을 작성합니다. 특정 클러스터의 토픽에는 발행하고 다른 클러스터의 토픽은 구독합니다.

다중 클러스터에서 토픽 공간 결합 및 격리
일부 토픽 공간을 특정 클러스터로 분리하고 연결된 모든 클러스터에서 액세스할 수 있도록 다른 토픽 공간을 결합합니다.

복수 클러스터에서 토픽 공간 발행 및 구독
중첩된 클러스터를 사용하여 복수 클러스터에서 토픽을 발행, 구독합니다. 이 기술은 클러스터의 토픽 공간이 중첩되지 않는 경우 사용할 수 있습니다.

관련 정보

분산 발행/구독 네트워크
토픽 공간
클러스터 토픽 정의

기존 *IBM WebSphere MQ 6* 토픽 공간에 버전 7 이상 큐 관리자 추가

기존 IBM WebSphere MQ 6 발행/구독 시스템이 동일한 큐 토픽 공간을 공유하는 버전 7 이상 큐 관리자와 상호 운용되도록 확장합니다.

시작하기 전에

기존 IBM WebSphere MQ 6 발행/구독 시스템이 있습니다.

새 서버에 IBM WebSphere MQ 7 이상을 설치하고 큐 관리자를 구성했습니다.

이 태스크 정보

기존 IBM WebSphere MQ 6 발행/구독 시스템에 버전 7 이상 큐 관리자를 사용할 수 있도록 확장할 수 있습니다.

큐된 발행/구독 인터페이스를 사용하는 IBM WebSphere MQ 6 발행/구독 시스템을 안정적으로 개발하기로 결정했습니다. IBM WebSphere MQ 7 이상 MQI를 사용하여 시스템에 확장 기능을 추가하려고 합니다. 현재 큐된 발행/구독 애플리케이션을 재작성할 계획은 없습니다.

향후 IBM WebSphere MQ 6 큐 관리자를 IBM WebSphere MQ 7 이상으로 업그레이드할 계획입니다. 현재는 IBM WebSphere MQ 7 이상 큐 관리자에서 기존 큐된 발행/구독 애플리케이션을 계속 실행합니다.

프로시저

1. 하나의 송신자 수신자 채널 세트를 작성하여 IBM WebSphere MQ 7 이상 큐 관리자를 두 디렉토리의 IBM WebSphere MQ 6 큐 관리자 중 하나와 연결하십시오.
2. 대상 큐 관리자의 이름으로 두 개 전송 큐를 작성하십시오. 어떠한 이유로 대상 큐 관리자의 이름을 전송 큐 이름으로 사용할 수 없는 경우 큐 관리자 알리어스를 사용하십시오.
3. 송신자 채널을 트리거하도록 전송 큐를 구성하십시오.
4. IBM WebSphere MQ 6 발행/구독 시스템이 스트림을 사용하는 경우, 스트림 추가에 설명된 대로 스트림을 IBM WebSphere MQ 7 이상의 큐 관리자에 추가하십시오.
5. IBM WebSphere MQ 7 이상 큐 관리자 **PSMODE**가 ENABLE로 설정되었는지 확인하십시오.

6. 해당 **PARENT** 속성이 IBM WebSphere MQ 6 큐 관리자 중 하나를 참조하도록 대체하십시오.
7. 큐 관리자 간의 상위-하위 관계 상태가 양방향에서 모두 활성화되었는지 확인하십시오.

다음에 수행할 작업

태스크를 완료하면 IBM WebSphere MQ 6 및 7이상 큐 관리자가 동일한 주제 영역을 공유합니다. 예를 들어, 다음 태스크를 모두 수행할 수 있습니다.

- IBM WebSphere MQ 6 와 7이상의 큐 관리자 사이의 발행물 및 구독을 교환합니다.
- IBM WebSphere MQ 7 이상 큐 관리자에서 기존 IBM WebSphere MQ 6 발행/구독 프로그램을 실행합니다.
- IBM WebSphere MQ 6 또는 7이상의 큐 관리자에서 주제 영역을 보고 수정하십시오.
- IBM WebSphere MQ 7 이상 발행/구독 애플리케이션을 작성하고 IBM WebSphere MQ 7 이상 큐 관리자에서 실행합니다.
- IBM WebSphere MQ 7 이상 애플리케이션으로 새 발행물과 구독물을 작성하고 IBM WebSphere MQ 6 애플리케이션과 교환합니다.

관련 태스크

발행/구독 클러스터에서 단일 토픽 공간 작성
발행/구독 시스템이 여러 큐 관리자에서 실행될 수 있도록 확장합니다. 발행/구독 클러스터를 사용하여 각 발행자 및 구독자에게 동일한 단일 토픽 공간을 제공하십시오.

다중 클러스터의 토픽 공간 결합
여러 클러스터에 분산되는 토픽 공간을 작성합니다. 특정 클러스터의 토픽에는 발행하고 다른 클러스터의 토픽은 구독합니다.

다중 클러스터에서 토픽 공간 결합 및 격리
일부 토픽 공간을 특정 클러스터로 분리하고 연결된 모든 클러스터에서 액세스할 수 있도록 다른 토픽 공간을 결합합니다.

복수 클러스터에서 토픽 공간 발행 및 구독
중첩된 클러스터를 사용하여 복수 클러스터에서 토픽을 발행, 구독합니다. 이 기술은 클러스터의 토픽 공간이 중첩되지 않는 경우 사용할 수 있습니다.

관련 정보

분산 발행/구독 네트워크
토픽 공간
클러스터 토픽 정의

다중 클러스터의 토픽 공간 결합

여러 클러스터에 분산되는 토픽 공간을 작성합니다. 특정 클러스터의 토픽에는 발행하고 다른 클러스터의 토픽은 구독합니다.

시작하기 전에

이 태스크는 직접 라우트된 기존 발행/구독 클러스터가 있으며 일부 클러스터 토픽을 모든 클러스터에 전파하려는 것으로 가정합니다.

참고: 토픽 호스트 라우트 발행/구독 클러스터에는 이 작업을 수행할 수 없습니다.

이 태스크 정보

특정 클러스터에서 다른 클러스터로 발행물을 전파하려면 계층에서 클러스터를 함께 결합해야 합니다(341 페이지의 그림 66 참조). 계층적 연결이 연결된 큐 관리자 간에 구독물과 발행물을 전파하며 클러스터는 클러스터 사이가 아닌 각 클러스터 내에 클러스터 토픽을 전파합니다.

이 두 메커니즘을 결합하여 모든 클러스터 간에 클러스터 토픽을 전파할 수 있습니다. 각 클러스터에서 클러스터 토픽 정의를 반복해야 합니다.

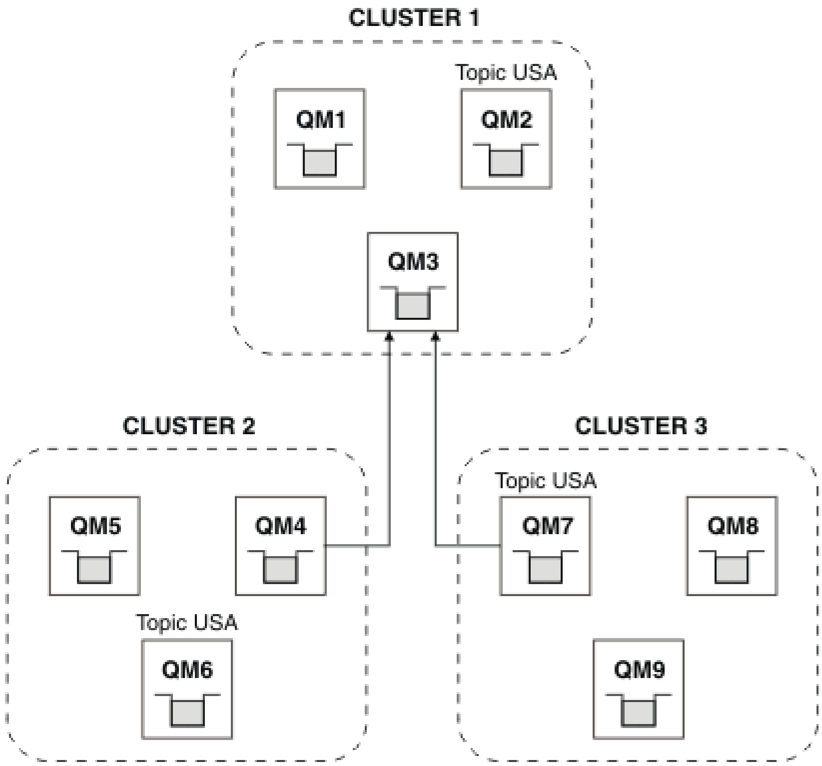


그림 66. 계층을 사용하여 클러스터 연결

다음 단계는 클러스터를 계층에 연결합니다.

프로시저

1. 두 방향 모두에서 QM3 및 QM4과(와) QM3 및 QM7을(를) 연결하려면 두 세트의 송신자-수신자 채널을 작성하십시오. 계층을 연결하기 위해 클러스터가 아닌 전통적인 송신자 수신자 채널과 전송 큐를 사용해야 합니다.
2. 대상 큐 관리자의 이름으로 세 개 전송 큐를 작성하십시오. 어떠한 이유로 대상 큐 관리자의 이름을 전송 큐 이름으로 사용할 수 없는 경우 큐 관리자 알리어스를 사용하십시오.
3. 송신자 채널을 트리거하도록 전송 큐를 구성하십시오.
4. QM3의 **PSMODE**, QM7의 QM4이(가) ENABLE로 설정되어 있는지 확인하십시오.
5. QM4 및 QM7의 **PARENT** 속성을 QM3(으)로 변경하십시오.
6. 큐 관리자 간의 상위-하위 관계 상태가 양방향에서 모두 활성화되었는지 확인하십시오.
7. 클러스터 1, 2 및 3에 있는 세 개의 클러스터 토픽 호스트 큐 관리자 각각에서 **CLUSTER**('CLUSTER 1'), **CLUSTER**('CLUSTER 2') 및 **CLUSTER**('CLUSTER 3') 속성을 사용하여 관리 토픽 USA을(를) 작성하십시오. 클러스터 토픽 호스트는 계층적으로 연결된 큐 관리자일 필요가 없습니다.

다음에 수행할 작업

이제 341 페이지의 그림 66의 클러스터 토픽 USA을(를) 발행하거나 구독할 수 있습니다. 발행물과 구독물은 세 개 클러스터 모두의 발행자와 구독자로 이동합니다.

다른 클러스터에서 USA을(를) 클러스터 토픽으로 작성하지 않았다고 가정하십시오. USA이(가) QM7에만 정의된 경우, USA에 대한 발행물과 구독은 QM7, QM8, QM9 및 QM3 사이에서 교환됩니다. QM7, QM8, QM9에서 실행되는 공개자 및 구독자는 관리 토픽 USA의 속성을 상속합니다. QM3의 공개자 및 구독자는 QM3에서 SYSTEM.BASE.TOPIC의 속성을 상속합니다.

342 페이지의 『다중 클러스터에서 토픽 공간 결합 및 격리』도 참조하십시오.

관련 태스크

발행/구독 클러스터에서 단일 토픽 공간 작성
발행/구독 시스템이 여러 큐 관리자에서 실행될 수 있도록 확장합니다. 발행/구독 클러스터를 사용하여 각 발행자 및 구독자에게 동일한 단일 토픽 공간을 제공하십시오.

기존 IBM WebSphere MQ 6 토픽 공간에 버전 7 이상 큐 관리자 추가
기존 IBM WebSphere MQ 6 발행/구독 시스템이 동일한 큐 토픽 공간을 공유하는 버전 7 이상 큐 관리자와 상호 운용되도록 확장합니다.

다중 클러스터에서 토픽 공간 결합 및 격리
일부 토픽 공간을 특정 클러스터로 분리하고 연결된 모든 클러스터에서 액세스할 수 있도록 다른 토픽 공간을 결합합니다.

복수 클러스터에서 토픽 공간 발행 및 구독
중첩된 클러스터를 사용하여 복수 클러스터에서 토픽을 발행, 구독합니다. 이 기술은 클러스터의 토픽 공간이 중첩되지 않는 경우 사용할 수 있습니다.

관련 정보

분산 발행/구독 네트워크

토픽 공간

클러스터 토픽 정의

다중 클러스터에서 토픽 공간 결합 및 격리

일부 토픽 공간을 특정 클러스터로 분리하고 연결된 모든 클러스터에서 액세스할 수 있도록 다른 토픽 공간을 결합합니다.

시작하기 전에

340 페이지의 『다중 클러스터의 토픽 공간 결합』 주제를 확인하십시오. 큐 관리자를 브릿지로 추가하지 않고도 요구를 충분히 충족시킬 수 있습니다.

참고: 이 태스크는 직접 라우트 발행/구독 클러스터를 사용해야 완료할 수 있습니다. 토픽 호스트 라우트 클러스터를 사용해서는 완료할 수 없습니다.

이 태스크 정보

340 페이지의 『다중 클러스터의 토픽 공간 결합』의 341 페이지의 그림 66에 표시된 토폴로지의 잠재적 개선 사항은 모든 클러스터에서 공유되지 않는 클러스터 토픽을 분리하는 것입니다. 클러스터에 없는 브릿지 큐 관리자를 작성하여 클러스터를 분리하십시오(343 페이지의 그림 67 참조). 브릿지 큐 관리자를 사용하면 특정 클러스터에서 다른 클러스터로 이동할 수 있는 발행물과 구독물을 필터링할 수 있습니다.

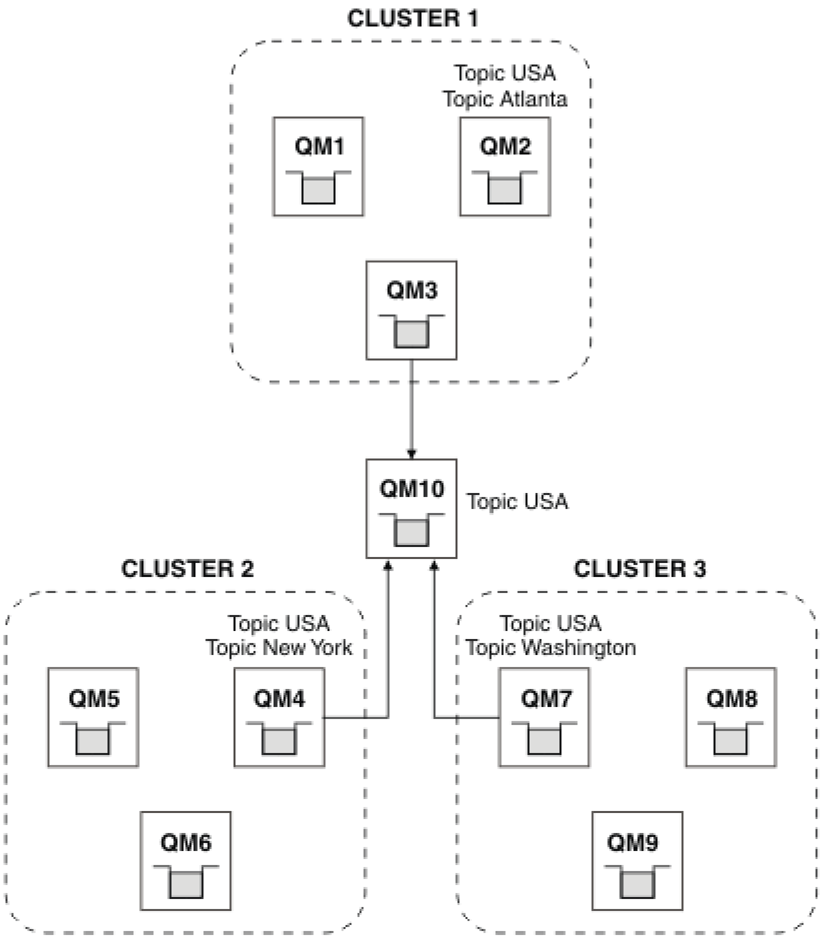


그림 67. 연결된 클러스터

다른 클러스터의 브릿지에서 공유하지 않을 클러스터 토픽을 분리하려면 브릿지를 사용하십시오. 343 페이지의 그림 67에서 USA은(는) 모든 클러스터에서 공유되는 클러스터 토픽이고, `Atlanta`, `New York` 및 `Washington`은(는) 각각 하나의 클러스터에서만 공유되는 클러스터 토픽입니다.

다음 프로시저를 사용하여 구성을 모델화하십시오.

프로시저

1. 모든 `SYSTEM.BASE.TOPIC` 토픽 오브젝트를 수정하여 **SUBSCOPE**(QMGR) 및 **PUBSCOPE**(QMGR)이(가) 모든 큐 관리자에 있도록 하십시오.

   명시적으로 **SUBSCOPE**(ALL) 및 **PUBSCOPE**(ALL)을(를) 클러스터 토픽의 루트 토픽에 설정하지 않는 한 주제(클러스터 토픽)도 다른 큐 관리자에 전파되지 않습니다.
2. **CLUSTER**(*clustername*), **SUBSCOPE**(ALL) 및 **PUBSCOPE**(ALL) 속성을 사용하여 각 클러스터에서 공유하려는 세 개의 클러스터 토픽 호스트 큐 관리자에 대한 토픽을 정의하십시오.

   일부 클러스터 토픽을 모든 클러스터에서 공유하려면 각 클러스터에서 동일한 토픽을 정의하십시오. 각 클러스터의 클러스터 이름을 클러스터 속성으로 사용하십시오.
3. 모든 클러스터 간에 공유하려는 클러스터 토픽의 경우 **SUBSCOPE**(ALL) 및 **PUBSCOPE**(ALL) 속성을 사용하여 브릿지 큐 관리자(QM10)에서 토픽을 다시 정의하십시오.

예

343 페이지의 그림 67의 예제에서는 USA에서 상속하는 주제만 세 개의 클러스터 사이에서 전파됩니다.

다음에 수행할 작업

SUBSCOPE(ALL)을(를) 사용하여 브릿지 큐 관리자에 정의된 토픽의 구독 및 **PUBSCOPE**(ALL)은(는) 클러스터 간에 전파됩니다.

속성이 **CLUSTER**(*clustername*), **SUBSCOPE**(ALL)인 각 클러스터 내에 정의된 토픽에 대한 구독 및 **PUBSCOPE**(ALL)은(는) 각 클러스터 내에서 전파됩니다.

다른 구독은 큐 관리자의 로컬입니다.

관련 태스크

발행/구독 클러스터에서 단일 토픽 공간 작성
발행/구독 시스템이 여러 큐 관리자에서 실행될 수 있도록 확장합니다. 발행/구독 클러스터를 사용하여 각 발행자 및 구독자에게 동일한 단일 토픽 공간을 제공하십시오.

기존 IBM WebSphere MQ 6 토픽 공간에 버전 7 이상 큐 관리자 추가
기존 IBM WebSphere MQ 6 발행/구독 시스템이 동일한 큐 토픽 공간을 공유하는 버전 7 이상 큐 관리자와 상호 운용되도록 확장합니다.

다중 클러스터의 토픽 공간 결합
여러 클러스터에 분산되는 토픽 공간을 작성합니다. 특정 클러스터의 토픽에는 발행하고 다른 클러스터의 토픽은 구독합니다.

복수 클러스터에서 토픽 공간 발행 및 구독
중첩된 클러스터를 사용하여 복수 클러스터에서 토픽을 발행, 구독합니다. 이 기술은 클러스터의 토픽 공간이 중첩되지 않는 경우 사용할 수 있습니다.

관련 정보

분산 발행/구독 네트워크
토픽 공간
클러스터 토픽 정의
발행 범위
구독 범위

복수 클러스터에서 토픽 공간 발행 및 구독

중첩된 클러스터를 사용하여 복수 클러스터에서 토픽을 발행, 구독합니다. 이 기술은 클러스터의 토픽 공간이 중첩되지 않는 경우 사용할 수 있습니다.

시작하기 전에

클러스터 간 교차 지점에서 일부 큐 관리자가 있는 여러 기존 클러스터를 작성하십시오.

이 태스크 정보

다양한 이유로 클러스터를 중첩시키도록 선택했을 수 있습니다.

1. 고가용성 서버 또는 큐 관리자의 수가 제한적입니다. 모든 클러스터 저장소를 배치하기로 결정하고 클러스터 토픽이 저장소를 호스팅합니다.
2. 게이트웨이 큐 관리자를 사용하여 연결되는 기존 큐 관리자 클러스터가 있습니다. 동일한 클러스터 토폴로지에 발행/구독 애플리케이션을 배치하려고 합니다.
3. 여러 가지 자체 포함 발행/구독 애플리케이션이 있습니다. 성능상의 이유로, 발행/구독 클러스터는 작게 또한 기존 클러스터와 별도로 유지하는 것이 좋습니다. 애플리케이션을 다른 클러스터에 배치하기로 결정했습니다. 그러나 모니터링 애플리케이션 사본 중 하나에 대한 라이센스만 있어 모든 발행/구독 애플리케이션을 하나의 큐 관리자에서 모니터하려고 합니다. 이 큐 관리자는 모든 클러스터의 클러스터 토픽 발행에 대한 액세스 권한이 있어야 합니다.

토픽이 중첩되지 않는 토픽 공간에 정의되었는지 확인함으로써 중첩 발행/구독 클러스터에 토픽을 배치할 수 있습니다(345 페이지의 그림 68 참조). 토픽 공간이 중첩될 때 중첩 클러스터에 배치하면 문제점이 발생합니다.

발행/구독 클러스터가 중첩되므로 중첩된 클러스터에서 큐 관리자를 사용하여 토픽 공간을 발행 및 구독할 수 있습니다.

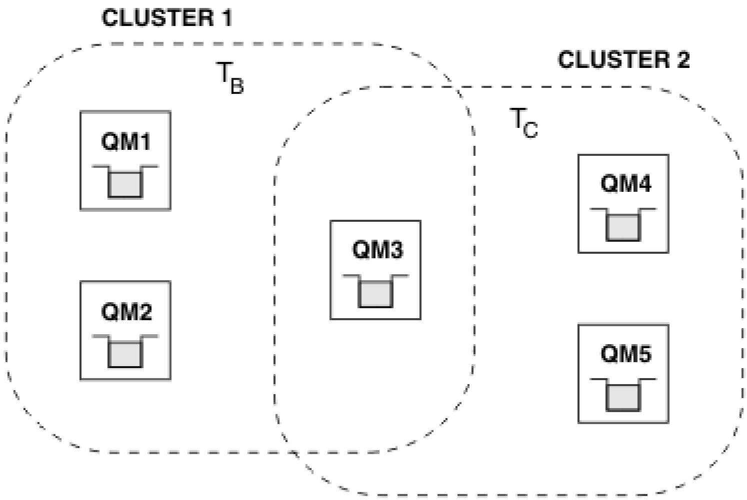


그림 *68.* 클러스터는 중첩되고 토픽 공간은 중첩되지 않는 경우

프로시저

토픽 공간이 중첩되지 않음을 확인할 수 있는 방법을 작성하십시오.
예를 들어, 각 토픽 공간에 고유한 루트 토픽을 정의하십시오. 루트 토픽을 클러스터 토픽을 만드십시오.

a) `DEFINE TOPIC(B) TOPICSTR('B') CLUSTER('CLUSTER 1') ...`

b) `DEFINE TOPIC(C) TOPICSTR('C') CLUSTER('CLUSTER 2') ...`

예

QM3에 연결된 게시자 및 구독자가 T_B 또는 T_C 에게 게시 또는 구독할 수 있는 345 페이지의 그림 68에서

다음에 수행할 작업

두 클러스터에서 모두 토픽을 사용하는 발행자와 구독자를 중첩된 클러스터의 큐 관리자로 연결하십시오.

특정 클러스터의 토픽만 사용해야 하는 발행자와 구독자를 중첩된 클러스터에 없는 큐 관리자에 연결하십시오.

관련 태스크

발행/구독 클러스터에서 단일 토픽 공간 작성
발행/구독 시스템이 여러 큐 관리자에서 실행될 수 있도록 확장합니다. 발행/구독 클러스터를 사용하여 각 발행자 및 구독자에게 동일한 단일 토픽 공간을 제공하십시오.

기존 IBM WebSphere MQ 6 토픽 공간에 버전 7 이상 큐 관리자 추가
기존 IBM WebSphere MQ 6 발행/구독 시스템이 동일한 큐 토픽 공간을 공유하는 버전 7 이상 큐 관리자와 상호 운용되도록 확장합니다.

다중 클러스터의 토픽 공간 결합
여러 클러스터에 분산되는 토픽 공간을 작성합니다. 특정 클러스터의 토픽에는 발행하고 다른 클러스터의 토픽은 구독합니다.

다중 클러스터에서 토픽 공간 결합 및 격리
일부 토픽 공간을 특정 클러스터로 분리하고 연결된 모든 클러스터에서 액세스할 수 있도록 다른 토픽 공간을 결합합니다.

관련 정보

분산 발행/구독 네트워크
토픽 공간
클러스터 토픽 정의

큐 관리자를 발행/구독 계층에 연결

계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

시작하기 전에

1. 발행/구독 계층의 큐 관리자는 고유한 큐 관리자 이름을 가져야 합니다.
2. 발행/구독 계층은 "큐잉된 발행/구독" 큐 관리자 기능에 의존합니다. 상위 및 하위 큐 관리자에서 모두 이 기능을 사용해야 합니다. 큐된 발행/구독 시작을 참조하십시오.
3. 발행/구독 관계는 큐 관리자 송신자 및 수신자 채널에 의존합니다. 채널은 다음 두 가지 방법으로 설정할 수 있습니다.
   - 상위 및 하위 큐 관리자를 모두 IBM MQ 클러스터에 추가합니다. 클러스터에 큐 관리자 추가를 참조하십시오.
   - 하위 큐 관리자에서 상위로, 상위에서 하위로의 송신자/수신자 채널 쌍을 설정하십시오. 각 채널은 대상 큐 관리자와 이름이 같은 전송 큐 또는 대상 큐 관리자와 이름이 같은 큐 관리자 알리어스를 사용해야 합니다. 포인트-투-포인트 채널 연결 설정 방법에 대한 자세한 정보는 IBM MQ 분산 큐잉 기술을 참조하십시오.

   각 채널 구성 유형에 계층을 구성하는 예는 다음 발행/구독 계층 시나리오 세트를 참조하십시오.

   - 시나리오 1: 큐 관리자 이름 알리어스가 있는 포인트-투-포인트 채널 사용
   - 시나리오 2: 전송 큐 및 리모트 큐 관리자에 대해 이름이 동일한 포인트-투-포인트 채널 사용
   - 시나리오 3: 클러스터 채널을 사용하여 큐 관리자 추가

이 태스크 정보

`ALTER QMGR PARENT (`*PARENT_NAME*`)` **runmqsc** 명령을 사용하여 하위를 상위에 연결하십시오. 이 구성은 하위 큐 관리자에서 수행됩니다. 여기서 *PARENT_NAME*은 상위 큐 관리자의 이름입니다.

프로시저

ALTER QMGR PARENT(*PARENT_NAME*)

예

첫 번째 예는 큐 관리자 QM2를 QM1의 하위로 첨부한 다음 QM2를 조회하여 ACTIVE의 **STATUS**로 하위가 되었는지 확인하는 방법을 보여줍니다.

```
C:>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2
alter qmgr parent(QM1)
     1 : alter qmgr parent(QM1)
AMQ8005: IBM MQ queue manager changed.
display pubsub all
     2 : display pubsub all
AMQ8723: Display pub/sub status details.
   QMNAME(QM2)                           TYPE(LOCAL)
   STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
   QMNAME(QM1)                           TYPE(PARENT)
STATUS(ACTIVE)
```

다음 예는 해당 연결에 대해 QM1을 조회한 결과를 보여줍니다.

```
C:\Documents and Settings\Admin>runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
display pubsub all
     2 : display pubsub all
AMQ8723: Display pub/sub status details.
   QMNAME(QM1)                           TYPE(LOCAL)
```

```
   STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
   QMNAME(QM2)                             TYPE(CHILD)
   STATUS(ACTIVE)
```

STATUS가 ACTIVE로 표시되지 않으면 하위와 상위 간의 채널이 올바르게 구성되고 실행 중인지 확인하십시오. 두 가지 큐 관리자 오류 로그에서 모두 가능한 오류를 확인하십시오.

다음에 수행할 작업

기본적으로, 단일 큐 관리자의 발행자 및 구독자가 사용하는 토픽은 계층 내 다른 큐 관리자의 발행자와 구독자가 공유합니다. 관리 토픽은 **SUBSCOPE** 및 **PUBSCOPE** 토픽 특성을 사용하여 공유 레벨을 제어하도록 구성될 수 있습니다. 분산 발행/구독 네트워크 구성을 참조하십시오.

관련 개념

발행 및 구독 범위 결합
IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

발행/구독 네트워크에서 토픽 공간 결합
발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

관련 태스크

발행/구독 클러스터 구성
큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

다른 큐 관리자로 클러스터 토픽 정의 이동
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가
토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

발행/구독 계층에서 큐 관리자 연결 끊기
발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

관련 정보

스트림 및 토픽
DISPLAY PUBSUB
발행/구독 메시징

발행/구독 계층에서 큐 관리자 연결 끊기

발행/구독 계층의 상위 큐 관리자에서 하위 큐 관리자 연결을 끊습니다.

이 태스크 정보

ALTER QMGR 명령을 사용하여 브로커 계층에서 큐 관리자의 연결을 끊을 수 있습니다. 언제든지 원하는 순서로 큐 관리자 연결을 끊을 수 있습니다.

상위를 업데이트하기 위한 해당 요청은 큐 관리자 사이의 연결이 실행 중일 때 송신됩니다.

프로시저

```
ALTER QMGR PARENT( '')
```

예

```
C:\Documents and Settings\Admin>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2.
    1 : alter qmgr parent('')
AMQ8005: IBM MQ queue manager changed.
    2 : display pubsub type(child)
AMQ8147: IBM MQ object  not found.
display pubsub type(parent)
    3 : display pubsub type(parent)
AMQ8147: IBM MQ object not found.
```

다음에 수행할 작업

더 이상 필요하지 않은 스트림, 큐, 수동으로 정의된 채널을 삭제할 수 있습니다.

관련 개념

발행 및 구독 범위 결합
IBM WebSphere MQ 7.0 이상에서는 발행과 구독 범위가 독립적으로 작용하여 큐 관리자 간의 발행 플로우를 판별합니다.

발행/구독 네트워크에서 토픽 공간 결합
발행/구독 클러스터 또는 계층에서 큐 관리자의 토픽 공간을 다른 큐 관리자와 결합합니다. 발행/구독 클러스터와, 계층이 있는 발행/구독 클러스터를 결합합니다.

관련 태스크

발행/구독 클러스터 구성
큐 관리자에서 토픽을 정의합니다. 이 토픽이 클러스터 토픽이 되게 하기 위해 **CLUSTER** 특성을 설정합니다. 이 토픽에 대해 발행물 및 구독에 사용할 라우팅을 선택하려면 **CLROUTE** 특성을 설정합니다.

다른 큐 관리자로 클러스터 토픽 정의 이동
토픽 호스트 라우트 클러스터나 직접 라우트 클러스터의 경우, 큐 관리자를 압축 해제할 때 또는 클러스터 큐 관리자가 실패했거나 오랫동안 사용할 수 없는 경우 클러스터 토픽 정의를 이동해야 합니다.

토픽 호스트 라우트 클러스터에 추가 토픽 호스트 추가
토픽 호스트 라우트 발행/구독 클러스터에서는 여러 큐 관리자를 사용하여 해당 큐 관리자에 동일한 클러스터 토픽 오브젝트를 정의함으로써 구독으로 발행을 라우트할 수 있습니다. 이는 가용성과 워크로드 밸런싱을 향상시키기 위해 사용할 수 있습니다. 동일한 클러스터 토픽 오브젝트에 대한 추가 토픽 호스트를 추가할 때 **PUB** 매개변수를 사용하여 새 토픽 호스트를 통해 발행물이 라우트되기 시작하는 시기를 제어할 수 있습니다.

큐 관리자를 발행/구독 계층에 연결
계층에서 하위 큐 관리자를 상위 큐 관리자에 연결합니다. 하위 큐 관리자가 이미 다른 계층 또는 클러스터의 멤버인 경우에는 이 연결이 계층을 함께 결합하거나 클러스터를 계층에 결합합니다.

ULW 다중 설치 구성

동일한 시스템에서 다중 설치를 사용하는 경우 설치 및 큐 관리자를 구성해야 합니다.

이 태스크 정보

이 정보는 UNIX, Linux, and Windows에 적용됩니다.

프로시저

- 다음 링크의 정보를 이용하여 설치를 구성하십시오.
  - 357 페이지의 『기본 설치 변경』
  - 359 페이지의 『큐 관리자를 설치와 연관』
  - 349 페이지의 『다중 설치 환경에서 애플리케이션 연결』

ULW 다중 설치 환경에서 애플리케이션 연결

UNIX, Linux, and Windows 시스템에서 IBM WebSphere MQ 7.1 이상의 라이브러리가 로드되면 추가 조치를 수행할 필요없이 IBM MQ가 자동으로 적절한 라이브러리를 사용합니다. IBM MQ는 애플리케이션이 연결되는 큐 관리자와 연관된 설치의 라이브러리를 사용합니다.

다음 개념은 애플리케이션이 IBM MQ에 연결되는 방법을 설명하는 데 사용됩니다.

링크
: 애플리케이션이 컴파일될 때 IBM MQ 라이브러리에 애플리케이션이 링크되어 애플리케이션이 실행될 때 로드되는 기능 내보내기를 가져옵니다.

로드 중
: 애플리케이션이 실행되면 IBM MQ 라이브러리를 찾아 로드됩니다. 라이브러리를 찾는 데 사용되는 특정 메커니즘은 운영 체제 및 애플리케이션 빌드 방법에 따라 다릅니다. 다중 설치 환경에서 라이브러리를 찾아 로드하는 방법에 대한 자세한 정보는 351 페이지의 『IBM MQ 라이브러리 로드』의 내용을 참조하십시오.

연결 중
: 예를 들어 MQCONN 또는 MQCONNX 호출을 사용하여 애플리케이션이 실행 중인 큐 관리자에 연결되면, 로드된 IBM MQ 라이브러리를 사용하여 연결됩니다.

서버 애플리케이션이 큐 관리자에 연결되면, 로드된 라이브러리는 큐 관리자와 연관된 설치의 라이브러리여야 합니다. 시스템에 여러 설치가 있는 경우, 이러한 제한사항으로 인해 운영 체제에서 로드할 IBM MQ 라이브러리를 찾기 위해 사용하는 메커니즘을 선택할 때 새로운 과제를 야기합니다.

- **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 변경하는 경우, 로드해야 하는 라이브러리가 변경됩니다.
- 애플리케이션이 다른 설치에 속하는 다중 큐 관리자에 연결되는 경우, 여러 라이브러리 세트를 로드해야 합니다.

그러나 IBM WebSphere MQ 7.1 이상의 라이브러리를 찾아 로드한 경우, IBM MQ는 추가 조치를 수행할 필요없이 해당 라이브러리를 로드하여 사용합니다. 애플리케이션이 큐 관리자에 연결되면, IBM MQ는 큐 관리자와 연관된 설치의 라이브러리를 로드합니다.

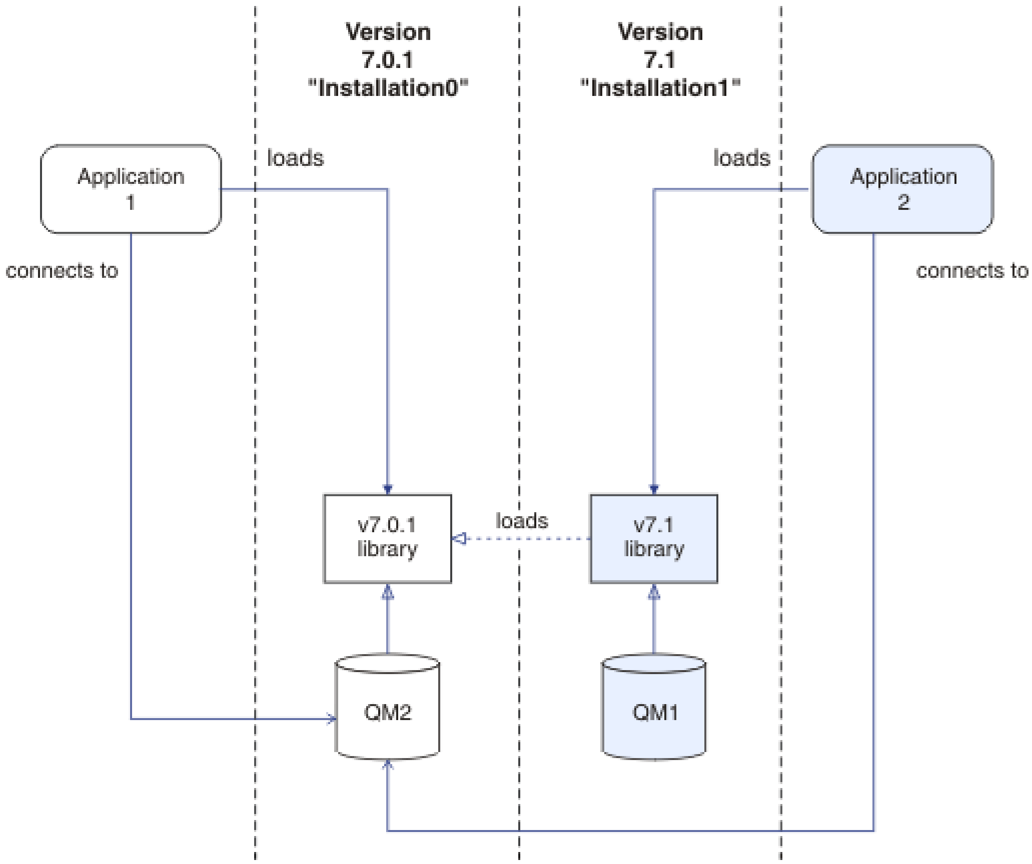


그림 69. 다중 설치 환경에서 애플리케이션 연결

예를 들어, 350 페이지의 그림 69 는 IBM WebSphere MQ 7.0.1 설치 (Installation0) 및 IBM WebSphere MQ 7.1 설치 (Installation1) 가 있는 다중 설치 환경을 표시합니다. 두 애플리케이션이 이 설치에 연결되어 있지만 서로 다른 라이브러리 버전을 로드합니다.

Application 1은 IBM WebSphere MQ 7.0.1 라이브러리를 직접 로드합니다. application 1이 QM2에 연결되면 IBM WebSphere MQ 7.0.1 라이브러리를 사용합니다. application 1이 QM1에 연결을 시도하거나 QM2가 Installation1과 연관된 경우 application 1은 2059(080B) (RC2059): MQRC_Q_MGR_NOT_AVAILABLE 오류로 실패합니다. IBM WebSphere MQ 7.0.1 라이브러리가 다른 라이브러리 버전을 로드할 수 없으므로 애플리케이션이 실패합니다. 즉, IBM WebSphere MQ 7.0.1 라이브러리가 직접 로드되면 상위 버전의 IBM MQ에서 설치와 연관된 큐 관리자를 사용할 수 없습니다.

Application 2은 IBM WebSphere MQ 7.1 라이브러리를 직접 로드합니다. application 2가 QM2에 연결되면 IBM WebSphere MQ 7.1 라이브러리가 IBM WebSphere MQ 7.0.1 라이브러리를 로드하고 사용합니다. application 2가 QM1에 연결되거나 QM2가 Installation1과 연관되면 IBM WebSphere MQ 7.1 라이브러리가 로드되고 애플리케이션이 예상대로 작동합니다.

다중 설치와 애플리케이션 연결 및 마이그레이션 시나리오는 UNIX, Linux, and Windows에서의 다중 설치 큐 관리자 공존에서 자세히 설명합니다.

IBM WebSphere MQ 7.1 라이브러리 로드 방법에 대한 자세한 정보는 351 페이지의 『IBM MQ 라이브러리 로드』의 내용을 참조하십시오.

지원 및 제한사항

다음 IBM WebSphere MQ 7.1 이상의 라이브러리가 있고 로드되는 경우 IBM MQ 는 자동으로 적절한 라이브러리를 로드하고 사용할 수 있습니다.

- C 서버 라이브러리

- C++ 서버 라이브러리
- XA 서버 라이브러리
- COBOL 서버 라이브러리
- COM+ 서버 라이브러리
- 비관리 모드의 .NET

IBM MQ는 또한 바인딩 모드에서 자동으로 Java 및 JMS 애플리케이션의 적절한 라이브러리를 로드하여 사용합니다.

다중 설치를 사용하는 애플리케이션에 대한 여러가지 제한사항이 있습니다. 자세한 정보는 354 페이지의 『다중 설치를 사용하는 애플리케이션에 대한 제한사항』의 내용을 참조하십시오.

관련 개념

354 페이지의 『다중 설치를 사용하는 애플리케이션에 대한 제한사항』
다중 설치 환경에서 CICS 서버 라이브러리, 빠른 경로 연결, 메시지 핸들, 엑시트를 사용할 경우 제한사항이 있습니다.

351 페이지의 『IBM MQ 라이브러리 로드』
IBM MQ 라이브러리를 로드하는 방법을 결정할 때는 사용자 환경, 기존 애플리케이션을 변경할 수 있는지 여부, 기본 설치를 원하는지 여부, IBM MQ 설치 위치, IBM MQ 위치가 변경되는지 여부를 포함하여 여러 요소를 고려해야 합니다.

관련 태스크

357 페이지의 『기본 설치 변경』
setmqinst 명령을 사용하여 설치를 기본 설치로 설정 또는 설정 해제할 수 있습니다.

359 페이지의 『큐 관리자를 설치와 연관』
큐 관리자를 작성하면 **crtmqm** 명령을 실행하는 설치와 자동으로 연관됩니다. UNIX, Linux, and Windows에서 **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 변경할 수 있습니다.

관련 정보

기본 설치 선택

ULW IBM MQ 라이브러리 로드

IBM MQ 라이브러리를 로드하는 방법을 결정할 때는 사용자 환경, 기존 애플리케이션을 변경할 수 있는지 여부, 기본 설치를 원하는지 여부, IBM MQ 설치 위치, IBM MQ 위치가 변경되는지 여부를 포함하여 여러 요소를 고려해야 합니다.

이 정보는 IBM WebSphere MQ 7.1 또는 그 이후 버전의 라이브러리에만 적용됩니다.

IBM MQ 라이브러리를 찾아 로드하는 방법은 사용자 설치 환경에 따라 다릅니다.

- UNIX and Linux 시스템의 경우, IBM WebSphere MQ 7.1 이상 버전의 사본이 기본 위치에 설치된 경우에는 기존 애플리케이션이 이전 버전과 동일한 방식으로 계속 작동합니다. 그러나 애플리케이션에 `/usr/lib`의 기호 링크가 필요한 경우에는 IBM WebSphere MQ 7.1또는 이후 버전을 선택하거나 설치를 기본 설치로 설정하거나 기호 링크를 수동으로 작성해야 합니다.
- IBM WebSphere MQ 7.1 이상 버전이 기본 이외의 위치에 설치된 경우(이 경우 IBM WebSphere MQ 7.0.1도 설치됨) 올바른 라이브러리가 로드되도록 기존 애플리케이션을 변경해야 합니다.

How IBM MQ 라이브러리를 찾아 로드하는 방법은 또한 라이브러리 로드하기 위한 기존 애플리케이션의 설정 방법에 따라서도 달라집니다. 라이브러리 로드 방법에 대한 자세한 정보는 353 페이지의 『운영 체제 라이브러리 로딩 메커니즘』의 내용을 참조하십시오.

가장 좋은 방법은 운영 체제로 로드되는 IBM MQ 라이브러리가 큐 관리자와 연관된 라이브러리인지 확인하는 것입니다.

IBM MQ 라이브러리를 로드하는 방법은 플랫폼마다 다르며 각각의 방법마다 장단점이 있습니다.

표 26. 라이브러리 로드 옵션의 장단점

플랫폼	옵션	장점	문제점
Linux UNIX UNIX and Linux 시스템	애플리케이션의 임베드된 런타임 검색 경로(RPath)를 설정하거나 변경합니다. 이 옵션에서는 사용자가 애플리케이션을 다시 컴파일하고 링크해야 합니다. 애플리케이션 컴파일 및 링크에 대한 자세한 정보는 프로시저 애플리케이션 빌드를 참조하십시오.	• 변경 범위가 명확합니다.	• 애플리케이션을 다시 컴파일하고 링크할 수 있어야 합니다. • IBM MQ 위치가 변경되면 RPath를 변경해야 합니다.
UNIX and Linux 시스템	setmqenv또는 crtmqenv를 사용하여 *LD_LIBRARY_PATH* 환경 변수를 **-k** 또는 **-l** 옵션과 함께 설정하십시오. AIX AIX에서 이 환경 변수는 *LIBPATH*입니다.	• 기존 애플리케이션을 변경할 필요가 없습니다. • 애플리케이션에서 임베드된 RPath를 대체합니다. • IBM MQ의 위치가 변경되면 변수를 변경하기가 쉽습니다.	• `setuid` 및 `setgid` 애플리케이션 또는 기타 방식으로 빌드된 애플리케이션에서는 보안 상의 이유로 *LD_LIBRARY_PATH*를 무시합니다. • 환경에 따라 다르므로 애플리케이션이 실행되는 각 환경에서 설정되어야 합니다. • *LD_LIBRARY_PATH*에 의존하는 기타 애플리케이션에 영향을 줄 수 있습니다. • HP-UX HP-UX: 애플리케이션이 컴파일되었을 때 사용한 옵션이 *LD_LIBRARY_PATH*를 사용 불가능으로 설정했을 수 있습니다. 자세한 정보는 HP-UX의 런타임 링크 고려사항을 참조하십시오. • Linux Linux: 애플리케이션을 빌드할 때 사용한 컴파일러가 *LD_LIBRARY_PATH*를 사용 불가능으로 설정했을 수 있습니다. 자세한 정보는 Linux의 런타임 링크 고려사항을 참조하십시오.
Windows Windows 시스템	setmqenv 또는 crtmqenv를 사용하여 PATH 변수를 설정합니다.	• 기존 애플리케이션을 변경할 필요가 없습니다. • IBM MQ의 위치가 변경되면 변수를 변경하기가 쉽습니다.	• 환경에 따라 다르므로 애플리케이션이 실행되는 각 환경에서 설정되어야 합니다. • 기타 애플리케이션에 영향을 줄 수 있습니다.

표 26. 라이브러리 로드 옵션의 장단점 (계속)			
플랫폼	옵션	장점	문제점
ULW UNIX, Linux, and Windows 시스템	기본 설치를 IBM WebSphere MQ 7.1 이상 설치로 설정합니다. 357 페이지의 『기본 설치 변경』의 내용을 참조하십시오. 기본 설치에 대한 자세한 정보는 기본 설치 선택을 참조하십시오.	• 기존 애플리케이션을 변경할 필요가 없습니다. • IBM MQ의 위치가 변경되면 기본 설치를 변경하기가 쉽습니다. • 이전 버전의 IBM MQ에서와 유사하게 작동합니다.	• IBM WebSphere MQ 7.0.1이 설치된 경우 기본 설치를 IBM WebSphere MQ 7.1 이상으로 설정할 수 없습니다. • Linux UNIX UNIX and Linux: /usr/lib가 기본 검색 경로에 없으면 작동하지 않습니다.

HP-UX의 라이브러리 로딩 고려사항

HP-UX

이전 버전의 IBM MQ 에 대한 제품 문서의 샘플 컴파일 명령에는 64비트애플리케이션에 대한 `-W1, +noenvvar` 링크 옵션이 포함되어 있습니다. 이 옵션은 공유 라이브러리를 로드하기 위한 *LD_LIBRARY_PATH* 사용을 불가능하게 합니다. 애플리케이션이 RPath에 지정된 위치 이외의 위치에서 IBM MQ 라이브러리를 로드하도록 하려면 애플리케이션을 업데이트해야 합니다. `-W1, +noenvvar` 링크 옵션 없이 다시 컴파일하고 링크함으로써 또는 **chatr** 명령을 사용하여 애플리케이션을 업데이트할 수 있습니다.

애플리케이션이 현재 라이브러리를 로드하는 방법을 확인하려면 353 페이지의 『운영 체제 라이브러리 로딩 메커니즘』의 내용을 참조하십시오.

Linux의 라이브러리 로딩 고려사항

Linux

gcc의 일부 버전(예: 버전 3.2.x)을 사용하여 컴파일된 애플리케이션에는 *LD_LIBRARY_PATH* 환경 변수를 사용하여 대체할 수 없는 임베드된 RPath가 있을 수 있습니다. `readelf -d` *applicationName* 명령을 사용하여 애플리케이션이 영향을 받는지 여부를 판별할 수 있습니다. RPATH 기호가 있고 RUNPATH 기호가 없는 경우에는 RPath를 대체할 수 없습니다.

Solaris의 라이브러리 로딩 고려사항

Solaris

이전 버전의 IBM MQ 에 대한 제품 문서의 샘플 컴파일 명령에는 `-lmqmcs -lmqmzse` 링크 옵션이 포함되어 있습니다. 이제 IBM MQ가 이러한 라이브러리의 적절한 버전을 자동으로 로드합니다. IBM MQ가 기본이 아닌 위치에 설치되거나 시스템에 다중 설치가 있는 경우, 애플리케이션을 업데이트해야 합니다. `-lmqmcs -lmqmzse` 링크 옵션 없이 다시 컴파일하고 링크하여 애플리케이션을 업데이트할 수 있습니다.

운영 체제 라이브러리 로딩 메커니즘

Windows 시스템에서는 다음과 같은 여러 디렉토리에서 라이브러리를 검색합니다.

- 애플리케이션이 로드되는 디렉토리
- 현재 디렉토리
- *PATH* 환경 변수(글로벌 *PATH* 변수 및 현재 사용자의 *PATH* 변수 둘 다)의 디렉토리

Linux UNIX UNIX and Linux 시스템에서는 다음과 같은 다양한 방법을 사용하여 라이브러리를 찾을 수 있습니다.

- Using the *LD_LIBRARY_PATH* environment variable (also *LIBPATH* on AIX및 HP-UX 의 *SHLIB_PATH*). 이 변수가 설정되면 필요한 IBM MQ 라이브러리를 검색하는 디렉토리 세트를 정의합니다. 이 디렉토리에서 라이

브러리를 찾으면 다른 방법을 사용하여 찾을 수 있는 라이브러리의 환경 설정에서 해당 라이브러리를 사용합니다.

- 임베드된 검색 경로(RPath) 사용. IBM MQ 라이브러리를 검색하는 디렉토리 세트가 애플리케이션에 포함될 수 있습니다. *LD_LIBRARY_PATH*가 설정되지 않거나 변수를 사용하여 필요한 라이브러리를 찾지 못한 경우에는 RPath에서 라이브러리를 검색합니다. 기존 애플리케이션에서 RPath를 사용하지만 애플리케이션을 다시 컴파일하고 링크할 수 없는 경우에는 기본 위치에 IBM WebSphere MQ 7.1을 설치하거나 다른 방법을 사용하여 라이브러리를 찾아야 합니다.
- 기본 라이브러리 경로 사용. *LD_LIBRARY_PATH* 변수와 RPath 위치를 검색한 후 IBM MQ 라이브러리를 찾지 못하면 기본 라이브러리 경로를 검색합니다. 일반적으로 이 경로에는 `/usr/lib` 또는 `/usr/lib64`가 포함됩니다. 기본 라이브러리 경로를 검색한 후 라이브러리를 찾지 못하면 종속 항목 누락으로 인해 애플리케이션이 시작되지 않습니다.

운영 체제 메커니즘를 사용하여 애플리케이션에 임베드된 검색 경로가 있는지 여부를 확인할 수 있습니다. 예를 들면, 다음과 같습니다.

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

관련 개념

354 페이지의 『다중 설치를 사용하는 애플리케이션에 대한 제한사항』
다중 설치 환경에서 CICS 서버 라이브러리, 빠른 경로 연결, 메시지 핸들, 엑시트를 사용할 경우 제한사항이 있습니다.

349 페이지의 『다중 설치 환경에서 애플리케이션 연결』
UNIX, Linux, and Windows 시스템에서 IBM WebSphere MQ 7.1 이상의 라이브러리가 로드되면 추가 조치를 수행할 필요없이 IBM MQ가 자동으로 적절한 라이브러리를 사용합니다. IBM MQ는 애플리케이션이 연결되는 큐 관리자와 연관된 설치의 라이브러리를 사용합니다.

관련 태스크

357 페이지의 『기본 설치 변경』
setmqinst 명령을 사용하여 설치를 기본 설치로 설정 또는 설정 해제할 수 있습니다.

359 페이지의 『큐 관리자를 설치와 연관』
큐 관리자를 작성하면 **crtmqm** 명령을 실행하는 설치와 자동으로 연관됩니다. UNIX, Linux, and Windows에서 **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 변경할 수 있습니다.

관련 정보

기본 설치 선택

ULW 다중 설치를 사용하는 애플리케이션에 대한 제한사항

다중 설치 환경에서 CICS 서버 라이브러리, 빠른 경로 연결, 메시지 핸들, 엑시트를 사용할 경우 제한사항이 있습니다.

CICS 서버 라이브러리

CICS 서버 라이브러리를 사용하는 경우 IBM MQ는 올바른 라이브러리 레벨을 자동으로 선택하지 않습니다. 애플리케이션이 연결되는 큐 관리자에 적합한 라이브러리 레벨로 애플리케이션을 컴파일 및 링크해야 합니다. 자세한 정보는 멀티플랫폼용 TXSeries® 버전 5와 함께 사용할 라이브러리 빌드의 내용을 참조하십시오.

메시지 핸들

특수값 `MQHC_UNASSOCIATED_HCONN`을 사용하는 메시지 핸들은 프로세스에 로드되는 첫 번째 설치에 사용하도록 제한됩니다. 특정 설치에서 메시지 핸들을 사용할 수 없는 경우 이유 코드 `MQRC_HMSG_NOT_AVAILABLE`이 리턴됩니다.

이 제한은 메시지 특성에 영향을 미칩니다. 메시지 핸들을 사용하면 한 설치의 큐 관리자에서 메시지 특성을 가져와 다른 설치의 큐 관리자로 삽입할 수 없습니다. 메시지 핸들에 대한 자세한 정보는 MQCRTMH - 메시지 핸들 작성을 참조하십시오.

엑시트

다중 설치 환경에서, IBM WebSphere MQ 7.1 이상 설치와 함께 사용하려면 기존 엑시트를 업데이트해야 합니다. **crtmqcvx** 명령을 사용하여 생성된 데이터 변환 엑시트는 업데이트된 명령을 사용하여 다시 생성되어야 합니다.

모든 엑시트는 MQIEP 구조를 사용하여 작성되어야 하며, 임베드된 RPATH를 사용하여 IBM MQ 라이브러리를 찾을 수 없고, IBM MQ 라이브러리에 링크할 수도 없습니다. 자세한 정보는 UNIX, Linux, and Windows에서 엑시트 및 설치 가능 서비스 작성을 참조하십시오.

바로 가기

다중 설치가 있는 서버에서, IBM WebSphere MQ 7.1 이상에 대한 빠른 경로 연결을 사용하는 애플리케이션은 다음 규칙을 준수해야 합니다.

1. 큐 관리자는 애플리케이션이 IBM MQ 런타임 라이브러리를 로드할 때 사용한 것과 동일한 설치와 연관되어야 합니다. 애플리케이션은 다른 설치와 연관된 큐 관리자에 대해 빠른 경로 연결을 사용할 수 없습니다. 연결을 시도하면 오류가 발생하고, 이유 코드는 MQRC_INSTALLATION_MISMATCH입니다.
2. 애플리케이션이 IBM MQ 런타임 라이브러리를 로드할 때 사용한 것과 동일한 설치와 연관된 큐 관리자에 빠르지 않은 경로를 연결하면 애플리케이션에서 빠른 경로 연결을 사용할 수 없게 됩니다. 단, 다음 조건 중 하나에 해당하는 경우는 예외입니다.
   - 애플리케이션이 빠른 경로 연결과 동일한 설치와 연관된 큐 관리자에 처음 연결됩니다.
   - 환경 변수 AMQ_SINGLE_INSTALLATION이 설정되어 있습니다.
3. IBM WebSphere MQ 7.1 이상의 설치와 연관된 큐 관리자로의 빠르지 않은 경로 연결은 애플리케이션이 빠른 경로에 연결할 수 있는지 여부에 영향을 미치지 않습니다.
4. IBM WebSphere MQ 7.0.1 설치와 연관된 큐 관리자에 대한 연결을 결합하고 빠른 경로를 IBM WebSphere MQ 7.1또는 이후 설치와 연관된 큐 관리자에 연결할 수 없습니다.

AMQ_SINGLE_INSTALLATION이 설정된 경우 큐 관리자 연결을 빠른 경로 연결로 전환할 수 있습니다. 그렇지 않은 경우 다음과 거의 동일한 제한사항이 적용됩니다.

- 설치는 IBM MQ 런타임 라이브러리가 로드된 설치와 동일해야 합니다.
- 동일한 프로세스에서의 모든 연결은 동일한 설치로의 연결이어야 합니다. 다른 설치와 연관된 큐 관리자에 연결하려고 하면 MQRC_INSTALLATION_MISMATCH 이유 코드와 함께 연결이 실패합니다.
  AMQ_SINGLE_INSTALLATION이 설정된 경우 이 제한사항은 빠른 경로 연결뿐 아니라 모든 연결에 적용됩니다.
- 빠른 경로 연결과 하나의 큐 관리자만 연결하십시오.

관련 정보

MQCONNX - 연결 큐 관리자(확장)
MQIEP 구조
2583 (0A17) (RC2583): MQRC_INSTALLATION_MISMATCH
2587 (0A1B) (RC2587): MQRC_HMSG_NOT_AVAILABLE
2590 (0A1E) (RC2590): MQRC_FASTPATH_NOT_AVAILABLE

ULW 다중 설치 환경에서 .NET 애플리케이션 연결

기본적으로 애플리케이션은 기본 설치의 .NET 어셈블리를 사용합니다. 기본 설치가 없거나 기본 설치 어셈블리를 사용하지 않으려면 애플리케이션 구성 파일 또는 *DEVPATH* 환경 변수를 업데이트해야 합니다.

시스템에 기본 설치가 있으면 해당 설치의 .NET 어셈블리와 정책 파일이 글로벌 어셈블리 캐시(GAC)에 등록됩니다. 다른 모든 설치의 .NET 어셈블리는 각 설치의 설치 경로에서 찾을 수 있지만 어셈블리는 GAC에 등록되지

않습니다. 따라서 애플리케이션은 기본적으로 기본 설치의 .NET 어셈블리를 사용하여 실행됩니다. 다음과 같은 경우 애플리케이션 구성 파일을 업데이트해야 합니다.

- 기본 설치가 없습니다.
- 애플리케이션에서 기본 설치 어셈블리를 사용하지 않습니다.
- 기본 설치가 애플리케이션이 컴파일된 버전보다 낮은 버전의 IBM MQ입니다.

애플리케이션 구성 파일을 업데이트하는 방법에 대한 정보는 356 페이지의 『애플리케이션 구성 파일을 사용하여 .NET 애플리케이션 연결』의 내용을 참조하십시오.

다음과 같은 경우 *DEVPATH* 환경 변수를 업데이트해야 합니다.

- 애플리케이션에서 비기본 설치의 어셈블리를 사용하려고 하지만 기본 설치가 비기본 설치와 동일한 버전입니다.

DEVPATH 변수 업데이트 방법에 대한 자세한 정보는 357 페이지의 『DEVPATH를 사용하여 .NET 애플리케이션 연결』의 내용을 참조하십시오.

애플리케이션 구성 파일을 사용하여 .NET 애플리케이션 연결

애플리케이션 구성 파일에서, 여러 태그를 설정하여 기본 설치가 아닌 어셈블리를 사용하도록 애플리케이션 경로를 재지정해야 합니다.

다음 표는 .NET 애플리케이션이 특정 어셈블리를 사용하여 연결될 수 있도록 애플리케이션 구성 파일에 작성될 필요가 있는 특정 변경사항을 보여줍니다.

표 *27.* 특정 어셈블리를 사용하도록 애플리케이션 구성

	하위 버전의 IBM MQ와 컴파일된 애플리케이션	**상위 버전의 IBM MQ와 컴파일된 애플리케이션**
상위 버전의 IBM MQ 기본 설치로 애플리케이션을 실행합니다. (GAC에 있는 상위 버전 어셈블리):	변경이 필요하지 않음	변경이 필요하지 않음
하위 버전의 IBM MQ 기본 설치로 애플리케이션을 실행합니다. (GAC에 있는 하위 버전 어셈블리):	변경이 필요하지 않음	애플리케이션 구성 파일에서: • GAC에 있는 하위 버전의 어셈블리 사용을 표시하려면 *bindingRedirect* 태그를 사용하십시오.
상위 버전의 IBM MQ 비기본 설치로 애플리케이션을 실행합니다. (설치 폴더에 있는 상위 버전 어셈블리):	애플리케이션 구성 파일에서: • 상위 버전 어셈블리의 위치를 가리키려면 *codebase* 태그를 사용하십시오. • 상위 버전 어셈블리 사용을 표시하려면 *bindingRedirect* 태그를 사용하십시오.	애플리케이션 구성 파일에서: • 상위 버전 어셈블리의 위치를 가리키려면 *codebase* 태그를 사용하십시오.
IBM MQ 비기본 설치의 하위 버전과 함께 애플리케이션을 실행하려면 다음을 수행하십시오 (설치 폴더에 있는 하위 버전 어셈블리).	애플리케이션 구성 파일에서: • 하위 버전 어셈블리의 위치를 가리키려면 *codebase* 태그를 사용하십시오. • *publisherpolicy Apply=no* 태그를 포함하십시오.	애플리케이션 구성 파일에서: • 하위 버전 어셈블리의 위치를 가리키려면 *codebase* 태그를 사용하십시오. • 하위 버전 어셈블리 사용을 표시하려면 *bindingRedirect* 태그를 사용하십시오. • *publisherpolicy Apply=no* 태그를 포함하십시오.

샘플 애플리케이션 구성 파일 `NonPrimaryRedirect.config`가 *MQ_INSTALLATION_PATH*`\tools\dotnet\samples\base` 폴더에 제공됩니다. 이 파일은 비기본 설치의 IBM MQ 설치 경로로 수정될 수 있습니다. 파일은 *linkedConfiguration* 태그를 사용하여 기타 구성 파일에 직접 포함될 수도 있습니다. 샘플은 `nmqsget.exe.config` 및 `nmqsput.exe.config`에 대해 제공됩니다. 두 샘플 모두 *linkedConfiguration* 태그를 사용하고 `NonPrimaryRedirect.config` 파일을 포함합니다.

DEVPATH를 사용하여 .NET 애플리케이션 연결

DEVPATH 환경 변수를 사용하여 어셈블리를 찾을 수 있습니다. *DEVPATH* 변수로 지정하는 어셈블리는 GAC에 있는 어셈블리에 대한 참조에서 사용됩니다. 이 변수 사용 시점에 대한 자세한 정보는 *DEVPATH*에 대한 해당 Microsoft 문서를 참조하십시오.

DEVPATH 환경 변수를 사용하여 어셈블리를 찾으려면 사용하려는 어셈블리가 있는 폴더로 *DEVPATH* 변수를 설정해야 합니다. 그런 다음 애플리케이션 구성 파일을 업데이트하고 다음 런타임 구성 정보를 추가해야 합니다.

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

관련 개념

349 페이지의 『다중 설치 환경에서 애플리케이션 연결』
UNIX, Linux, and Windows 시스템에서 IBM WebSphere MQ 7.1 이상의 라이브러리가 로드되면 추가 조치를 수행할 필요없이 IBM MQ가 자동으로 적절한 라이브러리를 사용합니다. IBM MQ는 애플리케이션이 연결되는 큐 관리자와 연관된 설치의 라이브러리를 사용합니다.

관련 정보

기본 설치 선택

.NET

다중 설치

ULW 기본 설치 변경

`setmqinst` 명령을 사용하여 설치를 기본 설치로 설정 또는 설정 해제할 수 있습니다.

이 태스크 정보

이 태스크는 UNIX, Linux, and Windows에 적용됩니다.

기본 설치는 요청되는 시스템 전체 위치에서 참조하는 설치입니다. 기본 설치에 대한 자세한 정보와 기본 설치 선택을 위한 고려사항은 기본 설치 선택을 참조하십시오.

IBM WebSphere MQ 7.1 이상 설치가 IBM WebSphere MQ 7.0.1 설치와 공존하는 경우에는 IBM WebSphere MQ 7.0.1 설치가 기본 설치여야 합니다. IBM WebSphere MQ 7.1 이상 버전이 설치되면 기본으로 플래그가 지정되고 IBM WebSphere MQ 7.1 이상 설치는 기본이 될 수 없습니다.

Windows Windows에서 설치 프로세스를 수행하는 동안 이 설치가 기본 설치가 되도록 지정할 수 있습니다.

Linux UNIX UNIX and Linux 시스템에서 설치를 기본 설치로 설정한 후 **`setmqinst`** 명령을 실행해야 합니다.

357 페이지의 『기본 설치 설정』.

358 페이지의 『기본 설치 설정 해제』.

기본 설치 설정

프로시저

설치를 기본 설치로 설정하려면 다음을 수행하십시오.

1. 다음 명령을 입력하여 설치가 기본 설치인지 확인하십시오.

   ```
   MQ_INSTALLATION_PATH/bin/dspmqinst
   ```

   여기서 *MQ_INSTALLATION_PATH*는 IBM WebSphere MQ 7.1 이상 설치의 설치 경로입니다.
2. 기존 IBM WebSphere MQ 7.1 이상의 설치가 기본 설치로 설정되어 있는 경우 358 페이지의 『기본 설치 설정 해제』의 지시사항에 따라 설정을 해제하십시오. 시스템에 IBM WebSphere MQ 7.0.1이 설치되어 있으면 기본 설치를 변경할 수 없습니다.
3. 적절한 권한을 사용하여 로그온했는지 확인하십시오.
   - UNIX and Linux의 경우 루트로
   - Windows 시스템의 경우 관리자 그룹의 구성원으로
4. 다음 명령 중 하나를 입력하십시오.
   - 기본 설치로 사용할 설치의 경로를 사용하여 기본 설치를 설정하려면 다음 명령을 실행하십시오.

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

   - 기본 설치로 사용할 설치의 이름을 사용하여 기본 설치를 설정하려면 다음 명령을 실행하십시오.

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

5. Windows 시스템에서 시스템을 재시작하십시오.

기본 설치 설정 해제

프로시저

설치를 기본 설치로 설정 해제하려면 다음을 수행하십시오.

1. 다음 명령을 입력하여 설치가 기본 설치인지 확인하십시오.

   ```
   MQ_INSTALLATION_PATH/bin/dspmqinst
   ```

   여기서 *MQ_INSTALLATION_PATH*는 IBM WebSphere MQ 7.1 이상 설치의 설치 경로입니다.

   IBM WebSphere MQ 7.0.1이 기본 설치이면 기본 설치를 설정 해제할 수 없습니다.
2. 적절한 권한을 사용하여 로그온했는지 확인하십시오.
   - UNIX and Linux의 경우 루트로
   - Windows 시스템의 경우 관리자 그룹의 구성원으로
3. 다음 명령 중 하나를 입력하십시오.
   - 더 이상 기본 설치로 사용하지 않을 설치의 경로를 사용하여 기본 설치를 설정 해제하려면 다음 명령을 실행하십시오.

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

   - 더 이상 기본 설치로 사용하지 않을 설치의 이름을 사용하여 기본 설치를 설정 해제하려면 다음 명령을 실행하십시오.

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

관련 정보
Windows에서 기본 설치에서만 사용할 수 있는 기능
UNIX and Linux에서 기본 설치에 대한 외부 라이브러리 및 제어 명령 링크
기본 설치의 설치 제거, 업그레이드 및 유지보수
설치 이름 선택
setmqinst

ULW 큐 관리자를 설치와 연관

큐 관리자를 작성하면 **crtmqm** 명령을 실행하는 설치와 자동으로 연관됩니다. UNIX, Linux, and Windows에서 **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 변경할 수 있습니다.

이 태스크 정보

해당 설치에서의 명령만으로 관리될 수 있도록 큐 관리자가 연관된 설치가 해당 큐 관리자를 제한합니다. 다음과 같은 세 가지 주요 예외 사항이 있습니다.

- **setmqm**은 큐 관리자와 연관된 설치를 변경합니다. 이 명령은 큐 관리자가 현재 연관된 설치가 아니라 큐 관리자와 연관시키려는 설치에서 실행되어야 합니다. **setmqm** 명령을 통해 지정된 설치 이름은 명령이 실행된 설치와 일치해야 합니다.
- 일반적으로 **strmqm**은 큐 관리자와 연관된 설치에서 실행되어야 합니다. 그러나 IBM WebSphere MQ 7.0.1 또는 이전 버전의 큐 관리자가 IBM WebSphere MQ 7.1 이상의 설치에서 처음으로 시작될 때 **strmqm**을 사용할 수 있습니다. 이 경우 **strmqm**이 큐 관리자를 시작하여 명령이 실행된 설치와 연관시킵니다.
- **dspmq**는 **dspmq** 명령과 동일한 설치와 연관된 큐 관리자만이 아니라 시스템에 있는 모든 큐 관리자에 대한 정보를 표시합니다. `dspmq -o installation` 명령은 설치와 연관된 큐 관리자에 대한 정보를 표시합니다.

HA 환경의 경우 **addmqinf** 명령이 자동으로 큐 관리자를 **addmqinf** 명령이 실행된 설치와 연관시킵니다. **strmqm** 명령이 **addmqinf** 명령과 동일한 설치에서 실행되는 동안 추가적인 설정이 필요하지 않습니다. 다른 설치를 사용하여 큐 관리자를 시작하려면 먼저 **setmqm** 명령을 사용하여 연관된 설치를 변경해야 합니다.

큐 관리자를 설치와 연관시키려는 경우 **setmqm** 명령을 다음 방법으로 사용할 수 있습니다.

- IBM MQ의 동일 버전 간 개별 큐 관리자 이동. 예를 들어, 테스트 시스템에서 프로덕션 시스템으로 큐 관리자를 이동합니다.
- 이전 버전의 IBM MQ에서 최신 버전의 IBM MQ로 개별 큐 관리자 마이그레이션. 버전 간에 큐 관리자를 마이그레이션할 때 여러 가지 함축된 내용을 알고 있어야 합니다. 마이그레이션에 대한 자세한 정보는 유지보수 및 마이그레이션을 참조하십시오.

프로시저

1. 현재 큐 관리자와 연관된 설치에서 **endmqm** 명령을 사용하여 큐 관리자를 중지하십시오.
2. 설치에서 **setmqm** 명령을 사용하여 큐 관리자를 다른 설치와 연관시키십시오.

   예를 들어 이름이 `Installation2`인 설치와 연관되도록 큐 관리자 `QMB`를 설정하려면 Installation2에서 다음 명령을 입력하십시오.

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

   여기서, *MQ_INSTALLATION_PATH*는 Installation2가 설치된 경로입니다.
3. 현재 큐 관리자와 연관된 설치에서 **strmqm** 명령을 사용하여 큐 관리자를 시작하십시오.

   이 명령은 필요한 큐 관리자 마이그레이션을 수행하며 큐 관리자를 사용 준비 상태로 만듭니다.

다음에 수행할 작업

큐 관리자가 연관된 설치가 삭제되었거나 큐 관리자 상태 정보를 사용할 수 없는 경우 큐 관리자를 다른 설치와 연관시키기 위한 **setmqm** 명령이 실패합니다. 이 경우 다음 조치를 수행하십시오.

1. 시스템에 있는 다른 설치를 보려면 **dspmqinst** 명령을 사용하십시오.
2. 다른 설치를 지정하려면 mqs.ini에 있는 `QueueManager` 스탠자의 `InstallationName` 필드를 수동으로 수정하십시오.
3. 큐 관리자를 삭제하려면 해당 설치에서 **dltmqm** 명령을 사용하십시오.

관련 개념

360 페이지의 『시스템에서 IBM MQ 설치 찾기』
시스템에 다중 IBM MQ 설치가 있는 경우 설치된 버전과 설치 위치를 확인할 수 있습니다.

81 페이지의 『IBM MQ 구성 파일 mqs.ini』
IBM MQ 구성 파일 `mqs.ini`에는 노드의 모든 큐 관리자와 관련된 정보가 있습니다. 이 파일은 설치 중 자동으로 작성됩니다.

관련 정보

기본 설치 선택
addmqinf
dspmq
dspmqinst
endmqm
setmqm
strmqm

ULW 시스템에서 IBM MQ 설치 찾기

시스템에 다중 IBM MQ 설치가 있는 경우 설치된 버전과 설치 위치를 확인할 수 있습니다.

다음 방법을 사용하여 시스템에서 IBM MQ 설치를 찾을 수 있습니다.

- **dspmqver** 명령을 사용합니다. 이 명령은 IBM WebSphere MQ 7.0.1 설치에서 실행된 경우 시스템의 모든 설치에 대한 자세한 내용을 제공하지 않습니다.
- 플랫폼 설치 도구를 사용하여 IBM MQ가 설치된 위치를 조회합니다. 그런 다음, IBM WebSphere MQ 7.1 이상 설치에서 **dspmqver** 명령을 사용합니다. 다음 명령은 IBM MQ가 설치된 위치를 조회하는 데 사용할 수 있는 명령 예제입니다.
  - AIX 시스템의 경우 다음과 같이 **lslpp** 명령을 사용할 수 있습니다.

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

  - HP-UX 시스템의 경우 다음과 같이 **swlist** 명령을 사용할 수 있습니다.

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

  - Linux 시스템의 경우 다음과 같이 **rpm** 명령을 사용할 수 있습니다.

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

  - Solaris 시스템의 경우 다음과 같이 **pkginfo** 및 **pkgparam** 명령을 사용할 수 있습니다.
    1. 다음 명령을 입력하여 설치된 패키지를 나열하십시오.

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

    2. 나열된 각 패키지에 대해 다음 명령을 입력하십시오.

       ```
       pkgparam pkgname BASEDIR
       ```

- Windows 시스템의 경우 다음과 같이 **wmic** 명령을 사용할 수 있습니다. 이 명령은 wmic 클라이언트를 설치할 수 있습니다.

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

- UNIX and Linux 시스템의 경우 다음 명령을 실행하여 IBM MQ가 설치된 위치를 확인하십시오.

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

  그런 다음, IBM WebSphere MQ 7.1 이상 설치에서 **dspmqver** 명령을 사용합니다.
- 32비트 Windows에서 시스템의 설치 세부사항을 표시하려면 다음 명령을 실행하십시오.

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

- 64비트 Windows의 경우 다음 명령을 실행하십시오.

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

참고: **reg.exe** 명령은 IBM WebSphere MQ 7.1 이상 설치에 대한 정보만 표시합니다.

관련 정보
dspmqver
dspmqinst
다중 설치

고가용성, 복구 및 재시작 구성

큐 관리자가 실패한 경우 큐 가용성을 유지보수하고 서버 또는 스토리지가 실패한 후에 메시지를 복구하여 애플리케이션의 가용성을 높일 수 있습니다.

이 태스크 정보

z/OS z/OS에서 고가용성이 플랫폼에 빌드되었습니다. 또한 큐 공유 그룹을 사용하여 서버 애플리케이션 가용성을 향상시킬 수 있습니다. 공유 큐 및 큐 공유 그룹을 참조하십시오.

Multi 멀티플랫폼에서 큐 관리자가 실패한 후 클라이언트를 자동으로 큐 관리자 그룹 간에 전환하거나 다중 인스턴스 큐 관리자의 새 활성 인스턴스로 전환하기 위해 클라이언트 다시 연결을 사용하여 클라이언트 애플리케이션의 가용성을 높일 수 있습니다. 자동 클라이언트 다시 연결은 IBM MQ classes for Java에서 지원되지 않습니다. 다중 인스턴스 큐 관리자가 여러 서버에서 단일 큐 관리자로 실행되도록 구성됩니다. 이 큐 관리자에 서버 애플리케이션을 배치하십시오. 활성 인스턴스를 실행하는 서버가 실패한 경우 다른 서버에 있는 같은 큐 관리자의 대기 인스턴스로 실행이 자동 전환됩니다. 서버 애플리케이션이 큐 관리자 서비스로 실행되도록 구성한 경우 대기 인스턴스가 활발히 실행하는 큐 관리자 인스턴스가 될 때 이 서버 애플리케이션은 재시작됩니다.

다중 플랫폼에서 서버 애플리케이션 가용성을 높이는 다른 방법은 서버 애플리케이션을 큐 관리자 클러스터에 있는 여러 컴퓨터에 배치하는 것입니다. IBM WebSphere MQ 7.1부터, 클러스터 오류 복구에서는 문제점이 해결될 때까지 문제점의 원인이 되는 조작을 재실행합니다. z/OS 외의 서버에서 클러스터 오류 복구 변경을 참조하십시오. 또한 IBM MQ for Multiplatforms를 다음과 같은 플랫폼별 클러스터링 솔루션의 일부로 구성할 수 있습니다.

- Microsoft 클러스터 서버
- IBM i IBM i의 HA 클러스터
- Linux UNIX PowerHA® for AIX(이전의 HACMP on AIX) 및 기타 UNIX and Linux 클러스터링 솔루션

메시징 시스템은 시스템에 입력된 메시지가 목적지에 전달되도록 합니다. IBM MQ는 **dspmqrte** 명령을 사용하여 한 큐 관리자에서 다른 큐 관리자로 메시지가 이동할 때 메시지의 라우트를 추적할 수 있습니다. 시스템이 실패하면 메시지는 실패 유형 및 시스템이 구성되는 방식에 따라 다양한 방식으로 복구될 수 있습니다. IBM MQ는 메시지의 수신, 전송 및 전달을 처리하는 큐 관리자의 활동 복구 로그를 유지보수합니다.이 로그 유형은 세 유형의 복구에 이러한 로그를 사용합니다.

1. 재시작 복구, 계획된 방식으로 IBM MQ를 중지하는 경우
2. 실패 복구, 실패로 인해 IBM MQ가 중지되는 경우
3. 매체 복원, 손상된 오브젝트를 복원하려는 경우

인플라이트 트랜잭션이 롤백되는 경우를 제외한 모든 경우에 복구는 큐 관리자가 중지될 때 인플라이트였던 모든 업데이트를 큐에서 제거하여 큐 관리자가 중지되었을 때의 상태로 큐 관리자를 복원합니다. 복구는 모든 지속 메시지를 복원하고 이 프로세스 중에 비지속 메시지는 손실될 수 있습니다.

자동 클라이언트 다시 연결

여러 개의 컴포넌트를 구성하여 추가 코드를 작성하지 않고 클라이언트 애플리케이션이 자동으로 다시 연결되도록 할 수 있습니다.

자동 클라이언트 다시 연결은 인라인입니다. 클라이언트 애플리케이션 프로그램의 어느 지점에서든지 연결이 자동으로 복원되며, 오브젝트를 열기 위한 핸들이 모두 복원됩니다.

반대로, 수동 다시 연결의 경우 클라이언트 애플리케이션이 MQCONN 또는 MQCONNX를 사용하여 연결을 다시 작성한 다음 오브젝트를 다시 열어야 합니다. 자동 클라이언트 다시 연결은 대부분의 클라이언트 애플리케이션에 적합하지만, 그렇다고 해서 모든 클라이언트 애플리케이션에 적합한 것은 아닙니다.

363 페이지의 표 28에서는 클라이언트 워크스테이션에 설치되어야 하는 IBM MQ 클라이언트 지원의 가장 초기 릴리스를 나열합니다. 애플리케이션이 자동 클라이언트 다시 연결을 사용하게 하려면 클라이언트 워크스테이션을 다음 레벨 중 하나로 업그레이드해야 합니다. 363 페이지의 표 29에서는 자동 클라이언트 다시 연결을 사용 가능하게 하기 위한 기타 요구사항을 나열합니다.

다시 연결 옵션에 대한 프로그램 액세스가 있는 클라이언트 애플리케이션은 다시 연결 옵션을 설정할 수 있습니다. JMS 및 XMS 클라이언트를 제외하고 클라이언트 애플리케이션이 다시 연결 옵션에 액세스할 수 있는 경우 이벤트 핸들러를 작성하여 다시 연결 이벤트를 핸들링할 수도 있습니다.

기존 클라이언트 애플리케이션은 재컴파일 및 링크 없이도 다시 연결 지원의 이점을 얻을 수 있습니다.

- 비JMS 클라이언트의 경우 다시 연결 옵션을 설정하려면 `mqclient.ini` 환경 변수를 DefRecon으로 설정하십시오. CCDT를 사용하여 큐 관리자에 연결하십시오. 클라이언트가 다중 인스턴스 큐 관리자에 연결해야 하는 경우 CCDT에서 활성 및 대기 큐 관리자 인스턴스의 네트워크 주소를 제공하십시오.
- JMS 클라이언트의 경우 연결 팩토리 구성에서 다시 연결 옵션을 설정하십시오. Java EE 서버의 EJB 컨테이너 내에서 실행하는 경우, MDB는 IBM MQ 자원 어댑터의 활성화 스펙 (또는 WebSphere Application Server에서 실행 중인 경우 리스너 포트) 에서 제공하는 재연결 메커니즘을 사용하여 IBM MQ 에 다시 연결할 수 있습니다. 그러나 이 시나리오에서는 자동 클라이언트 다시 연결은 지원되지 않으므로 애플리케이션이 MDB가 아닌 경우 (또는 웹 컨테이너에서 실행되는 경우) 애플리케이션은 그 자체 다시 연결 논리를 구현해야 합니다. IBM MQ 자원 어댑터는 메시지 구동 Bean으로의 메시지 전달을 위해 이 재연결 기능을 제공하지만 다른 Java EE 요소 (예: 서블릿) 는 자체 재연결을 구현해야 합니다.

참고: 자동 클라이언트 재연결은 IBM MQ classes for Java에서 지원되지 않습니다.

표 28. 지원되는 클라이언트			
클라이언트 인터페이스	**클라이언트**	**다시 연결 옵션에 대한 프로그램 액세스**	**다시 연결 지원**
메시징 API	C, C++, COBOL, 비관리 Visual Basic, XMS(Windows의 비관리 XMS)	7.0.1	7.0.1
	JMS(JSE 및 Java EE 클라이언트 컨테이너 및 관리 컨테이너)	7.0.1.3	7.0.1.3
	IBM MQ classes for Java	지원되지 않음	지원되지 않음
	관리 XMS 및 비관리 .NET 클라이언트: C#, Visual Basic,	7.1	7.1
기타 API	Windows Communication Foundation(비관리 1)	지원되지 않음	7.0.1
	Windows Communication Foundation(관리 1)	지원되지 않음	지원되지 않음
	Axis 1	지원되지 않음	지원되지 않음
	Axis 2	지원되지 않음	7.0.1.3
	HTTP(웹 2.0)	지원되지 않음	7.0.1.3

1. WCF 바인딩 구성에서 관리 또는 비관리 모드를 설정하십시오.

자동으로 다시 연결에는 다음 구성 요구사항이 있습니다.

표 29. 자동으로 다시 연결 구성 요구사항		
컴포넌트	**요구사항**	**요구사항을 충족시키지 못하는 경우의 결과**
IBM MQ MQI client 설치	363 페이지의 표 28 참조	`MQRC_OPTIONS_ERROR`
IBM MQ 서버 설치	레벨 7.0.1	`MQRC_OPTIONS_ERROR`
채널	`SHARECNV > 0`	`MQRC_ENVIRONMENT_ERROR`
애플리케이션 환경	스레드되어야 함	`MQRC_ENVIRONMENT_ERROR`
MQI	다음 중 하나입니다. • `MQCNO Options`이 `MQCNO_RECONNECT` 또는 `MQCNO_RECONNECT_Q_MGR`로 설정된 `MQCONNX`입니다. • `mqclient.ini`의 `Defrecon=YES|QMGR` • JMS에서 연결 팩토리의 CLIENTRECONNECTOPTIONS 특성을 설정합니다.	연결이 끊기거나 큐 관리자가 종료 또는 실패하는 경우 `MQCC_FAILED`입니다.

364 페이지의 그림 70에서는 클라이언트 다시 연결에 포함된 컴포넌트 간 기본 상호작용을 보여줍니다.

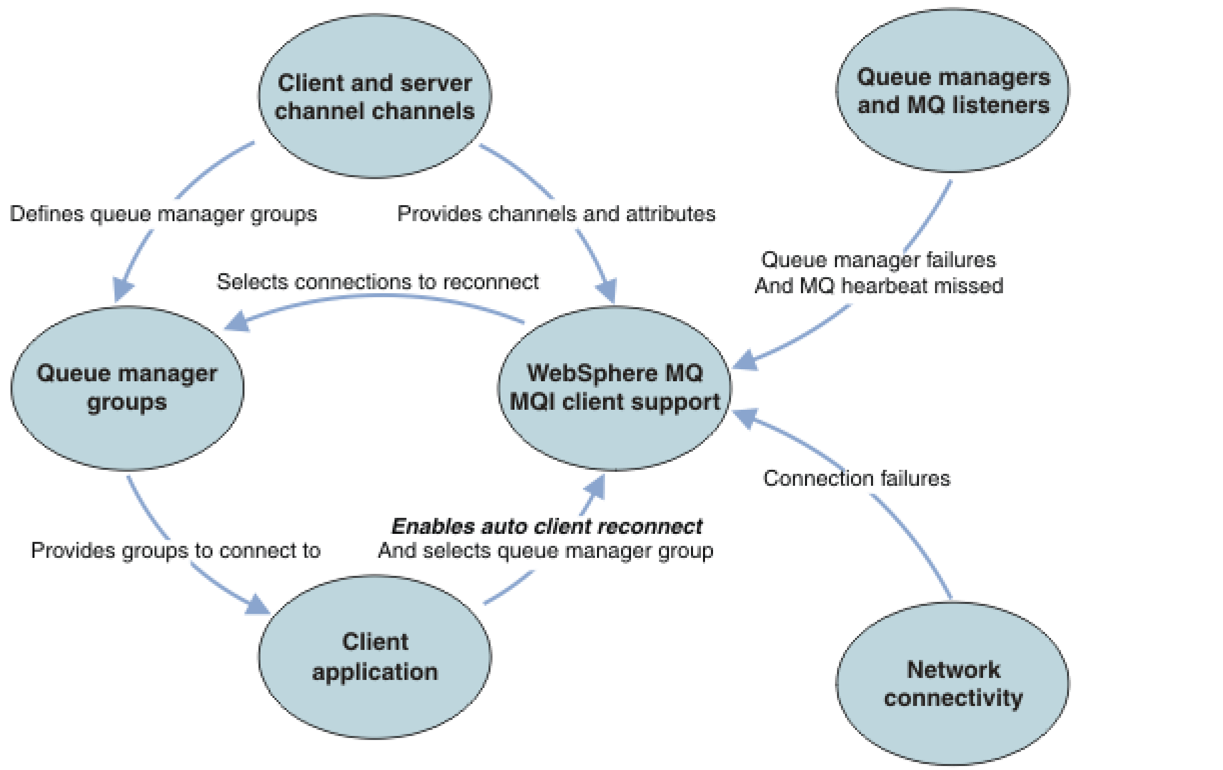


그림 *70.* 자동 클라이언트 다시 연결

클라이언트 애플리케이션

클라이언트 애플리케이션은 IBM MQ MQI client입니다.

- 기본적으로 클라이언트는 자동으로 다시 연결되지 않습니다. MQCONNX MQCNO Option MQCNO_RECONNECT 또는 MQCNO_RECONNECT_Q_MGR을 설정하여 자동 클라이언트 다시 연결을 사용 가능하게 하십시오.
- 다수의 애플리케이션은 추가 코드화 없이 자동 연결을 이용할 수 있는 방식으로 작성됩니다. 코드 변경을 하지 않고 mqclient.ini 구성 파일의 Channels 스탠자에서 DefRecon 속성을 설정하여 기존 프로그램에 대한 자동으로 다시 연결을 사용 가능하게 하십시오.
- 다음 세 가지 옵션 중 하나를 사용하십시오.
  1. 논리가 다시 연결의 영향을 받지 않도록 프로그램을 수정하십시오. 예를 들어, 동기점 내에서 MQI 호출을 발행하고 백아웃된 트랜잭션을 다시 제출해야 할 수도 있습니다.
  2. 다시 연결을 감지하기 위한 이벤트 핸들러를 추가하고 연결이 재설정될 때 클라이언트 애플리케이션의 상태를 복원하십시오.
  3. 자동으로 다시 연결을 사용 설정하지 마십시오. 대신 클라이언트의 연결을 끊고 새 MQCONN 또는 MQCONNX MQI 호출을 발행하여 같은 큐 관리자 그룹에서 실행되고 있는 다른 큐 관리자 인스턴스를 찾으십시오.

  이러한 세 가지 옵션에 대한 추가 세부사항은 443 페이지의 『애플리케이션 복구』의 내용을 참조하십시오.
- 같은 이름의 큐 관리자에 다시 연결한다고 해도 이 큐 관리자의 같은 인스턴스에 다시 연결되었음을 보장하지는 않습니다.

  MQCNO 옵션 MQCNO_RECONNECT_Q_MGR을 사용하여 같은 큐 관리자의 인스턴스에 다시 연결하십시오.
- 클라이언트는 다시 연결의 상태에 대한 알림을 받을 수 있도록 이벤트 핸들러에 등록할 수 있습니다. 이벤트 핸들러에 전달된 MQHCONN은 사용할 수 없습니다. 다음 이유 코드가 제공됩니다.

  MQRC_RECONNECTING
  : 연결에 실패했으며 시스템이 다시 연결을 시도합니다. 다시 연결을 여러 번 시도하면 다중 MQRC_RECONNECTING 이벤트를 수신합니다.

MQRC_RECONNECTED
다시 연결 모드 및 모든 핸들이 성공적으로 재설정되었습니다.

MQRC_RECONNECT_FAILED
다시 연결에 실패했습니다.

MQRC_RECONNECT_QMID_MISMATCH
다시 연결 가능한 연결이 MQCNO_RECONNECT_Q_MGR을 지정했고 연결이 다른 큐 관리자에 다시 연결하려고 시도했습니다.

MQRC_RECONNECT_Q_MGR_REQD
MQGET 호출의 MQMO_MATCH_MSG_TOKEN과 같은 옵션이 동일한 큐 관리자에 대한 다시 연결을 요구하는 클라이언트 프로그램에서 지정되었습니다.

- 다시 연결 가능 클라이언트는 연결 후에만 자동으로 다시 연결할 수 있습니다. 즉, MQCONNX 호출 자체는 실패한 경우 다시 시도되지 않습니다. 예를 들어, MQCONNX에서 리턴 코드 2543 - MQRC_STANDBY_Q_MGR을 수신하는 경우 짧은 지연 후에 호출을 재발행하십시오.

MQRC_RECONNECT_INCOMPATIBLE
이 이유 코드는 다시 연결 옵션이 설정된 경우 애플리케이션이 MQPMO_LOGICAL_ORDER(MQPUT 및 MQPUT1과 함께) 또는 MQGMO_LOGICAL_ORDER(MQGET과 함께)를 사용하려고 시도할 때 리턴됩니다. 이 이유 코드를 리턴하는 이유는 애플리케이션이 그러한 경우에 다시 연결을 사용하지 않도록 하기 위해서입니다.

MQRC_CALL_INTERRUPTED
이 이유 코드는 커미트 호출을 실행하는 중에 연결이 끊기고 클라이언트가 다시 연결되면 리턴됩니다. 동기점 외부 지속 메시지의 MQPUT에 인해서도 같은 이유 코드가 애플리케이션에 리턴됩니다.

다중 인스턴스 큐 관리자

다중 인스턴스 큐 관리자가 자동 클라이언트 다시 연결을 사용하여 그 대기 인스턴스를 활성화한 후에는 IBM MQ MQI client 애플리케이션 재시작을 단순화하십시오.

다중 인스턴스 큐 관리자의 대기 인스턴스는 일반적으로 활성 인스턴스에 대한 다른 네트워크 주소에 있습니다. 두 인스턴스 모두의 네트워크 주소를 클라이언트 연결 정의 테이블(CCDT)에 포함시키십시오. **CONNAME** 매개변수에 대한 네트워크 주소 목록을 제공하거나 CCDT에서 큐 관리자에 대한 다중 행을 정의하십시오.

일반적으로 IBM MQ MQI clients는 큐 관리자 그룹에 있는 모든 큐 관리자에 다시 연결됩니다. IBM MQ MQI client가 동일한 큐 관리자에만 다시 연결되게 하려고 하는 경우가 있습니다. 이는 큐 관리자에 대한 연관관계가 있을 수도 있습니다. 클라이언트가 다른 큐 관리자에 다시 연결하는 것을 막을 수 있습니다. MQCNO 옵션 MQCNO_RECONNECT_Q_MGR을 설정하십시오. IBM MQ MQI client는 다른 큐 관리자에 다시 연결하는 경우 실패합니다. MQCNO 옵션인 MQCNO_RECONNECT_Q_MGR을 설정하는 경우 같은 큐 관리자 그룹에 다른 큐 관리자를 포함시키지 마십시오. 다시 연결하는 큐 관리자가 원래 연결했던 큐 관리자와 다른 경우 클라이언트는 오류를 리턴합니다.

큐 관리자 그룹

같은 이름의 큐 관리자, 동일한 큐 관리자 또는 클라이언트 연결 테이블에서 같은 QMNAME 값으로 정의되는 일련의 큐 관리자 중 클라이언트 애플리케이션이 항상 연결 및 다시 연결할 대상을 선택할 수 있습니다.

- 클라이언트 채널 정의에 있는 큐 관리자 이름 속성 QMNAME은 큐 관리자 그룹의 이름입니다.
- 클라이언트 애플리케이션에서 MQCONN 또는 MQCONNX QmgrName 매개변수의 값을 큐 관리자 이름으로 설정하면 클라이언트가 해당 이름을 가진 큐 관리자에만 연결됩니다. 큐 관리자 이름에 별표(*)로 접두부를 지정하는 경우 클라이언트는 큐 관리자 그룹에 있는 큐 관리자 중 같은 QMNAME 값을 가지는 큐 관리자에 연결됩니다. 자세한 설명은 CCDT의 큐 관리자 그룹을 참조하십시오.

큐 공유 그룹

z/OS z/OS 큐 공유 그룹에 대한 자동 클라이언트 다시 연결은 다른 환경과 같은 다시 연결 메커니즘을 사용합니다. 클라이언트는 원래 연결에 대해 구성된 것과 같이 선택된 큐 관리자에 다시 연결됩니다. 예를 들어, 클라이언트 채널 정의 테이블을 사용하는 경우 관리자는 테이블의 모든 항목이 같은 z/OS 큐 공유 그룹으로 해석되도록 해야 합니다.

클라이언트 및 서버 채널 정의

클라이언트 및 서버 채널 정의는 클라이언트 애플리케이션이 다시 연결할 수 있는 큐 관리자 그룹을 정의합니다. 이러한 정의는 다시 연결 선택 및 시간 선택, 그리고 보안과 같은 기타 요소도 결정합니다. 관련 주제를 참조하십시오. 다시 연결에 고려해야 할 가장 관련된 채널 속성이 다음 두 그룹에 나열되어 있습니다.

클라이언트 연결 속성

연결 연관관계(AFFINITY) AFFINITY
: 연결 연관관계입니다.

클라이언트 채널 위젯(CLNTWGHT) CLNTWGHT
: 클라이언트 채널 위젯입니다.

연결 이름(CONNAME) CONNAME
: 연결 정보입니다.

하트비트 간격(HBINT) HBINT
: 하트비트 간격 서버 연결 채널에서 하트비트 간격을 설정하십시오.

활성 유지(keepalive) 간격(KAINT) KAINT
: 활성 유지(keepalive) 간격입니다. 서버 연결 채널에서 활성유지(keepalive) 간격을 설정하십시오.

z/OS KAINT는 z/OS에만 적용됨을 참고하십시오.

큐 관리자 이름(QMNAME) QMNAME
: 큐 관리자 이름.

서버 연결 속성

하트비트 간격(HBINT) HBINT
: 하트비트 간격 클라이언트 연결 채널에서 하트비트 간격을 설정하십시오.

활성 유지(keepalive) 간격(KAINT) KAINT
: 활성 유지(keepalive) 간격입니다. 클라이언트 연결 채널에서 활성유지(keepalive) 간격을 설정하십시오.

z/OS KAINT는 z/OS에만 적용됨을 참고하십시오.

KAINT는 네트워크 계층 하트비트이고 HBINT는 클라이언트와 큐 관리자 간 IBM MQ 하트비트입니다. 이러한 하트비트를 짧은 시간으로 설정하는 것에는 다음과 같은 두 가지 목적이 있습니다.

1. 연결에서 활동을 시뮬레이션하면 비활성 연결을 닫는 것을 담당하는 네트워크 계층 소프트웨어가 연결을 닫을 가능성이 낮아집니다.
2. 연결이 종료되면 연결이 끊긴 연결이 감지되기 전까지의 지연이 단축됩니다.

기본 TCP/IP 활성 유지(keepalive) 간격은 2시간입니다. KAINT 및 HBINT 속성을 짧게 설정하는 것을 고려해 보십시오. 네트워크의 정상적 작동이 자동으로 다시 연결의 요구에 적합한 것으로 가정하지 마십시오. 예를 들어, 일부 방화벽은 10분 정도 후에 활성 TCP/IP 연결을 종료할 수 있습니다.

네트워크 연결성

네트워크에 의해 IBM MQ MQI client에 전달되는 네트워크 실패만 클라이언트의 자동으로 다시 연결 기능에 의해 핸들링됩니다.

- 전송에 의해 자동으로 수행되는 다시 연결은 IBM MQ에 표시되지 않습니다.
- HBINT를 설정하면 IBM MQ에 표시되지 않는 네트워크 실패를 처리하는 데 도움이 됩니다.

큐 관리자 및 IBM MQ 리스너

클라이언트 다시 연결은 서버 실패, 큐 관리자 실패, 네트워크 연결 실패 및 다른 큐 관리자 인스턴스로 전환하는 관리자에 의해 트리거됩니다.

- 다중 인스턴스 큐 관리자를 사용하는 경우 활성 큐 관리자 인스턴스를 대기 인스턴스로 제어를 전환할 때 클라이언트 다시 연결의 추가 원인이 발생합니다.
- 기본 **endmqm** 명령을 사용하여 큐 관리자를 종료하면 자동 클라이언트 다시 연결을 트리거하지 않습니다. **endmqm** 명령에 `-r` 옵션을 추가하여 자동 클라이언트 다시 연결을 요청하거나 `-s` 옵션을 추가하여 종료 후 대기 큐 관리자 인스턴스로 전송하십시오.

IBM MQ MQI client 자동으로 다시 연결 지원

IBM MQ MQI client에서 자동 클라이언트 다시 연결 지원을 사용하면 큐 관리자에 다시 연결하기 위해 MQCONN 또는 MQCONNX MQI 호출을 발행하지 않고도 클라이언트 애플리케이션이 자동으로 다시 연결하고 처리를 계속합니다.

- 자동 클라이언트 다시 연결은 다음의 경우 중 하나에 의해 처리됩니다.
  - 큐 관리자 실패
  - 큐 관리자가 종료되고 **endmqm** 명령에 다시 연결 옵션인 `-r` 지정
- MQCONNX MQCNO 옵션은 자동 클라이언트 다시 연결을 사용 설정했는지 여부를 제어합니다. 이 옵션은 다시 연결 옵션에 설명되어 있습니다.
- 자동 클라이언트 다시 연결은 프로그램이 끊어진 연결로 인해 발생한 MQI 오류를 처리한 후에 정상적인 처리를 계속할 수 있도록 애플리케이션을 대신해서 MQI 호출을 발행하여 연결 핸들 및 다른 열린 오브젝트에 대한 핸들을 복원합니다. 445 페이지의 『자동으로 다시 연결된 클라이언트의 복구』의 내용을 참조하십시오.
- 연결에 대한 채널 엑시트 프로그램을 작성한 경우 엑시트는 이러한 추가 MQI 호출을 수신합니다.
- 다시 연결이 시작되고 끝날 때 트리거되는 다시 연결 이벤트 핸들러를 등록할 수 있습니다.

예상한 다시 연결 시간은 1분을 초과하지 않더라도 큐 관리자가 관리할 자원이 많을 수 있기 때문에 다시 연결에 더 긴 시간이 소요될 수 있습니다. 이 시간 동안 클라이언트 애플리케이션은 IBM MQ 자원에 속하지 않은 잠금을 보유할 수 있습니다. 클라이언트가 다시 연결을 대기하는 시간을 제한하기 위해 구성할 수 있는 제한시간 값이 있습니다. 이 값(초)은 `mqclient.ini` 파일에서 설정됩니다.

```
Channels:
MQReconnectTimeout = 1800
```

제한시간이 만료된 후에는 다시 연결 시도가 이루어지지 않습니다. 시스템이 제한시간이 만료되었음을 감지하면 `MQRC_RECONNECT_FAILED` 오류를 리턴합니다.

z/OS 콘솔 메시지 모니터링

IBM MQ for z/OS에서 큐 관리자 또는 채널 시작기가 발행하는 특별히 중요하게 고려해야 하는 다양한 정보 메시지가 있습니다. 이러한 메시지는 그 자체로는 문제점을 표시하지 않지만 처리가 필요한 잠재적 문제를 나타내기 때문에 추적에 도움이 될 수 있습니다.

다음 콘솔 메시지가 있을 경우 사용자 애플리케이션이 대량의 메시지를 페이지 세트에 넣고 있음을 표시하는 것일 수 있고 이는 더 큰 문제점의 증상일 수 있습니다.

- 비제어 루프와 같이 메시지를 넣는 사용자 애플리케이션에 문제점이 있습니다.
- 큐에서 메시지를 가져오는 사용자 애플리케이션이 더 이상 작동하지 않습니다.

모니터할 콘솔 메시지

다음 목록은 잠재적으로 더 큰 문제점을 나타낼 수 있는 메시지의 개요를 제공합니다. 시스템 자동화에 대해 이러한 메시지를 추적하는 것이 필요한지 여부를 판별하고 잠재적 문제점을 효과적으로 후속 조치할 수 있도록 적절한 문서를 제공하십시오.

CSQI004I: *csect-name* CONSIDER INDEXING *queue-name* BY *index-type* FOR *connection-type* CONNECTION *connection-name*, *num-msgs* MESSAGES SKIPPED

- 큐 관리자가 색인이 정의되지 않은 큐에서 메시지 ID 또는 상관 ID별로 메시지를 수신하는 애플리케이션을 감지했습니다.
- 로컬 큐 오브젝트 *queue-name*의 INDXTYPE 속성이 *index-type* 값을 가지도록 대체하여 식별된 큐에 대해 색인을 설정하는 것을 고려하십시오.

CSQI031I: *csect-name* THE NEW EXTENT OF PAGE SET *psid* HAS FORMATTED SUCCESSFULLY

- 이 페이지 세트에 할당된 큐의 curdepth를 검사하십시오.
- 이 메시지를 처리하려면 실패의 원인을 조사하십시오.

•

CSQI041I: *csect-name* JOB *jobname* USER *userid* HAD ERROR ACCESSING PAGE SET *psid*

- 페이지 세트가 큐 관리자에게 할당되는지 여부를 판별하십시오.
- **DISPLAY USAGE** 명령을 발행하여 페이지 세트의 상태를 판별하십시오.
- 추가 오류 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.

•

CSQI045I: *csect-name* Log RBA has reached *rba*. Plan a log reset

- 편리할 때 큐 관리자 중지를 계획하고 로그를 재설정하십시오.
- 큐 관리자가 6바이트 로그 RBA를 사용 중인 경우에는 8바이트 로그 RBA를 사용하도록 큐 관리자를 변환하는 것을 고려하십시오.

CSQI046E: *csect-name* Log RBA has reached *rba*. Perform a log reset

- 편리할 때 큐 관리자 중지를 계획하고 로그를 재설정하십시오.
- 큐 관리자가 6바이트 로그 RBA를 사용 중인 경우에는 8바이트 로그 RBA를 사용하도록 큐 관리자를 변환하는 것을 고려하십시오.

CSQI047E: *csect-name* Log RBA has reached *rba*. Stop queue manager and reset logs

- 큐 관리자를 즉시 중지하고 로그를 재설정하십시오.
- 큐 관리자가 6바이트 로그 RBA를 사용 중인 경우에는 8바이트 로그 RBA를 사용하도록 큐 관리자를 변환하는 것을 고려하십시오.

CSQJ004I: ACTIVE LOG COPY *n* INACTIVE, LOG IN SINGLE MODE, ENDRBA= *ttt*

- 큐 관리자가 '단일' 로깅 모드를 활성화했습니다. 이는 로그 오프로드 문제점을 나타내는 경우일 때 있습니다.
- **DISPLAY LOG** 명령을 발행하여 활성 로그 및 아카이브 로그의 이중화를 위한 설정을 판별하십시오. 이 표시는 오프로드 처리가 필요한 활성 로그 수도 표시합니다.
- 추가 오류 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.

CSQJ031D: *csect-name*, THE LOG RBA RANGE MUST BE RESET. REPLY 'Y' TO CONTINUE STARTUP OR 'N' TO SHUTDOWN

- 큐 관리자를 중지하고 가능한 빨리 로그를 재설정하십시오.
- 큐 관리자가 6바이트 로그 RBA를 사용 중인 경우에는 8바이트 로그 RBA를 사용하도록 큐 관리자를 변환하는 것을 고려하십시오.

CSQJ032E: *csect-name alert-lvl* - APPROACHING END OF THE LOG RBA RANGE OF *max-rba*. CURRENT LOG RBA IS *current-rba*.

- 큐 관리자를 중지하고 가능한 빨리 로그를 재설정하십시오.
- 큐 관리자가 6바이트 로그 RBA를 사용 중인 경우에는 8바이트 로그 RBA를 사용하도록 큐 관리자를 변환하는 것을 고려하십시오.

CSQJ110E: LAST COPY*n* ACTIVE LOG DATA SET IS *nnn* PERCENT FULL

- 표시 요청을 수행하여 로그 오프로드 프로세스와 관련된 미해결 요청을 확인하여 다른 대기 중인 오프로드 태스크를 완료하는 단계를 수행하십시오. 필요한 조치를 수행하여 요청을 충족시키고 오프로드가 계속될 수 있도록 허용하십시오.
- 충분한 활성 로그 데이터 세트가 있는지 여부를 고려하십시오. 필요한 경우, DEFINE LOG 명령을 사용하여 동적으로 추가 로그 데이터 세트를 추가하십시오.

CSQJ111A: 활성 로그 데이터 SETS IN 공간

- 표시 요청을 수행하여 로그 오프로드 프로세스와 관련된 미해결 요청이 없는지 확인하십시오. 필요한 조치를 수행하여 요청을 충족시키고 오프로드가 계속될 수 있도록 허용하십시오.
- 충분한 활성 로그 데이터 세트가 있는지 여부를 고려하십시오. 필요한 경우, DEFINE LOG 명령을 사용하여 동적으로 추가 로그 데이터 세트를 추가하십시오.

- 오프로드에 필요한 자원이 부족하여 지연이 발생한 경우, 필수 자원이 사용 가능해야 오프로드가 완료되어 로깅이 진행되도록 허용할 수 있습니다. 이 조건의 복구에 대한 정보는 아카이브 로그 문제점을 참조하십시오.

CSQJ114I: ERROR ON ARCHIVE DATA SET, OFFLOAD CONTINUING WITH ONLY ONE ARCHIVE DATA SET BEING GENERATED

- 추가 오류 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.
- 아카이브 로그의 두 번째 사본을 작성하고 BSDS를 수동으로 업데이트하십시오.

CSQJ115E: 오프로드 실패, 아카이브 데이터 세트를 할당할 수 없습니다.

메시지 CSQJ103E 또는 CSQJ073E의 오류 상태 정보를 검토하십시오. 데이터 세트 할당 오류의 원인이 되는 조건을 수정하여 재시도 시 오프로드가 발생할 수 있도록 하십시오.

CSQJ136I: UNABLE TO ALLOCATE TAPE UNIT FOR CONNECTION-ID= *xxxx* CORRELATION-ID= *yyyyyy*, *m* ALLOCATED *n* ALLOWED

- 추가 오류 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.

.

CSQJ151I: *csect-name* ERROR READING RBA *rrr*, CONNECTION-ID= *xxxx* CORRELATION-ID= *yyyyyy* REASON CODE= *ccc*

- 추가 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.
- **`DISPLAY CONN`** 명령을 발행하여 활동을 커미트하지 않는 연결을 판별하십시오.
- 애플리케이션이 해당 업데이트를 커미트할 수 있는지 확인하십시오.

CSQJ160I: LONG-RUNNING UOW FOUND, URID= *urid* CONNECTION NAME= *name*

- 추가 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.
- **`DISPLAY CONN`** 명령을 발행하여 활동을 커미트하지 않는 연결을 판별하십시오.
- 애플리케이션이 해당 업데이트를 커미트할 수 있는지 확인하십시오.

CSQJ161I: UOW UNRESOLVED AFTER *n* OFFLOADS, URID= *urid* CONNECTION NAME= *name*

- 페이지 세트가 큐 관리자에게 할당되는지 여부를 판별하십시오.
- **`DISPLAY USAGE`** 명령을 발행하여 페이지 세트의 상태를 판별하십시오.
- 추가 메시지가 있는지 큐 관리자 작업 로그를 검사하십시오.

CSQP011E: CONNECT ERROR STATUS *ret-code* FOR PAGE SET *psid*

- 이 페이지 세트에 할당된 큐의 curdepth를 검사하십시오.
- 메시지를 처리하려면 실패의 원인을 조사하십시오.

CSQP013I: *csect-name* NEW EXTENT CREATED FOR PAGE SET psid. NEW EXTENT WILL NOW BE FORMATTED

- 이 페이지 세트에 할당된 큐의 curdepth를 검사하십시오.
- 메시지를 처리하려면 실패의 원인을 조사하십시오.
- 큐를 다른 페이지 세트에 재할당해야 하는지 여부를 판별하십시오.
- 볼륨이 가득 찬 경우 페이지 세트를 다중 볼륨 데이터 세트로 설정해야 하는지 여부를 판별하십시오. 페이지 세트가 이미 다중 볼륨인 경우에는 사용 중인 스토리지 그룹에 추가 볼륨을 추가하는 것을 고려해 보십시오. 추가 공간이 사용 가능해지면 페이지 세트 **EXPAND** 메소드를 **SYSTEM**으로 설정하여 확장을 재시도하십시오. 재시도가 필요한 경우에는 **EXPAND**를 **SYSTEM**으로 전환한 다음 일반 설정으로 돌아오십시오.

CSQP014E: *csect-name* EXPANSION FAILED FOR PAGE SET psid. FUTURE REQUESTS TO EXTEND IT WILL BE REJECTED

- 이 페이지 세트에 할당된 큐의 curdepth를 검사하십시오.
- 메시지를 처리하려면 실패의 원인을 조사하십시오.
- 큐를 다른 페이지 세트에 재할당해야 하는지 여부를 판별하십시오.

CSQP016E: *csect-name* PAGE SET *psid* HAS REACHED THE MAXIMUM NUMBER OF EXTENTS. IT CANNOT BE EXTENDED AGAIN

- 이 페이지 세트에 할당된 큐의 curdepth를 검사하십시오.
- 메시지를 처리하려면 실패의 원인을 조사하십시오.

CSQP017I: *csect-name* EXPANSION STARTED FOR PAGE SET *psid*
DISPLAY THREAD 명령을 발행하여 IBM MQ에서 작업 단위의 상태를 판별하십시오.

CSQP047E: Unavailable page sets can cause problems - take action to correct this situation

- 시스템 프로그래머 응답을 따르십시오.

CSQQ008I: *nn* units of recovery are still in doubt in queue manager *qqqq*

- 데드-레터 큐의 상태를 조사하십시오. 데드-레터 큐가 넣기 사용 안함으로 설정되지 않았는지 확인하십시오.
- 데드-레터 큐가 MAXMSG 한계에 있지 않은지 확인하십시오.

CSQQ113I: *psb-name region-id* This message cannot be processed

- CSQOUTX 데이터 세트를 검사하여 CSQINPX 실패의 원인을 판별하십시오.
- 일부 명령은 처리되지 않을 수도 있습니다.

CSQX035I: *csect-name* Connection to queue manager *qmgr-name* stopping or broken, MQCC= *mqcc* MQRC= *mqrc* (*mqrc-text*

- MQRC를 검사하여 실패의 원인을 판별하십시오.
- 이러한 코드는 IBM MQ for z/OS 메시지, 완료 및 이유 코드에서 설명합니다.

CSQX032I: *csect-name* Initialization command handler terminated

- MQRC를 검사하여 실패의 원인을 판별하십시오.
- 이러한 코드는 IBM MQ for z/OS 메시지, 완료 및 이유 코드에서 설명합니다.

CSQX048I: *csect-name* Unable to convert message for *name*, MQCC= *mqcc* MQRC= *mqrc* (*mqrc-text*)

- 작업 로그를 검사하여 TCP/IP 실패의 원인을 판별하십시오.
- 오류가 있는지 TCP/IP 주소 공간을 검사하십시오.

CSQX234I: *csect-name* Listener stopped, TRPTYPE= *trptype* INDISP= *disposition*

- **STOP** 명령 후에도 리스너가 중지되지 않는 경우 오류가 있는지 TCP/IP 주소 공간을 검사하십시오.
- 시스템 프로그래머 응답을 따르십시오.

CSQX407I: *csect-name* Cluster queue *q-name* definitions inconsistent

- 클러스터 내 여러 클러스터 큐의 값이 불일치합니다. 그 차이를 조사하고 해결하십시오.

CSQX411I: *csect-name* Repository manager stopped

- 오류로 인해 저장소 관리자가 중지된 경우 메시지가 있는지 작업 로그를 검사하십시오.

CSQX417I: *csect-name* Cluster-senders remain for removed queue manager *qmgr-name*

- 시스템 프로그래머 응답을 따르십시오.

CSQX418I: *csect-name* Only one repository for cluster *cluster_name*

- 증가된 고가용성을 위해 클러스터는 두 개의 전체 저장소로 구성되어야 합니다.

CSQX419I: *csect-name* No cluster-receivers for cluster *cluster_name*

- 시스템 프로그래머 응답을 따르십시오.

CSQX420I: *csect-name* No repositories for cluster *cluster_name*

- 시스템 프로그래머 응답을 따르십시오.

CSQX448E: *csect-name* Repository manager stopping because of errors. *n*초 후에 재시작

- 시스템 프로그래머 응답을 따르십시오.

이 메시지는 다음 명령을 사용하여 SYSTEM.CLUSTER.COMMAND.QUEUE가 사용으로 설정될 때까지 600초(10분) 간격으로 기록됩니다.

```
ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(ENABLED)
```

첫 번째 CSQX448E 메시지가 발행되기 전에 저장소 관리자가 종료되는 문제점을 해결하기 위해서는 큐를 사용으로 설정하기 전에 수동 개입이 필요할 수 있습니다.

고가용성 구성

IBM MQ 큐 관리자를 고가용성(HA) 구성에서 작동시키려는 경우 큐 관리자가 AIX용 PowerHA(이전의 HACMP) 또는 MSCS(Microsoft Cluster Service)와 같은 고가용성 관리자 또는 IBM MQ 다중 인스턴스 큐 관리자와 함께 작업하도록 설정할 수 있습니다. V 9.0.4 Linux 시스템에서는 고가용성을 제공하기 위해 쿼럼 기반 그룹을 사용하는 복제 데이터 큐 관리자(RDQM)를 배치할 수도 있습니다.

다음 구성 정의에 대해 알고 있어야 합니다.

큐 관리자 클러스터
하나 이상의 컴퓨터에 있는 둘 이상의 큐 관리자 그룹으로, 자동 상호연결을 제공하고 로드 밸런싱 및 중복성을 위해 큐가 서로 공유될 수 있도록 합니다. IBM WebSphere MQ 7.1부터, 클러스터 오류 복구에서는 문제점이 해결될 때까지 문제점의 원인이 되는 조작을 재실행합니다.

HA 클러스터
HA 클러스터는 둘 이상의 컴퓨터와 디스크 및 네트워크와 같은 자원으로 이루어진 그룹이며 이러한 구성요소는 서로 연결되어 하나가 실패하면 HACMP(UNIX) 또는 MSCS(Windows)와 같은 고가용성 관리자가 장애 복구를 수행하는 방식으로 구성됩니다. 장애 복구는 애플리케이션의 상태 데이터를 실패한 컴퓨터에서 클러스터 내 다른 컴퓨터로 전송하고 그 위치에서 해당 조작을 다시 시작합니다. 이는 HA 클러스터 내에서 실행되는 서비스의 고가용성을 제공합니다. IBM MQ 클러스터와 HA 클러스터 간 관계는 372 페이지의 『큐 관리자 클러스터에 대한 HA 클러스터의 관계』에 설명되어 있습니다.

다중 인스턴스 큐 관리자
둘 이상의 컴퓨터에서 구성된 동일한 큐 관리자의 인스턴스입니다. 다중 인스턴스를 시작하면 한 인스턴스가 활성 인스턴스가 되고 나머지 다른 인스턴스는 대기 인스턴스가 됩니다. 활성 인스턴스가 실패할 경우 다른 컴퓨터에서 실행 중인 대기 인스턴스가 자동으로 인계받습니다. HACMP 또는 MSCS와 같은 클러스터 기술 없이도 다중 인스턴스 큐 관리자를 사용하여 IBM MQ를 기반으로 하는 자체 고가용성 메시징 시스템을 구성할 수 있습니다. HA 클러스터 및 다중 인스턴스 큐 관리자는 큐 관리자의 가용성을 높이기 위한 대체 방법입니다. HA 클러스터에 다중 인스턴스 큐 관리자를 넣어 결합하지 마십시오.

V 9.0.4 **고가용성 복제 데이터 큐 관리자(HA RDQM)**
세 Linux 서버로 구성된 그룹의 각 노드에서 구성된 동일한 큐 관리자의 인스턴스입니다. 세 인스턴스 중 하나가 활성 인스턴스입니다. 활성 큐 관리자의 데이터는 다른 두 인스턴스에 동기적으로 복제되므로, 실패 발생 시 이러한 인스턴스 중 하나가 인계할 수 있습니다. 서버 그룹화는 Pacemaker로 제어되고 복제는 DRBD로 제어됩니다.

V 9.0.5 **재해 복구 복제 데이터 큐 관리자(DR RDQM)**
큐 관리자는 한 사이트에서 기본 노드에 실행되고 다른 사이트에서 복구 노드에 두 번째 인스턴스를 가질 수 있습니다. 데이터는 기본 인스턴스와 보조 인스턴스 사이에 복제되고 만약 어떤 이유로 기본 노드가 유실되면 보조 노드가 기본 인스턴스가 되어 시작됩니다. 두 노드 모두 Linux 서버여야 합니다. 복제는 DRBD로 제어됩니다.

다중 인스턴스 큐 관리자와 HA 클러스터 간 차이점

다중 인스턴스 큐 관리자와 HA 클러스터는 큐 관리자에 대한 높은 가용성을 얻기 위한 대체 방법입니다. 다음은 두 가지 접근법 간 차이점을 강조하는 몇 가지 포인트입니다.

다중 인스턴스 큐 관리자에는 다음과 같은 기능이 있습니다.

- IBM MQ로 통합된 기본 장애 복구 지원

- HA 클러스터보다 빠른 장애 복구
- 단순한 구성 및 조작
- IBM MQ Explorer와 통합

다중 인스턴스 큐 관리자의 제한사항은 다음과 같습니다.

- 가용성이 높고 고성능 네트워크 스토리지가 필요함
- 큐 관리자가 장애 복구 시 IP 주소를 변경함으로 인한 복잡한 네트워크 구성

HA 클러스터에는 다음과 같은 기능이 있습니다.

- 애플리케이션 서버 또는 데이터베이스와 같은 여러 자원을 조정하는 기능
- 셋 이상의 노드로 구성된 클러스터 등 보다 유연한 구성 옵션
- 운영자의 개입 없이도 여러 번 장애 복구 가능
- 장애 복구의 일부로 큐 관리자의 IP 주소 인계

HA 클러스터의 제한사항은 다음과 같습니다.

- 추가 제품 구매 및 기술이 필요함
- 클러스터의 노드 사이에서 전환할 수 있는 디스크가 필요함
- HA 클러스터의 구성이 상대적으로 복잡함
- 과거에는 장애 복구가 비교적 느렸지만 최근 HA 클러스터 제품에서는 개선됨
- 큐 관리자와 같은 자원을 모니터하는 데 사용되는 스크립트에 결함이 있는 경우 불필요한 장애 복구가 발생할 수 있음

큐 관리자 클러스터에 대한 HA 클러스터의 관계

큐 관리자 클러스터는 큐 관리자 클러스터 큐의 사용 가능한 인스턴스 전체에서 메시지의 로드 밸런싱을 제공합니다. 이는 단일 큐 관리자보다 높은 가용성을 제공합니다. 큐 관리자의 실패 후에도 메시징 애플리케이션이 큐 관리자 클러스터 큐의 남아 있는 인스턴스에 대해 계속해서 메시지를 송신하고 액세스할 수 있기 때문입니다. 그러나 큐 관리자 클러스터가 클러스터 내 사용 가능한 큐 관리자에게 자동으로 새 메시지를 라우트한다고 해도 사용 불가능한 큐 관리자에 현재 큐 대기 중인 메시지는 큐 관리자를 재시작할 때까지 사용 불가능합니다. 이러한 이유로 큐 관리자 클러스터만으로는 모든 메시지 데이터의 고가용성을 제공하지 않지만 큐 관리자 실패의 자동 감지 및 큐 관리자 재시작 또는 장애 복구의 자동 트리거를 제공합니다. 고가용성(HA) 클러스터는 이러한 기능을 제공합니다. 좋은 효과를 얻기 위해 두 유형의 클러스터를 함께 사용할 수 있습니다. 큐 관리자 클러스터에 대한 소개는 클러스터 설계를 참조하십시오.

Linux UNIX UNIX and Linux의 HA 클러스터

UNIX and Linux 플랫폼에서 고가용성 (HA) 클러스터와 함께 IBM MQ 를 사용할 수 있습니다. 예를 들어, AIX (이전의 HACMP), Veritas Cluster Server, HP Service가드 또는 Red Hat Cluster Suite와 함께 Red Hat Enterprise Linux 클러스터를 사용할 PowerHA 있습니다.

IBM WebSphere MQ 7.0.1 전에는 HA 클러스터 구성을 지원하기 위해 SupportPac MC91이 제공되었습니다. IBM WebSphere MQ 7.0.1은 큐 관리자가 자신의 데이터를 저장하는 이전 버전에 비해 훨씬 더 높은 수준의 제어를 제공했습니다. 이는 HA 클러스터에서 큐 관리자를 구성하는 것을 더욱 용이하게 합니다. SupportPac MC91에서 제공되는 스크립트의 대부분은 더 이상 필요하지 않으며 SupportPac의 제공은 중단됩니다.

이 절에서는 372 페이지의 『HA 클러스터 구성』, 큐 관리자 클러스터에 대한 HA 클러스터의 관계, 373 페이지의 『IBM MQ 클라이언트』 및 373 페이지의 『HA 클러스터에서 작동하는 IBM MQ』에 대해 소개하며 단계를 통해 안내하고 HA 클러스터와 큐 관리자 구성에 맞게 조정할 수 있는 스크립트 예를 제공합니다.

이 절에서 설명하는 구성 단계에 대한 지원은 사용자 환경에 맞는 HA 클러스터 문서를 참조하십시오.

HA 클러스터 구성

이 절에서 용어 노드는 운영 체제 및 HA 소프트웨어를 실행하는 엔티티를 참조하는 데 사용되며 "컴퓨터", "시스템" 또는 "머신" 또는 "파티션" 또는 "블레이드"는 이 사용법에서 동의어로 간주될 수 있습니다. IBM MQ를 사용

하여 모든 클러스터 노드가 IBM MQ 워크로드를 실행하는 상호 인수를 포함하여 대기 또는 인수 구성을 설정하는 데 도움을 줄 수 있습니다.

대기 구성은 한 노드가 작업을 수행하는 동안 다른 노드는 대기 역할만 하는 가장 기본적인 HA 클러스터입니다. 대기 노드는 작업을 수행하지 않으며 유휴로 참조됩니다. 이 구성은 경우에 따라 콜드 대기라고 합니다. 이러한 구성에는 높은 수준의 하드웨어 중복성이 필요합니다. 하드웨어를 절약하기 위해 단일 대기 노드와 다중 작업자 노드를 가지도록 이 구성을 확장할 수 있습니다. 요점은 대기 노드가 다른 작업자 노드의 작업을 인수할 수 있다는 것입니다. 이 구성은 계속해서 대기 구성으로 참조되며 "N+1" 구성이라고 하는 경우도 있습니다.

인수 구성은 모든 노드가 일부 작업을 수행하고 노드 실패가 발생할 경우 중요한 작업은 인계될 수 있는 고급 구성입니다.

한 측 인수 구성은 대기 노드가 일부 추가 작업, 비중요 작업 및 이동 불가능한 작업을 수행하는 구성입니다. 이 구성은 대기 구성과 비슷하지만 (비중요) 작업이 대기 노드에 의해 수행됩니다.

상호 인수 구성은 모든 노드가 높은 가용성의 (이동 가능) 작업을 수행하는 구성입니다. 이 유형의 HA 클러스터 구성은 가끔 "활성/활성"으로도 참조되어 모든 노드가 중요한 워크로드를 활발하게 처리함을 표시합니다.

확장된 대기 구성 또는 인수 구성 중 하나에서 다른 노드의 작업을 인수할 수 있는 노드에 배치될 수 있는 피크 로드를 고려하는 것은 중요합니다. 그러한 노드는 허용 가능한 수준의 성능을 유지하기 위해 충분한 용량을 소유해야 합니다.

큐 관리자 클러스터에 대한 HA 클러스터의 관계

큐 관리자 클러스터는 관리 작업을 줄이고 큐 관리자 클러스터 큐의 인스턴스 전체에서 메시지의 로드 밸런싱을 제공합니다. 이는 또한 단일 큐 관리자보다 높은 가용성을 제공합니다. 큐 관리자의 실패 후에도 메시징 애플리케이션이 큐 관리자 클러스터 큐의 남아 있는 인스턴스에 대해 계속해서 액세스할 수 있기 때문입니다. 그러나 큐 관리자 클러스터만으로는 큐 관리자 실패의 자동 감지 및 큐 관리자 재시작 또는 장애 복구의 자동 트리거를 제공하지 않습니다. HA 클러스터는 이러한 기능을 제공합니다. 좋은 효과를 얻기 위해 두 유형의 클러스터를 함께 사용할 수 있습니다.

IBM MQ 클라이언트

재시작 또는 인수의 대상일 수 있는 큐 관리자와 통신하는 IBM MQ 클라이언트를 작성하여 끊긴 연결을 허용하고 다시 연결을 반복적으로 시도해야 합니다. IBM WebSphere MQ 7에서는 CCDT(Client Channel Definition Table)의 처리에서 연결 가용성 및 워크로드 밸런싱을 지원하는 기능을 도입했지만 이러한 기능은 장애 복구 시스템에서 작업할 때 직접적으로 관련되지는 않습니다.

트랜잭션 가능한 기능을 사용하면 IBM MQ MQI client가 같은 큐 관리자에 연결되는 한 2단계 트랜잭션에 참여할 수 있습니다. 트랜잭션 가능한 기능은 큐 관리자 목록에서 선택하기 위해 IP 로드 밸런서와 같은 기술을 사용할 수 없습니다. HA 제품을 사용하는 경우 큐 관리자는 실행하는 노드에 상관없이 그 ID(이름 및 주소)를 유지보수하므로 트랜잭션 가능한 기능은 HA 제어를 받는 큐 관리자에서 사용할 수 있습니다.

HA 클러스터에서 작동하는 IBM MQ

모든 HA 클러스터에는 장애 복구 단위의 개념이 있습니다. 이는 고가용성 서비스를 구성하는 모든 자원을 포함하는 일련의 정의입니다. 장애 복구 단위는 서비스 자체 및 서비스가 의존하는 다른 모든 자원을 포함합니다.

HA 솔루션은 장애 복구 단위에 다른 용어를 사용합니다.

- AIX용 PowerHA에서는 장애 복구 단위를 자원 그룹이라고 합니다.
- Veritas Cluster Server에서는 이를 서비스 그룹이라고 합니다.
- Serviceguard에서는 패키지라고 합니다.

이 주제에서는 장애 복구 단위를 의미하기 위해 자원 그룹이라는 용어를 사용합니다.

IBM MQ에 대한 가장 작은 장애 복구 단위는 큐 관리자입니다. 일반적으로 큐 관리자를 포함하는 자원 그룹은 자원 그룹에서 사용하도록 독점 예약된 볼륨 그룹 또는 디스크 그룹의 공유 디스크 및 큐 관리자에 연결하는 데 사용되는 IP 주소도 포함합니다. 같은 자원 그룹에 있는 리스너 또는 트리거 모니터와 같은 다른 IBM MQ 자원을 개별 자원으로 또는 큐 관리자 자체의 제어를 받는 자원으로 포함시킬 수도 있습니다.

HA 클러스터에서 사용되는 큐 관리자는 클러스터의 노드 사이에서 공유되는 디스크에 그 데이터 및 로그가 있어야 합니다. HA 클러스터는 한 번에 클러스터의 한 노드만 디스크에 쓸 수 있도록 합니다. HA 클러스터는 모니터 스크립트를 사용하여 큐 관리자의 상태를 모니터할 수 있습니다.

큐 관리자와 관련된 데이터 및 로그 둘 다에 대해 단일 공유 디스크를 사용할 수 있습니다. 그러나 개별적으로 크기를 지정하고 조정할 수 있도록 개별 공유 파일 시스템을 사용하는 것이 일반적인 사례입니다.

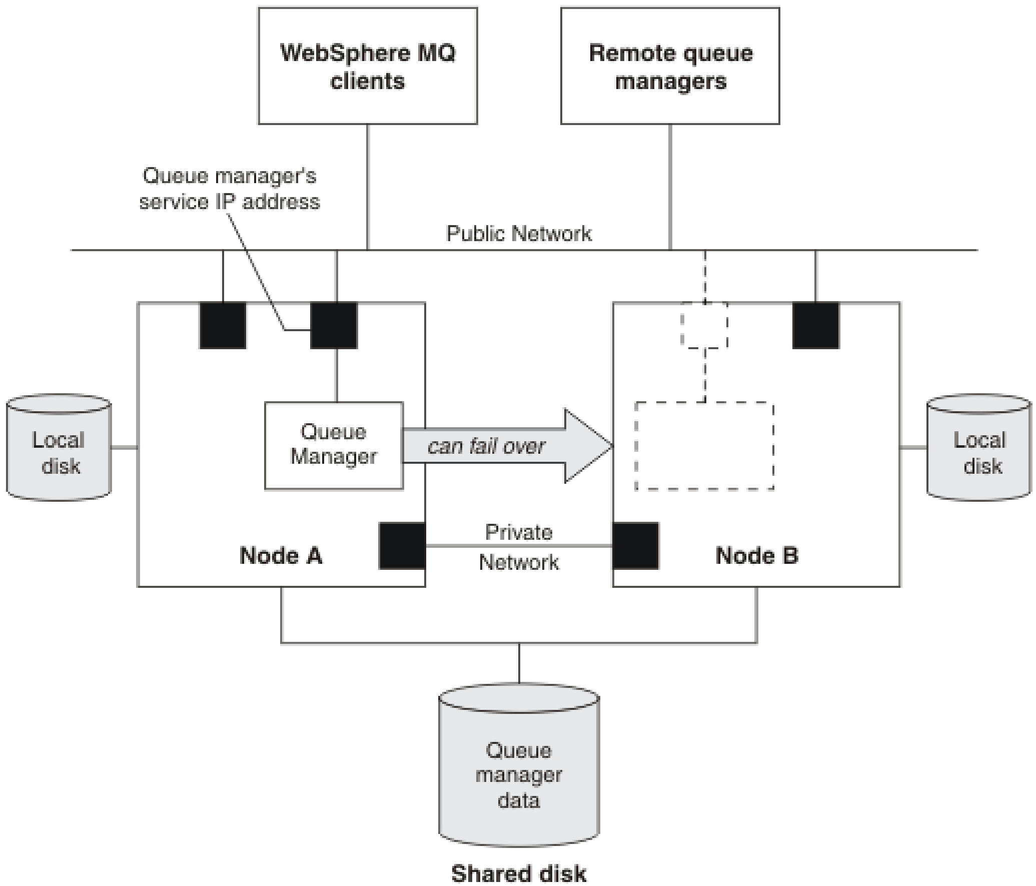


그림 71. *HA 클러스터(HA cluster)*

그림 1은 두 개의 노드가 있는 HA 클러스터를 보여줍니다. HA 클러스터는 자원 그룹에서 정의된 큐 관리자의 가용성을 관리합니다. 이는 활성/수동 또는 콜드 대기 구성입니다. 한 개의 노드인 노드 A가 현재 큐 관리자를 실행하고 있기 때문입니다. 이 큐 관리자는 공유 디스크에 있는 데이터와 로그 파일로 작성되었습니다. 큐 관리자에는 HA 클러스터가 관리하는 서비스 IP 주소가 있습니다. 큐 관리자는 공유 디스크 및 그 서비스 IP 주소에 따라 달라집니다. HA 클러스터가 노드 A에서 노드 B까지 큐 관리자의 장애를 복구하는 경우 먼저 큐 관리자의 종속 자원을 노드 B로 이동한 다음 큐 관리자를 시작합니다.

HA 클러스터에 둘 이상의 큐 관리자가 있을 경우 사용자의 HA 클러스터 구성으로 인해 장애 복구 후 동일 노드에서 둘 이상의 큐 관리자가 실행될 수 있습니다. HA 클러스터의 각 큐 관리자에는 큐 관리자가 특정 시간에 활성이 될 수도 있는 클러스터 노드에서 사용하는 자체 포트 번호가 지정되어야 합니다.

일반적으로 HA 클러스터는 루트 사용자로 실행됩니다. IBM MQ는 mqm 사용자로 실행됩니다. IBM MQ의 관리는 mqm 그룹의 멤버에 부여됩니다. mqm 사용자 및 그룹 둘 다 모든 HA 클러스터 노드에 존재해야 합니다. 사용자 ID 및 그룹 ID는 클러스터 전체에서 일관되어야 합니다. 루트 사용자에 의한 IBM MQ의 관리는 허용되지 않으며 스크립트를 시작, 중지 또는 모니터하는 스크립트는 mqm 사용자로 전환되어야 합니다.

참고: IBM MQ는 모든 노드에 올바르게 설치되어야 하며 제품 실행 파일을 공유할 수 없습니다.

Linux UNIX *UNIX and Linux*에서 공유 디스크 구성

HA 클러스터의 IBM MQ 큐 관리자에서는 데이터 파일 및 로그 파일이 공유 디스크의 공용 이름 지정된 원격 파일 시스템에 있어야 합니다.

이 태스크 정보

그림 1은 HA 클러스터에서 큐 관리자에 대해 가능한 레이아웃을 표시합니다. 큐 관리자의 데이터 및 로그 디렉토리는 둘 다 /MQHA/QM1에 마운트되는 공유 디스크에 있습니다. 이 디스크는 장애 복구가 발생하면 큐 관리자가 재시작되는 위치에 관계없이 데이터가 사용 가능하도록 HA 클러스터의 노드 사이에서 전환됩니다. `mqs.ini` 파일에는 QM1 큐 관리자에 대한 스탠자가 있습니다. `qm.ini` 파일의 Log 스탠자에는 LogPath의 값이 있습니다.

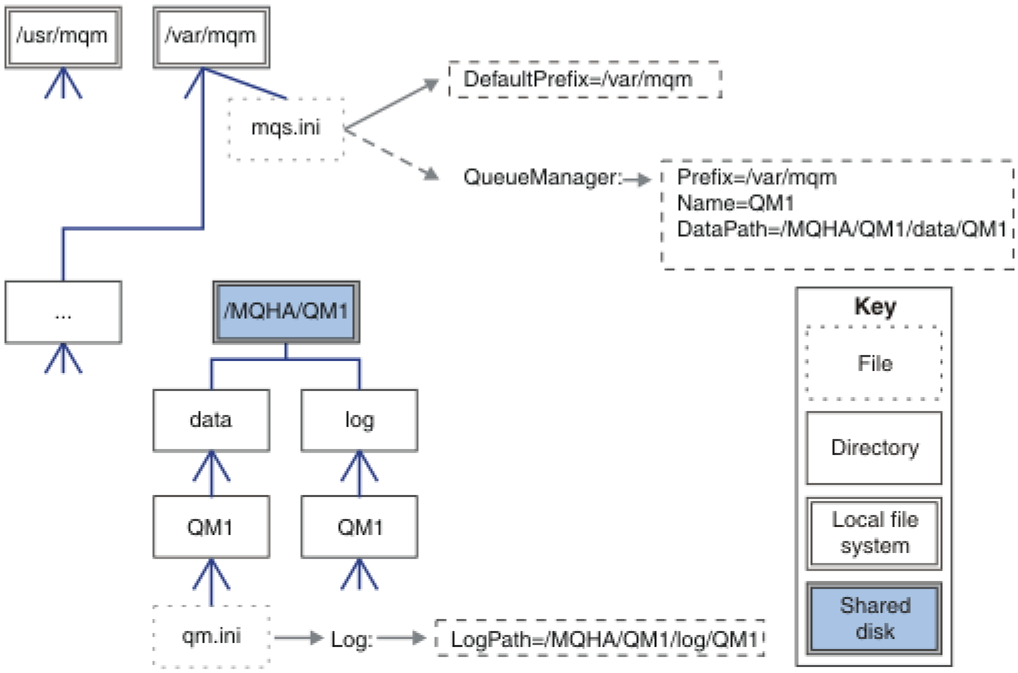


그림 72. 공유 이름 지정된 *data* 및 *log* 디렉토리

프로시저

1. 큐 관리자의 파일 시스템의 마운트 지점 이름을 결정하십시오.

   예를 들어, 큐 관리자 데이터 파일의 경우 `/MQHA/qmgrname/data`이고 해당 로그 파일의 경우 `/MQHA/qmgrname/log`입니다.
2. 큐 관리자의 데이터 및 로그 파일이 포함될 볼륨 그룹(또는 디스크 그룹)을 작성하십시오.

   이 볼륨 그룹은 동일한 자원 그룹의 고가용성(HA) 클러스터에 의해 큐 관리자로 관리됩니다.
3. 볼륨 그룹의 로그 파일과 큐 관리자의 데이터를 위한 파일 시스템을 작성하십시오.
4. 각 노드에 대해 차례대로 파일 시스템의 마운트 지점을 작성하고 파일 시스템을 마운트할 수 있는지 확인하십시오.

   mqm 사용자는 마운트 지점을 소유해야 합니다.

Linux UNIX *UNIX and Linux*에서 *HA* 클러스터 큐 관리자 작성

고가용성 클러스터에서 큐 관리자를 사용하기 위한 첫 번째 단계는 노드 중 하나에서 큐 관리자를 작성하는 것입니다.

이 태스크 정보

HA 클러스터에서 사용할 큐 관리자를 작성하려면 큐 관리자를 작성할 클러스터에 있는 노드 중에서 하나를 선택한 다음, 이 노드에서 다음 단계를 수행하십시오.

프로시저

1. 큐 관리자의 파일 시스템을 노드에 마운트하십시오.
2. **crtmqm** 명령을 사용하여 큐 관리자를 작성하십시오.

예를 들면, 다음과 같습니다.

```
crtmqm -md /MQHA/qmgrname/data -ld /MQHA/qmgrname/log qmgrname
```

3. **strmqm** 명령을 사용하여 큐 관리자를 수동으로 시작하십시오.
4. 큐 및 채널 작성, 큐 관리자를 시작할 때 리스너가 자동으로 시작되도록 큐 관리자 설정과 같은 큐 관리자의 초기 구성을 완료하십시오.
5. **endmqm** 명령을 사용하여 큐 관리자를 중지하십시오.
6. **dspmqinf** 명령을 사용하여 **addmqinf** 명령을 표시하십시오.

   ```
   dspmqinf -o command qmgrname
   ```

   여기서, qmgrname은 큐 관리자의 이름입니다.

   addmqinf 명령 사용에 대한 자세한 정보는 376 페이지의 『UNIX and Linux에서 다른 HA 클러스터 노드에 큐 관리자 구성 추가』의 내용을 참조하십시오.

   addmqinf 명령은 다음 예제와 유사한 방식으로 표시됩니다.

   ```
   addmqinf -sQueueManager -vName=qmgrname -vDirectory=qmgrname \
   -vPrefix=/var/mqm -vDataPath=/MQHA/qmgrname/data/qmgrname
   ```
7. 표시되는 명령을 주의하여 기록해 두십시오.
8. 큐 관리자의 파일 시스템을 마운트 취소하십시오.

다음에 수행할 작업

이제 376 페이지의 『UNIX and Linux에서 다른 HA 클러스터 노드에 큐 관리자 구성 추가』에 설명된 단계를 완료할 준비가 되었습니다.

Linux UNIX *UNIX and Linux*에서 다른 *HA* 클러스터 노드에 큐 관리자 구성 추가

큐 관리자 구성 정보를 HA 클러스터에 있는 다른 노드에 추가해야 합니다.

시작하기 전에

이 태스크를 완료하기 전에 375 페이지의 『UNIX and Linux에서 HA 클러스터 큐 관리자 작성』의 단계를 이미 완료한 상태여야 합니다. 큐 관리자를 작성한 후 각각의 다른 노드에서 다음 단계를 수행하여 HA 클러스터의 다른 노드에 큐 관리자에 대한 구성 정보를 추가해야 합니다.

이 태스크 정보

HA 클러스터에서 사용할 큐 관리자를 작성할 경우, 375 페이지의 『UNIX and Linux에서 HA 클러스터 큐 관리자 작성』에서 설명한대로 먼저 큐 관리자를 작성할 클러스터에 있는 노드 중에서 하나를 선택해야 합니다.

프로시저

1. 큐 관리자 파일 시스템을 마운트하십시오.
2. 큐 관리자 구성 정보를 노드에 추가하십시오.

   채널은 다음 두 가지 방법으로 설정할 수 있습니다.

   - `/var/mqm/mqs.ini`을 직접 편집합니다.
   - 375 페이지의 『UNIX and Linux에서 HA 클러스터 큐 관리자 작성』의 6단계에 있는 **dspmqinf** 명령에서 표시한 **addmqinf** 명령을 실행합니다.
3. 큐 관리자를 시작 및 중지하여 구성을 확인하십시오.

   큐 관리자를 시작하고 중지하는 데 사용되는 명령은 **addmqinf** 명령과 같은 IBM MQ 설치에서 발행되어야 합니다. 현재 큐 관리자와 연관된 다른 설치에서 큐 관리자를 시작 및 중지하려면 먼저 **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 설정해야 합니다. 자세한 정보는 setmqm을 참조하십시오.
4. 큐 관리자 파일 시스템을 마운트 취소하십시오.

Linux UNIX UNIX and Linux에서 HA 클러스터 큐 관리자 시작

큐 관리자는 HA 클러스터에서 자원으로 표시됩니다. HA 클러스터는 큐 관리자를 시작하고 중지할 수 있어야 합니다. 대부분의 경우 쉘 스크립트를 사용하여 큐 관리자를 시작할 수 있습니다. 네트워크 파일 시스템을 사용하거나 각 로컬 디스크에 복사하여 클러스터에 있는 모든 노드의 같은 위치에서 이러한 스크립트가 사용 가능하게 해야 합니다.

참고: 실패한 큐 관리자를 재시작하기 전에 큐 관리자의 해당 인스턴스로부터 애플리케이션의 연결을 끊어야 합니다. 이를 수행하지 않으면 큐 관리자가 올바르게 재시작하지 않을 수 있습니다.

적당한 쉘 스크립트의 예가 여기에 제공됩니다. 이러한 쉘 스크립트를 필요에 맞게 조정하여 HA 클러스터의 제어 하에 큐 관리자를 시작하는 데 사용할 수 있습니다.

다음 쉘 스크립트는 큐 관리자가 성공적으로 시작할 수 있도록 HA 클러스터 사용자에서 mqm 사용자로 전환하는 방법의 예입니다.

```
#!/bin/ksh

# A simple wrapper script to switch to the mqm user.

su mqm -c name_of_your_script $*
```

다음 쉘 스크립트는 큐 관리자의 현재 상태에 대한 어떠한 가정을 하지 않고도 큐 관리자를 시작하는 방법의 예입니다. 이 쉘 스크립트는 큐 관리자에 속한 프로세스를 종료하는 데 매우 돌발적인 메소드를 사용함을 참고하십시오.

```
#!/bin/ksh
#
# This script robustly starts the queue manager.
#
# The script must be run by the mqm user.

# The only argument is the queue manager name. Save it as QM variable
QM=$1

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
fi

# End any queue manager processes which might be running.

srchstr="( |-m)$QM *.*$"
for process in amqzmuc0 amqzxma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
               amqzfuma amqzmuf0 amqzmur0 amqzmgr0
 do
  ps -ef | tr "\t" " " | grep $process | grep -v grep | \
    egrep "$srchstr" | awk '{print $2}'| \
       xargs kill -9 > /dev/null 2>&1
done

# It is now safe to start the queue manager.
# The strmqm command does not use the -x flag.
strmqm ${QM}
```

다른 관련된 프로그램을 시작하도록 스크립트를 수정할 수 있습니다.

Linux UNIX UNIX and Linux에서 HA 클러스터 큐 관리자 중지

대부분의 경우 쉘 스크립트를 사용하여 큐 관리자를 중지할 수 있습니다. 적당한 쉘 스크립트의 예가 여기에 제공됩니다. 이러한 쉘 스크립트를 필요에 맞게 조정하여 HA 클러스터의 제어 하에 큐 관리자를 중지하는 데 사용할 수 있습니다.

다음 스크립트는 큐 관리자의 현재 상태에 대한 어떠한 가정을 하지 않고도 즉시 중지하는 방법의 예입니다. 스크립트는 mqm 사용자가 실행해야 합니다. 그러므로 HA 클러스터 사용자에서 mqm 사용자로 전환하려면 쉘 스크

립트에 있는 이 스크립트를 랩핑해야 합니다. (쉘 스크립트 예제는 377 페이지의 『UNIX and Linux에서 HA 클러스터 큐 관리자 시작』에서 제공됩니다.)

```
#!/bin/ksh
#
# The script ends the QM by using two phases, initially trying an immediate
# end with a time-out and escalating to a forced stop of remaining
# processes.
#
# The script must be run by the mqm user.
#
# There are two arguments: the queue manager name and a timeout value.
QM=$1
TIMEOUT=$2

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
fi

if [ -z "$TIMEOUT" ]
then
  echo "ERROR! No timeout specified"
  exit 1
fi

for severity in immediate brutal
do
  # End the queue manager in the background to avoid
  # it blocking indefinitely. Run the TIMEOUT timer
  # at the same time to interrupt the attempt, and try a
  # more forceful version. If the brutal version fails,
  # nothing more can be done here.

  echo "Attempting ${severity} end of queue manager '${QM}'"
  case $severity in

  immediate)
    # Minimum severity of endmqm is immediate which severs connections.
    # HA cluster should not be delayed by clients
    endmqm -i ${QM} &
    ;;

  brutal)
    # This is a forced means of stopping queue manager processes.

    srchstr="( |-m)$QM *.*$"
    for process in amqzmuc0 amqzxma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
               amqzfuma amqzmuf0 amqzmur0 amqzmgr0
    do
      ps -ef | tr "\t" " " | grep $process | grep -v grep | \
         egrep "$srchstr" | awk '{print $2}'| \
            xargs kill -9 > /dev/null 2>&1
    done

  esac

  TIMED_OUT=yes
  SECONDS=0
  while (( $SECONDS < ${TIMEOUT} ))
  do
   TIMED_OUT=yes
   i=0
   while [ $i -lt 5 ]
   do
     # Check for execution controller termination
     srchstr="( |-m)$QM *.*$"
     cnt=`ps -ef | tr "\t" " " | grep amqzxma0 | grep -v grep | \
       egrep "$srchstr" | awk '{print $2}' | wc -l `
     i=`expr $i + 1`
     sleep 1
     if [ $cnt -eq 0 ]
     then
       TIMED_OUT=no
       break
     fi
   done
```

```
    if [ ${TIMED_OUT} = "no" ]
    then
      break
    fi

    echo "Waiting for ${severity} end of queue manager '${QM}'"
    sleep 1
  done # timeout loop

  if [ ${TIMED_OUT} = "yes" ]
  then
    continue        # to next level of urgency
  else
    break           # queue manager is ended, job is done
  fi

done # next phase
```

참고: 특정 큐 관리자에 대해 실행되는 프로세스에 따라 이 스크립트에 포함되는 큐 관리자 프로세스 목록이 완벽한 목록이 아니거나 해당 큐 관리자에 대해 실행 중인 프로세스보다 더 많은 프로세스가 포함될 수 있습니다.

```
for process in amqzmuc0 amqzxma0 amqfcxba amqfqpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmfa amqrmppa \
               amqzfuma amqzmuf0 amqzmur0 amqzmgr0
```

구성되는 기능 및 특정 큐 관리자에 대해 실행 중인 프로세스에 따라 프로세스가 목록에 포함되거나 제외될 수 있습니다. 프로세스의 전체 목록과 특정 순서로 프로세스를 중지하는 것에 대한 정보는 UNIX 및 Linux에서 큐 관리자 수동 중지를 참조하십시오.

Linux UNIX *UNIX and Linux*에서 *HA* 클러스터 큐 관리자 모니터링

일반적으로 고가용성(HA) 클러스터가 큐 관리자의 상태를 주기적으로 모니터하는 방법이 제공됩니다. 대부분의 경우 이에 쉘 스크립트를 사용할 수 있습니다. 적당한 쉘 스크립트의 예가 여기에 제공됩니다. 이러한 스크립트를 필요에 맞게 조정하여 사용자 환경에 맞는 추가 모니터링 검사를 수행하는 데 사용할 수 있습니다.

IBM WebSphere MQ 7.1에서 한 시스템에 공존하는 다중 IBM MQ 설치가 있을 수 있습니다. 다중 설치에 대한 자세한 정보는 다중 설치를 참조하십시오. IBM WebSphere MQ 7.1 이상에서의 설치를 포함한 다중 설치 전체에서 모니터링 스크립트를 사용하려는 경우 몇 가지 추가 단계를 수행해야 합니다. 기본 설치가 있거나 이 스크립트를 IBM WebSphere MQ 7.1 이전 버전에서 사용하는 경우 스크립트를 사용하기 위해 *MQ_INSTALLATION_PATH*를 지정할 필요는 없습니다. 그렇지 않으면 다음 단계에서 *MQ_INSTALLATION_PATH*가 올바르게 식별되는지 확인하십시오.

1. IBM WebSphere MQ 7.1 설치에서 **crtmqenv** 명령을 사용하여 큐 관리자의 정확한 *MQ_INSTALLATION_PATH*를 식별하십시오.

   ```
   crtmqenv -m qmname
   ```

   이 명령은 *qmname*으로 지정되는 큐 관리자에 대해 올바른 *MQ_INSTALLATION_PATH* 값을 리턴합니다.
2. 적절한 *qmname* 및 *MQ_INSTALLATION_PATH* 매개변수를 사용하여 모니터링 스크립트를 실행하십시오.

참고: AIX용 PowerHA는 큐 관리자의 모니터링 프로그램에 매개변수를 제공하는 방법을 제공하지 않습니다. 각 큐 관리자에 대해 큐 관리자 이름을 캡슐화하는 개별 모니터링 프로그램을 작성해야 합니다. 다음은 큐 관리자 이름을 캡슐화하기 위해 AIX에서 사용되는 스크립트의 예입니다.

```
#!/bin/ksh
su mqm -c name_of_monitoring_script qmname  MQ_INSTALLATION_PATH
```

여기서, *MQ_INSTALLATION_PATH*는 큐 관리자 *qmname*이 연관되는 IBM MQ 설치 경로를 지정하는 선택적 매개변수입니다.

다음 스크립트는 **runmqsc**가 정지될 가능성에 대해 확신하지 않습니다. 일반적으로 HA 클러스터는 정지 모니터링 스크립트를 실패로 처리하고 스스로 이 가능성에 대해 확신합니다.

그러나 이 스크립트는 큐 관리자가 시작 상태인 것은 허용합니다. 이는 HA 클러스터가 큐 관리자가 시작되자마자 큐 관리자를 모니터링하기 시작하는 것이 일반적이기 때문입니다. 일부 HA 클러스터는 시작 단계와 자원에

대한 실행 단계를 구별하지만 시작 단계의 지속 기간을 구성해야 합니다. 큐 관리자를 시작하는 데 걸리는 시간은 수행해야 하는 작업의 양에 따라 다르기 때문에 큐 관리자를 시작하는 데 걸리는 최대 시간을 선택하기는 어렵습니다. 너무 낮은 값을 선택하면 HA 클러스터는 큐 관리자가 시작을 완료하지 않은 경우에 실패한 것으로 잘못 가정합니다. 이로 인해 장애 복구가 끝없이 연속되는 결과가 나타날 수 있습니다.

이 스크립트는 mqm 사용자에 의해 실행되어야 하므로 사용자를 HA 클러스터에서 mqm으로 전환하기 위해 쉘 스크립트에 이 스크립트를 랩핑해야 할 수도 있습니다(쉘 스크립트 예는 377 페이지의 『UNIX and Linux에서 HA 클러스터 큐 관리자 시작』에 제공되어 있음).

```
#!/bin/ksh
#
# This script tests the operation of the queue manager.
#
# An exit code is generated by the runmqsc command:
# 0  => Either the queue manager is starting or the queue manager is running and responds.
#       Either is OK.
# >0 => The queue manager is not responding and not starting.
#
# This script must be run by the mqm user.
QM=$1
MQ_INSTALLATION_PATH=$2

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
fi

if [ -z "$MQ_INSTALLATION_PATH" ]
then
  # No path specified, assume system primary install or MQ level < 7.1.0.0
  echo "INFO: Using shell default value for MQ_INSTALLATION_PATH"
else
  echo "INFO: Prefixing shell PATH variable with $MQ_INSTALLATION_PATH/bin"
  PATH=$MQ_INSTALLATION_PATH/bin:$PATH
fi

# Test the operation of the queue manager. Result is 0 on success, non-zero on error.
echo "ping qmgr" | runmqsc ${QM} > /dev/null 2>&1
pingresult=$?

if [ $pingresult -eq 0 ]
then # ping succeeded

  echo "Queue manager '${QM}' is responsive"
  result=0

else # ping failed

  # Don't condemn the queue manager immediately, it might be starting.
  srchstr="( |-m)$QM *.*$"
  cnt=`ps -ef | tr "\t" " " | grep strmqm | grep "$srchstr" | grep -v grep \
              | awk '{print $2}' | wc -l`
  if [ $cnt -gt 0 ]
  then
    # It appears that the queue manager is still starting up, tolerate
    echo "Queue manager '${QM}' is starting"
    result=0
  else
    # There is no sign of the queue manager starting
    echo "Queue manager '${QM}' is not responsive"
    result=$pingresult
  fi

fi

exit $result
```

Linux UNIX *UNIX and Linux*에서 큐 관리자를 *HA* 클러스터 제어 하에 넣기

큐 관리자 IP 주소 및 공유 디스크를 사용하여 HA 클러스터의 제어 아래 큐 관리자를 구성해야 합니다.

이 태스크 정보

큐 관리자를 HA 클러스터의 제어 아래에 배치하려면 큐 관리자 및 모든 연관된 자원을 포함하는 자원 그룹을 정의해야 합니다.

프로시저

1. 큐 관리자, 큐 관리자의 볼륨 또는 디스크 그룹 및 큐 관리자의 IP 주소를 포함하는 자원 그룹을 작성하십시오.

   IP 주소는 컴퓨터의 IP 주소가 아닌 가상 IP 주소입니다.
2. HA 클러스터가 자원을 클러스터 노드 사이에서 올바르게 전환하고 큐 관리자를 제어할 준비가 되었는지 확인하십시오.

Linux UNIX *UNIX and Linux*에서 *HA* 클러스터 큐 관리자 삭제

더 이상 큐 관리자를 실행하지 않아도 되는 노드에서 큐 관리자를 제거하려고 합니다.

이 태스크 정보

HA 클러스터에 있는 노드에서 큐 관리자를 제거하려면 해당 구성 정보를 제거해야 합니다.

프로시저

1. HA 클러스터가 노드에서 큐 관리자를 활성화하려고 시도하지 않도록 HA 클러스터에서 이 노드를 제거하십시오.
2. 다음 **rmvmqinf** 명령을 사용하여 큐 관리자의 구성 정보를 제거하십시오.

   ```
   rmvmqinf qmgrname
   ```
3. 옵션: 큐 관리자를 완전히 삭제하려면 **dltmqm** 명령을 사용하십시오.

   중요사항: dltmqm 명령을 사용하여 큐 관리자를 삭제하면 큐 관리자의 데이터 및 로그 파일이 완전히 삭제됩니다.

   큐 관리자 삭제를 마쳤으면 **rmvmqinf** 명령을 사용하여 다른 노드에서 남아 있는 구성 정보를 제거할 수 있습니다.

Windows MSCS(Microsoft Cluster Service) 지원

가상 서버의 장애 복구를 지원하는 MSCS를 소개하고 설정합니다.

이 정보는 IBM MQ for Windows에만 적용됩니다.

MSCS(Microsoft Cluster Service)를 사용하면 서버를 클러스터에 연결하여 데이터 및 애플리케이션의 고가용성을 제공하고 시스템 관리를 용이하게 할 수 있습니다. MSCS는 서버 또는 애플리케이션 실패를 자동으로 감지하고 복구할 수 있습니다.

MSCS는 애플리케이션, 웹 사이트, 인쇄 큐 또는 파일 공유에 해당하는 가상 서버의 장애 복구를 지원합니다(예: 해당 디스크 스핀들, 파일 및 IP 주소 등).

장애 복구는 MSCS가 클러스터의 한 컴퓨터에서 애플리케이션의 실패를 감지하고 순서대로 문제가 발생한 애플리케이션을 종료하며 해당 상태 데이터를 다른 컴퓨터에 전송하고 그 위치에서 애플리케이션을 다시 시작하는 데 사용하는 프로세스입니다.

이 절에서는 MSCS 클러스터를 소개하고 다음 절에서는 MSCS 지원 설정에 대해 설명합니다.

- 382 페이지의 『MSCS 클러스터 소개』
- 383 페이지의 『MSCS 클러스터링을 위한 IBM MQ 설정』

다음 절에 MSCS 클러스터링을 위한 IBM MQ 구성 방법에 대해 설명합니다.

- 384 페이지의 『MSCS에서 사용할 큐 관리자 작성』
- 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』
- 386 페이지의 『MSCS 제어 아래 큐 관리자 넣기』

• 392 페이지의 『MSCS 제어에서 큐 관리자 제거』

그런 다음 다음 절에 IBM MQ에서의 MSCS 사용에 대한 몇 가지 유용한 힌트를 제공하고 IBM MQ MSCS 지원 유틸리티 프로그램에 대한 세부사항을 설명합니다.

• 393 페이지의 『MSCS 사용의 힌트 및 팁』
• 395 페이지의 『MSCS 유틸리티 프로그램에 대한 지원』

Windows *MSCS* 클러스터 소개

MSCS 클러스터는 둘 이상의 컴퓨터로 구성된 그룹이며 이 두 컴퓨터는 서로 연결되어 하나가 실패하면 MSCS가 애플리케이션의 상태 데이터를 실패한 컴퓨터에서 클러스터 내 다른 컴퓨터로 전송하고 해당 조작을 그 위치에서 다시 시작하여 장애 복구를 수행하는 방식으로 구성됩니다.

371 페이지의 『고가용성 구성』에는 MSCS 클러스터, 다중 인스턴스 큐 관리자 및 IBM MQ 클러스터 간 비교가 포함됩니다.

이 절 및 하위 주제에서 용어 클러스터는 단독으로 사용될 때 **항상** MSCS 클러스터를 의미합니다. 이는 이 안내서 다른 위치에 설명되어 있는 IBM MQ 클러스터와는 구분됩니다.

두 시스템 클러스터는 가상 *IP* 주소를 사용한 클라이언트 액세스를 위해 네트워크에 함께 연결된 두 대의 컴퓨터(예: A 및 B)로 구성됩니다. 이 두 컴퓨터는 하나 이상의 사설 네트워크를 통해 서로 연결될 수도 있습니다. A와 B는 각 서버 애플리케이션이 사용할 디스크를 하나 이상 공유합니다. 독점적 MSCS 사용을 위해 독립 디스크(*RAID*) 레벨 1의 중복 배열이어야 하는 다른 공유 디스크도 있으며 쿼럼 디스크라고 합니다. MSCS는 두 컴퓨터를 모두 모니터하여 하드웨어 및 소프트웨어가 올바르게 실행되고 있는지 검사합니다.

이와 같은 단순 설정에서 두 컴퓨터 모두에는 애플리케이션이 설치되어 있지만 컴퓨터 A만 라이브 애플리케이션과 함께 실행되고 컴퓨터 B는 실행 및 대기만 합니다. 컴퓨터 A에서 다양한 문제점 중 하나가 발견되면 MSCS가 문제점이 발생한 애플리케이션을 순서대로 종료하고 그 상태 데이터를 다른 컴퓨터에 전송하며 그 위치에서 애플리케이션을 다시 시작합니다. 이를 장애 복구라고 합니다. 애플리케이션이 MSCS와 완전히 상호작용하고 원활하게 장애 복구할 수 있도록 애플리케이션을 클러스터 인지로 설정할 수 있습니다.

두 컴퓨터 클러스터에 대한 일반적인 설정은 382 페이지의 그림 73에 표시된 대로입니다.

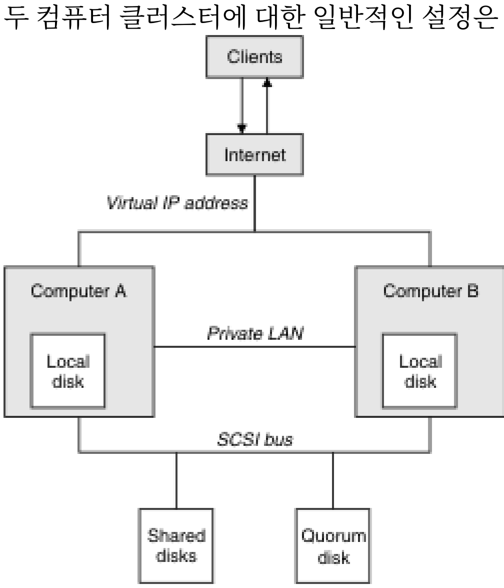


그림 *73.* 두 컴퓨터 *MSCS* 클러스터

각 컴퓨터는 공유 디스크에 액세스할 수 있지만 MSCS의 제어 하에 한 번에 하나씩만 가능합니다. 장애 복구 시 MSCS는 액세스를 다른 컴퓨터로 전환합니다. 공유 디스크 자체는 일반적으로 RAID지만 반드시 RAID일 필요는 없습니다.

각 컴퓨터는 클라이언트 액세스를 위해 외부 네트워크에 연결되고 각각 IP 주소를 가지고 있습니다. 그러나 이 클러스터와 통신하는 외부 클라이언트는 한 개의 가상 *IP* 주소만 인식하고 MSCS는 클러스터 내 IP 트래픽을 적절하게 라우트합니다.

MSCS는 예를 들어, 하트비트를 사용하여 상태를 모니터하고 해당 데이터베이스를 동기화하기 위해 하나 이상의 개인용 연결 또는 공용 네트워크를 통해 두 컴퓨터 간 자체 통신도 수행합니다.

Windows *MSCS* 클러스터링을 위한 *IBM MQ* 설정

큐 관리자를 MSCS에 대한 장애 복구 단위로 설정하여 클러스터링을 위한 IBM MQ를 구성합니다. 큐 관리자를 MSCS에 대한 자원으로 정의하면 큐 관리자를 모니터하고 문제점이 있을 경우 클러스터 내 다른 컴퓨터로 전송할 수 있습니다.

이를 위해 시스템을 설정하려면 클러스터 내 각 컴퓨터에서 IBM MQ를 설치하여 시작합니다.

큐 관리자는 IBM MQ 설치 이름과 연관되므로 클러스터 내 모든 컴퓨터에서 IBM MQ 설치 이름은 같아야 합니다. 설치 및 설치 제거를 참조하십시오.

큐 관리자 스스로는 큐 관리자를 작성하는 컴퓨터에만 존재해야 합니다. 장애 복구 시 MSCS는 다른 컴퓨터에서 큐 관리자를 시작합니다. 그러나 큐 관리자는 로컬 드라이브가 아닌 클러스터 공유 디스크에 로그 및 데이터 파일을 가져야 합니다. 이미 로컬 드라이브에 설치된 큐 관리자가 있는 경우 IBM MQ에서 제공된 도구를 사용하여 이 큐 관리자를 마이그레이션할 수 있습니다. 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』의 내용을 참조하십시오. MSCS에서 사용할 새 큐 관리자를 작성하려는 경우 384 페이지의 『MSCS에서 사용할 큐 관리자 작성』의 내용을 참조하십시오.

설치 및 마이그레이션 후 MSCS 클러스터 관리자를 사용하여 MSCS가 큐 관리자를 인지하도록 설정하십시오. 386 페이지의 『MSCS 제어 아래 큐 관리자 넣기』의 내용을 참조하십시오.

MSCS 제어에서 큐 관리자를 제거하기로 결정한 경우 392 페이지의 『MSCS 제어에서 큐 관리자 제거』에 설명되어 있는 프로시저를 사용하십시오.

Windows 대칭 및 *MSCS* 설정

애플리케이션이 한 노드에서 다른 노드로 전환할 때 노드에 관계없이 같은 방식으로 작동해야 합니다. 이를 보장하는 최고의 방법은 환경을 동일하게 설정하는 것입니다.

가능한 경우 각 컴퓨터에서 동일한 하드웨어, 운영 체제 소프트웨어, 제품 소프트웨어 및 구성으로 클러스터를 설정하십시오. 특히, 두 컴퓨터에 설치된 모든 필수 소프트웨어가 버전, 유지보수 레벨, SupportPacs, 경로 및 엑시트 면에서 동일한지 및 383 페이지의 『MSCS 보안』에 설명된 대로 공통 네임스페이스(보안 환경)가 있는지 확인하십시오.

Windows *MSCS* 보안

성공적인 MSCS 보안을 위해 다음 가이드라인을 따릅니다.

가이드라인은 다음과 같습니다.

- 클러스터 내 각 컴퓨터의 소프트웨어 설치가 동일한지 확인하십시오.
- 클러스터 전체에서 공통 네임스페이스(보안 환경)를 작성하십시오.
- 클러스터 소유자인 사용자 계정이 도메인 계정인 도메인의 MSCS 클러스터 멤버 노드를 작성하십시오.
- 클러스터의 다른 사용자 계정도 도메인 계정으로 설정하여 두 노드 모두에서 사용 가능하도록 합니다. 이는 자동으로 이미 도메인이 있는 경우이고 IBM MQ에 관련된 계정이 도메인 계정입니다. 현재 도메인이 없는 경우 미니 도메인을 설정하여 클러스터 노드 및 관련 계정을 CATER하는 것을 고려해 보십시오. 목적은 두 컴퓨터의 클러스터가 단일 컴퓨팅 자원처럼 보이게 하는 것입니다.

  한 컴퓨터에 로컬인 계정은 다른 컴퓨터에서는 존재하지 않음에 유의하십시오. 다른 컴퓨터에서 같은 이름을 사용하여 계정을 작성하더라도 그 보안 ID(SID)는 다르므로 애플리케이션이 다른 노드로 이동하면 권한이 해당 노드에 존재하지 않습니다.

장애 복구 또는 이동 중에 IBM MQ MSCS 지원은 큐 관리자 오브젝트를 포함하는 모든 파일이 목적지 노드에서 동등한 권한을 가지도록 보장합니다. 명시적으로 이 노드는 관리자 및 mqm 그룹과 시스템 SYSTEM 계정에 완전한 제어가 있는지 및 Everyone에 이전 노드에 대한 읽기 액세스가 있는지, 권한이 목적지 노드에서 추가되는지 검사합니다.

도메인 계정을 사용하여 IBM MQ 서비스를 실행할 수 있습니다. 이 계정이 클러스터 내 각 컴퓨터의 로컬 mqm 그룹에 존재하는지 확인하십시오.

Windows *MSCS*에서 다중 큐 관리자 사용

컴퓨터에서 둘 이상의 큐 관리자를 실행하는 경우 이러한 설정 중 하나를 선택할 수 있습니다.

이 설정은 다음과 같습니다.

- 모든 큐 관리자가 단일 그룹에 있습니다. 이 구성에서 큐 관리자에 문제점이 발생하는 경우 그룹의 모든 큐 관리자는 그룹으로서 다른 컴퓨터로 장애 복구합니다.
- 단일 큐 관리자가 각 그룹에 있습니다. 이 구성에서 큐 관리자에 문제점이 발생하는 경우 이 큐 관리자는 다른 큐 관리자에 영향을 주지 않고 다른 컴퓨터로 장애 복구합니다.
- 첫 번째 두 설정의 혼합입니다.

Windows 클러스터 모드 및 *MSCS*

Windows에서 IBM MQ와 함께 클러스터 시스템을 실행할 수 있는 두 개의 모드 활성/수동 또는 활성/활성이 있습니다.

참고: MSCS를 Microsoft Transaction Server(COM+)와 함께 사용하는 경우 활성/활성 모드를 사용할 수 없습니다.

활성/수동 모드

활성/수동 모드에서는 컴퓨터 A에 실행 중인 애플리케이션이 있고 컴퓨터 B가 백업으로, MSCS가 문제점을 감지하는 경우에만 사용됩니다.

이 모드는 공유 디스크에서만 사용할 수 있지만 애플리케이션으로 인해 장애 복구가 발생하는 경우 **모든** 애플리케이션은 그룹으로 전송되어야 합니다(한 번에 한 컴퓨터만 공유 디스크에 액세스할 수 있기 때문).

A를 선호 컴퓨터로 MSCS를 구성할 수 있습니다. 그런 다음 컴퓨터 A가 복구되거나 바뀌고 다시 올바르게 작동하는 경우 MSCS는 이를 감지하고 자동으로 애플리케이션을 컴퓨터 A로 다시 전환합니다.

둘 이상의 큐 관리자를 실행하는 경우 각각에 대해 개별 공유 디스크를 두는 것을 고려해 보십시오. 그런 다음 MSCS에서 각 큐 관리자를 개별 그룹에 넣으십시오. 이 방법으로 큐 관리자는 다른 큐 관리자에 영향을 주지 않고 다른 컴퓨터로 장애를 복구할 수 있습니다.

활성/활성 모드

활성/활성 모드에서 컴퓨터 A 및 B 둘 다에 실행 중인 애플리케이션이 있고 각 컴퓨터의 그룹은 다른 컴퓨터를 백업으로 사용하도록 설정됩니다. 컴퓨터 A에서 실패가 감지되면 MSCS가 상태 데이터를 컴퓨터 B로 전송하고 그 위치에서 애플리케이션을 다시 시작합니다. 그러면 컴퓨터 B가 자체 애플리케이션과 A의 애플리케이션을 실행합니다.

이 설정을 위해서는 최소 둘 이상의 공유 디스크가 필요합니다. A를 A의 애플리케이션에 대한 선호 컴퓨터로, B를 B의 애플리케이션에 대한 선호 컴퓨터로 MSCS를 구성할 수 있습니다. 장애 복구 및 수리 후에 각 애플리케이션은 자동으로 자체 컴퓨터에서 종료됩니다.

IBM MQ에서 이는 예를 들어, 각각 자체 컴퓨터의 전체 용량을 사용하는 두 큐 관리자를 A 및 B에서 각각 실행할 수 있음을 의미합니다. 컴퓨터 A에서 장애가 발생한 후 두 큐 관리자가 컴퓨터 B에서 실행됩니다. 이는 한 컴퓨터의 전원을 공유하는 것을 의미하며, 속도에서 대량의 데이터를 처리하는 능력이 감소합니다. 그러나 A에서 결함을 찾아 수리하는 동안 중요한 애플리케이션은 계속 사용 가능합니다.

Windows *MSCS*에서 사용할 큐 관리자 작성

이 프로시저는 새 큐 관리자를 MSCS 제어 하에서 준비하고 배치하기에 적합한 방법으로 작성할 수 있도록 합니다.

로컬 드라이브에서 모든 해당 자원을 사용하여 큐 관리자를 작성하여 시작한 다음 로그 파일 및 데이터 파일을 공유 디스크로 마이그레이션합니다. (이 조작을 되돌릴 수 있습니다.) 공유 드라이브에서 해당 자원으로 큐 관리자를 작성하려는 시도를 하지 **마십시오**.

MSCS에서 사용할 큐 관리자를 명령 프롬프트 또는 IBM MQ Explorer의 두 가지 방식으로 작성할 수 있습니다. 명령 프롬프트 사용의 장점은 큐 관리자가 중지됨으로 작성되고 수동 시동으로 설정되는 것이며 이는 MSCS에 대한 준비가 된 것입니다. (IBM MQ Explorer는 자동으로 새 큐 관리자를 시작하고 이를 작성 후 자동 시작으로 설정합니다. 이를 변경해야 합니다.)

명령 프롬프트에서 큐 관리자 작성

다음 단계에 따라 MSCS에서 사용할 큐 관리자를 명령 프롬프트에서 작성하십시오.

1. MQSPREFIX 환경 변수가 로컬 드라이브(예: C:\IBM MQ)를 참조하도록 설정되어 있는지 확인하십시오. 이를 변경하는 경우 시스템 계정이 변경사항을 적용하도록 시스템을 다시 시작하십시오. 이 변수를 설정하지 않으면 큐 관리자가 큐 관리자의 IBM MQ 기본 디렉토리에서 작성됩니다.
2. **crtmqm** 명령을 사용하여 큐 관리자를 작성하십시오. 예를 들어, 기본 디렉토리에서 mscs_test라는 큐 관리자를 작성하려면 다음을 사용하십시오.

```
crtmqm mscs_test
```

3. 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』 단계로 진행하십시오.

IBM MQ Explorer를 사용하여 큐 관리자 작성

다음 단계에 따라 IBM MQ Explorer를 사용하여 MSCS에서 사용할 큐 관리자를 작성하십시오.

1. 시작 메뉴에서 IBM MQ Explorer를 시작하십시오.
2. 네비게이터 보기에서 트리 노드를 펼쳐서 큐 관리자 트리 노드를 찾으십시오.
3. 큐 관리자 트리 노드를 마우스 오른쪽 단추로 클릭하고 **새로 작성** > **큐 관리자**를 선택하십시오. 큐 관리자 작성 패널이 표시됩니다.
4. 대화 상자(단계 1)를 완료한 후 **다음**을 클릭하십시오.
5. 대화 상자(단계 2)를 완료한 후 **다음**을 클릭하십시오.
6. 대화 상자(단계 3)를 완료하여 큐 관리자 시작 및 서버 연결 채널 작성이 선택되지 않도록 한 후 **다음**을 클릭하십시오.
7. 대화 상자(단계 4)를 완료한 후 **완료**를 클릭하십시오.
8. 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』 단계로 진행하십시오.

Windows *MSCS* 스토리지로 큐 관리자 이동

이 프로시저는 기존 큐 관리자를 MSCS 제어 아래에 넣기에 적당하게 되도록 구성합니다.

이렇게 하려면 로그 파일 및 데이터 파일을 공유 디스크로 이동하여 실패가 발생할 경우 다른 컴퓨터에서 사용 가능하도록 해야 합니다. 예를 들어, 기존 큐 관리자에는 C:\WebSphere MQ\log*QMname* 및 C:\WebSphere MQ\qmgrs*QMname*과(와) 같은 경로가 있을 수 있습니다.

주의: 직접 파일을 이동하려고 시도하지 마십시오. 이 주제에서 설명된 대로 IBM MQ MSCS 지원의 일부로 제공된 유틸리티 프로그램을 사용하십시오.

이동할 큐 관리자가 TLS 연결을 사용하고 TLS 키 저장소가 로컬 시스템의 큐 관리자 데이터 디렉토리에 있는 경우 이 키 저장소는 나머지 큐 관리자와 함께 공유 디스크로 이동됩니다. 기본적으로, TLS키 저장소 위치인 SSLKEYR을 지정하는 큐 관리자 속성은 큐 관리자 데이터 디렉토리 아래에 있는 *MQ_INSTALLATION_PATH*\qmgrs\QMGRNAME\ssl\key로 설정됩니다. *MQ_INSTALLATION_PATH*는 IBM MQ가 설치된 상위 레벨 디렉토리를 나타냅니다. hamvmqm 명령은 이 큐 관리자 속성을 수정하지 않습니다. 이 상황에서는 새 TLS키 저장소 파일을 가리키도록 IBM MQ Explorer 또는 MQSC 명령 ALTER QMGR를 사용하여 큐 관리자 속성 SSLKEYR을 수정해야 합니다.

이 프로시저는 다음과 같습니다.

1. 큐 관리자를 종료하고 오류가 없는지 검사하십시오.
2. 큐 관리자의 로그 파일 또는 큐 파일이 이미 공유 디스크에 저장된 경우 이 프로시저의 나머지 부분은 건너뛰고 386 페이지의 『MSCS 제어 아래 큐 관리자 넣기』로 바로 이동하십시오.
3. 큐 파일 및 로그 파일의 전체 매체 백업을 작성하고 안전한 위치에 백업을 저장하십시오(이 조치가 중요한 이유는 394 페이지의 『큐 관리자 로그 파일』 참조).
4. 이미 적합한 공유 디스크 자원이 있는 경우 6단계로 진행하십시오. 그렇지 않으면, MSCS 클러스터 관리자를 사용하여 큐 관리자 로그 파일 및 데이터(큐) 파일을 저장할 충분한 용량을 갖는 공유 디스크 유형의 자원을 작성하십시오.
5. 공유 디스크를 한 클러스터 노드에서 다른 클러스터 노드로 이동하고 다시 돌아오게 하려면 MSCS 클러스터 관리자를 사용하여 공유 디스크를 테스트하십시오.

6. 공유 디스크가 큐 관리자 로그 및 데이터 파일이 로컬로 저장되는 클러스터 노드에서 온라인인지 확인하십시오.
7. 다음과 같이 유틸리티 프로그램을 실행하여 큐 관리자를 이동하십시오.

   ```
   hamvmqm /m qmname /dd " e: \
   IBM MQ " /ld " e: \
   IBM MQ \log"
   ```

   *qmname*은 큐 관리자 이름으로, *e*는 공유 디스크 드라이브 문자로, *IBM MQ*는 선택한 디렉토리로 대체합니다. 디렉토리는 아직 없는 경우에 작성됩니다.
8. 큐 관리자가 작동하는지 확인하려면 IBM MQ Explorer를 사용하여 큐 관리자를 테스트하십시오. 예를 들면, 다음과 같습니다.
   a. 큐 관리자 트리 노드를 마우스의 오른쪽 단추로 클릭한 후 **시작**을 선택하십시오. 큐 관리자가 시작됩니다.
   b. 큐 트리 노드를 마우스 오른쪽 단추로 누른 후 **새로 작성** > **로컬 큐 ...**를 선택하십시오. 그리고 큐에 이름을 제공한다.
   c. **완료**를 클릭하십시오.
   d. 마우스의 오른쪽 단추로 큐를 클릭한 후 **테스트 메시지 넣기...**를 선택하십시오. 테스트 메시지 넣기 패널이 표시됩니다.
   e. 일부 메시지 텍스트를 입력한 다음 **테스트 메시지 넣기**를 클릭하고 패널을 닫으십시오.
   f. 큐에서 마우스의 오른쪽 단추를 클릭한 다음 **메시지 찾아보기...**를 선택하십시오. 메시지 브라우저 패널이 표시됩니다.
   g. 메시지가 큐에 있는지 확인한 다음 **닫기**를 클릭하십시오. 메시지 브라우저 패널이 닫힙니다.
   h. 큐를 마우스의 오른쪽 단추로 클릭한 후 **메시지 지우기...**를 선택하십시오. 큐에 있는 메시지가 지워집니다.
   i. 큐를 마우스의 오른쪽 단추로 클릭한 후 **삭제...**를 선택하십시오. 확인 패널이 표시되면 **확인**을 클릭하십시오. 큐가 삭제됩니다.
   j. 큐 관리자 트리 노드를 마우스의 오른쪽 단추로 클릭한 후 **중지...**를 선택하십시오. 큐 관리자 종료 패널이 표시됩니다.
   k. **확인**을 클릭하십시오. 큐 관리자가 중지됩니다.
9. IBM MQ 관리자로서 큐 관리자의 시동 속성이 수동으로 설정되어 있는지 확인하십시오. IBM MQ Explorer에서 큐 관리자 등록 정보 패널에서 시작 필드를 `manual` 로 설정하십시오.
10. 386 페이지의 『MSCS 제어 아래 큐 관리자 넣기』 단계로 진행하십시오.

Windows *MSCS* 제어 아래 큐 관리자 넣기

필수조건 태스크를 포함하여 큐 관리자를 MSCS의 제어 아래 배치하는 데 포함되는 태스크입니다.

MSCS 제어 아래 큐 관리자를 넣기 전에

MSCS 제어 아래 큐 관리자를 넣기 전에 다음 태스크를 수행하십시오.

1. IBM MQ 및 해당 MSCS 지원이 클러스터 내 두 시스템에 모두 설치되어 있는지 및 383 페이지의 『MSCS 클러스터링을 위한 IBM MQ 설정』에 설명된 대로 각 컴퓨터의 소프트웨어가 동일한지 확인하십시오.
2. **haregtyp** 유틸리티 프로그램을 사용하여 모든 클러스터 노드에서 IBM MQ를 MSCS 자원 유형으로 등록하십시오. 추가 정보는 395 페이지의 『MSCS 유틸리티 프로그램에 대한 지원』의 내용을 참조하십시오.
3. 아직 큐 관리자를 작성하지 않은 경우에는 384 페이지의 『MSCS에서 사용할 큐 관리자 작성』의 내용을 참조하십시오.
4. 큐 관리자를 작성했거나 큐 관리자가 이미 존재하는 경우 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』의 프로시저를 수행했는지 확인하십시오.
5. 명령 프롬프트 또는 IBM MQ 탐색기를 사용하여 실행 중인 큐 관리자를 중지하십시오.
6. 이 주제의 다음 Windows 프로시저 중 하나로 이동하기 전에 공유 디스크의 MSCS 조작을 테스트하십시오.

Windows Server 2012

Windows Server 2012에서 큐 관리자를 MSCS 제어 아래 배치하려면 다음 프로시저를 사용하십시오.

1. 큐 관리자를 호스팅하는 클러스터 노드 컴퓨터에 로그인하거나 클러스터 관리 권한이 있는 사용자로 원격 워크스테이션에 로그인하고 큐 관리자를 호스팅하는 클러스터 노드에 연결하십시오.
2. 장애 복구 클러스터 관리 도구를 시작하십시오.
3. **장애 조치 클러스터 관리 > 클러스터 연결 ...** 을 마우스 오른쪽 단추로 클릭하십시오. 클러스터에 대한 연결을 열 수 있습니다.
4. Windows의 이전 버전에서 MSCS 클러스터 관리자에 사용된 그룹 설계와는 반대로 장애 복구 클러스터 관리 도구는 서비스 및 애플리케이션의 개념을 사용합니다. 구성된 서비스 또는 애플리케이션은 클러스터될 한 애플리케이션에 필요한 모든 자원을 포함합니다. 다음과 같이 MSCS 아래에서 큐 관리자를 구성할 수 있습니다.
   a. 클러스터에서 마우스의 오른쪽 단추를 클릭하고 **역할 구성**을 선택하여 구성 마법사를 시작하십시오.
   b. "서비스 또는 애플리케이션 선택" 패널에서 **기타 서버**를 선택하십시오.
   c. 적절한 IP 주소를 클라이언트 액세스 지점으로 선택하십시오.

      이 주소는 클라이언트 및 다른 큐 관리자가 가상 큐 관리자에 연결하는 데 사용될 미사용 IP 주소여야 합니다. 이 IP 주소는 두 노드 중 하나의 정상(정적) 주소가 아니며 두 노드 사이에서 플로트하는 추가 주소입니다. MSCS가 이 주소의 라우팅을 핸들링하지만 주소에 도달할 수 있는지는 확인하지 **않습니다**.
   d. 큐 관리자가 독점적으로 사용할 수 있는 스토리지 디바이스를 지정하십시오. 이 디바이스를 지정하려면 먼저 자원 인스턴스로 이 디바이스를 작성해야 합니다.

      한 개의 드라이브를 사용하여 로그 및 큐 파일 둘 다 저장하거나 이러한 파일을 여러 드라이브 전체에 분할할 수 있습니다. 둘 중 어느 경우든 각 큐 관리자에 자체 공유 디스크가 있는 경우 이 큐 관리자가 사용하는 모든 드라이브가 이 큐 관리자에 대해 독점적이어야 합니다. 즉, 그 외 어떤 것도 해당드라이브에 종속되지 않아야 합니다. 또한 큐 관리자가 사용하는 모든 드라이브에 대해 자원 인스턴스를 작성해야 합니다.

      드라이브에 대한 자원 유형은 사용하고 있는 SCSI 지원에 따라 다릅니다. SCSI 어댑터 지시사항을 참조하십시오. 이미 각 공유 드라이브에 대한 그룹 및 자원이 있을 수 있습니다. 이 경우 각 드라이브에 대해 자원 인스턴스를 작성하지 않아도 됩니다. 이를 현재 그룹에서 큐 관리자에 대해 작성된 그룹으로 이동하십시오.

      각 드라이브 자원에 대해 가능한 소유자를 두 노드로 설정하십시오. 종속 자원을 없음으로 설정하십시오.
   e. "자원 유형 선택" 패널에서 **MQSeries MSCS** 자원을 선택하십시오.
   f. 마법사에서 나머지 단계를 완료하십시오.
5. 자원을 온라인으로 설정하기 전에 다음과 같은 MQSeries® MSCS 자원을 추가 구성해야 합니다.
   a. '새 MQSeries MSCS'라고 하는 자원을 포함하는 새로 정의된 서비스를 선택하십시오.
   b. MQ 자원의 **특성**에서 마우스의 오른쪽 단추를 클릭하십시오.
   c. 다음 자원을 구성하십시오.
      - `Name` ; 어떤 큐 관리자를 나타내는 이름인지 식별하기 쉬운 이름을 선택하십시오.
      - `Run in a separate Resource Monitor` ; 더 나은 격리를 위해
      - `Possible owners` ; 두 노드로 설정합니다.
      - `Dependencies` ; 이 큐 관리자의 드라이브 및 IP 주소를 추가하십시오.

        경고: 이러한 종속항목 추가에 실패하는 것은 IBM MQ가 장애 복구 중 큐 관리자 상태를 잘못된 클러스터 디스크에 기록하려고 시도하는 것을 의미합니다. 다수의 프로세스가 동시에 이 디스크에 기록하려고 시도할 수 있기 때문에 일부 IBM MQ 프로세스는 실행이 차단될 수 있습니다.
      - `Parameters` ; 다음과 같습니다.
        - `QueueManagerName`(필수); 이 자원이 제어할 큐 관리자의 이름입니다. 이 큐 관리자는 로컬 컴퓨터에 있어야 합니다.
        - `PostOnlineCommand`(선택적); 큐 관리자 자원이 그 상태를 오프라인에서 온라인으로 변경할 때마다 실행할 프로그램을 지정할 수 있습니다. 자세한 정보는 395 페이지의 『MSCS의 PostOnlineCommand 및 PreOfflineCommand』의 내용을 참조하십시오.

- PreOfflineCommand(선택적); 큐 관리자 자원이 그 상태를 온라인에서 오프라인으로 변경할 때마다 실행할 프로그램을 지정할 수 있습니다. 자세한 정보는 395 페이지의 『MSCS의 PostOnlineCommand 및 PreOfflineCommand』의 내용을 참조하십시오.

  참고: *looksAlive* 폴 간격은 기본값 5,000밀리초로 설정됩니다. *isAlive* 폴 간격은 기본값 60,000밀리초로 설정됩니다. 이러한 기본값은 자원 정의를 완료한 후에만 수정할 수 있습니다. 추가 세부사항은 392 페이지의 『MSCS에서 looksAlive 및 isAlive 폴링』의 내용을 참조하십시오.

d. 선택적으로 선호 노드를 설정하십시오(그러나 395 페이지의 『MSCS의 선호 노드 사용』의 주석 참고).

e. 장애 복구 정책은 기본적으로 실제 값으로 설정되지만, 자원 장애 복구 및 그룹 장애 복구를 제어하는 임계값 및 기간을 조정하여 큐 관리자에 놓여진 로드에 일치시킬 수 있습니다.

6. MSCS 클러스터 관리자에서 큐 관리자를 온라인으로 설정하고 테스트 워크로드를 부과하여 큐 관리자를 테스트하십시오. 테스트 큐 관리자를 실험하고 있다면 IBM MQ 탐색기를 사용하십시오. 예를 들면, 다음과 같습니다.

   a. 큐 트리 노드를 마우스 오른쪽 단추로 누른 후 **새로 작성** > **로컬 큐 ...**를 선택하십시오. 그리고 큐에 이름을 제공한다.

   b. **완료**를 클릭하십시오. 큐가 작성되고 컨텐츠 보기에 표시됩니다.

   c. 마우스의 오른쪽 단추로 큐를 클릭한 후 **테스트 메시지 넣기...**를 선택하십시오. 테스트 메시지 넣기 패널이 표시됩니다.

   d. 일부 메시지 텍스트를 입력한 다음 **테스트 메시지 넣기**를 클릭하고 패널을 닫으십시오.

   e. 큐에서 마우스의 오른쪽 단추를 클릭한 다음 **메시지 찾아보기...**를 선택하십시오. 메시지 브라우저 패널이 표시됩니다.

   f. 메시지가 큐에 있는지 확인한 다음 **닫기**를 클릭하십시오. 메시지 브라우저 패널이 닫힙니다.

   g. 큐를 마우스의 오른쪽 단추로 클릭한 후 **메시지 지우기...**를 선택하십시오. 큐에 있는 메시지가 지워집니다.

   h. 큐를 마우스의 오른쪽 단추로 클릭한 후 **삭제...**를 선택하십시오. 확인 패널이 표시되면 **확인**을 클릭하십시오. 큐가 삭제됩니다.

7. MSCS 클러스터 관리자를 사용하여 큐 관리자를 오프라인으로 가져오고 다시 온라인으로 되돌릴 수 있는지 테스트하십시오.

8. 장애 복구를 시뮬레이션하십시오.

   MSCS 클러스터 관리자에서 큐 관리자를 포함하는 그룹에서 마우스의 오른쪽 단추를 클릭하고 `Move Group`을 선택하십시오. 이를 수행하는 데에는 몇 분 정도 소요될 수 있습니다. (다른 때 큐 관리자를 다른 노드로 신속하게 이동하려고 하는 경우 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』의 프로시저에 따르십시오.) 또한 마우스의 오른쪽 단추를 클릭하고 `Initiate Failure`를 선택할 수 있습니다. 이 조치(로컬 재시작 또는 장애 복구)는 현재 상태 및 구성 설정에 따라 다릅니다.

Windows Server 2008

Windows Server 2008에서 큐 관리자를 MSCS 제어 하에 배치하려면 다음 프로시저를 사용하십시오.

1. 큐 관리자를 호스팅하는 클러스터 노드 컴퓨터에 로그인하거나 클러스터 관리 권한이 있는 사용자로 원격 워크스테이션에 로그인하고 큐 관리자를 호스팅하는 클러스터 노드에 연결하십시오.

2. 장애 복구 클러스터 관리 도구를 시작하십시오.

3. **장애 조치 클러스터 관리 > 클러스터 관리 ...** 를 마우스 오른쪽 단추로 클릭하십시오. 클러스터에 대한 연결을 열 수 있습니다.

4. Windows의 이전 버전에서 MSCS 클러스터 관리자에 사용된 그룹 설계와는 반대로 장애 복구 클러스터 관리 도구는 서비스 및 애플리케이션의 개념을 사용합니다. 구성된 서비스 또는 애플리케이션은 클러스터될 한 애플리케이션에 필요한 모든 자원을 포함합니다. 다음과 같이 MSCS 아래에서 큐 관리자를 구성할 수 있습니다.

   a. **서비스 및 애플리케이션 > 서비스 또는 애플리케이션 구성 ...** 을 마우스 오른쪽 단추로 클릭하십시오. 구성 마법사를 시작한다.

   b. **서비스 또는 애플리케이션 선택** 패널에서 **기타 서버**를 선택하십시오.

   c. 적절한 IP 주소를 클라이언트 액세스 지점으로 선택하십시오.

이 주소는 클라이언트 및 다른 큐 관리자가 가상 큐 관리자에 연결하는 데 사용될 미사용 IP 주소여야 합니다. 이 IP 주소는 두 노드 중 하나의 정상(정적) 주소가 아니며 두 노드 사이에서 플로트하는 추가 주소입니다. MSCS가 이 주소의 라우팅을 핸들링하지만 주소에 도달할 수 있는지는 확인하지 **않습니다**.

d. 큐 관리자가 독점적으로 사용할 수 있는 스토리지 디바이스를 지정하십시오. 이 디바이스를 지정하려면 먼저 자원 인스턴스로 이 디바이스를 작성해야 합니다.

   한 개의 드라이브를 사용하여 로그 및 큐 파일 둘 다 저장하거나 이러한 파일을 여러 드라이브 전체에 분할할 수 있습니다. 둘 중 어느 경우든 각 큐 관리자에 자체 공유 디스크가 있는 경우 이 큐 관리자가 사용하는 모든 드라이브가 이 큐 관리자에 대해 독점적이어야 합니다. 즉, 그 외 어떤 것도 해당드라이브에 종속되지 않아야 합니다. 또한 큐 관리자가 사용하는 모든 드라이브에 대해 자원 인스턴스를 작성해야 합니다.

   드라이브에 대한 자원 유형은 사용하고 있는 SCSI 지원에 따라 다릅니다. SCSI 어댑터 지시사항을 참조하십시오. 이미 각 공유 드라이브에 대한 그룹 및 자원이 있을 수 있습니다. 이 경우 각 드라이브에 대해 자원 인스턴스를 작성하지 않아도 됩니다. 이를 현재 그룹에서 큐 관리자에 대해 작성된 그룹으로 이동하십시오.

   각 드라이브 자원에 대해 가능한 소유자를 두 노드로 설정하십시오. 종속 자원을 없음으로 설정하십시오.

e. **자원 유형 선택** 패널에서 **MQSeries MSCS** 자원을 선택하십시오.

f. 마법사에서 나머지 단계를 완료하십시오.

5. 자원을 온라인으로 설정하기 전에 다음과 같은 MQSeries MSCS 자원을 추가 구성해야 합니다.

   a. '새 MQSeries MSCS'라고 하는 자원을 포함하는 새로 정의된 서비스를 선택하십시오.

   b. MQ 자원의 **특성**에서 마우스의 오른쪽 단추를 클릭하십시오.

   c. 다음 자원을 구성하십시오.

      - `Name` ; 어떤 큐 관리자를 나타내는 이름인지 식별하기 쉬운 이름을 선택하십시오.
      - `Run in a separate Resource Monitor` ; 더 나은 격리를 위해
      - `Possible owners` ; 두 노드로 설정합니다.
      - `Dependencies` ; 이 큐 관리자의 드라이브 및 IP 주소를 추가하십시오.

        경고: 이러한 종속항목 추가에 실패하는 것은 IBM MQ가 장애 복구 중 큐 관리자 상태를 잘못된 클러스터 디스크에 기록하려고 시도하는 것을 의미합니다. 다수의 프로세스가 동시에 이 디스크에 기록하려고 시도할 수 있기 때문에 일부 IBM MQ 프로세스는 실행이 차단될 수 있습니다.

      - `Parameters` ; 다음과 같습니다.

        – `QueueManagerName`(필수); 이 자원이 제어할 큐 관리자의 이름입니다. 이 큐 관리자는 로컬 컴퓨터에 있어야 합니다.

        – `PostOnlineCommand`(선택적); 큐 관리자 자원이 그 상태를 오프라인에서 온라인으로 변경할 때마다 실행할 프로그램을 지정할 수 있습니다. 자세한 정보는 395 페이지의 『MSCS의 PostOnlineCommand 및 PreOfflineCommand』의 내용을 참조하십시오.

        – `PreOfflineCommand`(선택적); 큐 관리자 자원이 그 상태를 온라인에서 오프라인으로 변경할 때마다 실행할 프로그램을 지정할 수 있습니다. 자세한 정보는 395 페이지의 『MSCS의 PostOnlineCommand 및 PreOfflineCommand』의 내용을 참조하십시오.

        참고: *looksAlive* 폴 간격은 기본값 5,000밀리초로 설정됩니다. *isAlive* 폴 간격은 기본값 60,000밀리초로 설정됩니다. 이러한 기본값은 자원 정의를 완료한 후에만 수정할 수 있습니다. 추가 세부사항은 392 페이지의 『MSCS에서 looksAlive 및 isAlive 폴링』의 내용을 참조하십시오.

   d. 선택적으로 선호 노드를 설정하십시오(그러나 395 페이지의 『MSCS의 선호 노드 사용』의 주석 참고).

   e. 장애 복구 정책은 기본적으로 실제 값으로 설정되지만, 자원 장애 복구 및 그룹 장애 복구를 제어하는 임계값 및 기간을 조정하여 큐 관리자에 놓여진 로드에 일치시킬 수 있습니다.

6. MSCS 클러스터 관리자에서 큐 관리자를 온라인으로 설정하고 테스트 워크로드를 부과하여 큐 관리자를 테스트하십시오. 테스트 큐 관리자를 실험하고 있다면 IBM MQ 탐색기를 사용하십시오. 예를 들면, 다음과 같습니다.

   a. 큐 트리 노드를 마우스 오른쪽 단추로 누른 후 **새로 작성** > **로컬 큐 ...**를 선택하십시오. 그리고 큐에 이름을 제공한다.

b. **완료**를 클릭하십시오. 큐가 작성되고 컨텐츠 보기에 표시됩니다.
c. 마우스의 오른쪽 단추로 큐를 클릭한 후 **테스트 메시지 넣기...**를 선택하십시오. **테스트 메시지 넣기** 패널이 표시됩니다.
d. 일부 메시지 텍스트를 입력한 다음 **테스트 메시지 넣기**를 클릭하고 패널을 닫으십시오.
e. 큐에서 마우스의 오른쪽 단추를 클릭한 다음 **메시지 찾아보기...**를 선택하십시오. **메시지 브라우저** 패널이 표시됩니다.
f. 메시지가 큐에 있는지 확인한 다음 **닫기**를 클릭하십시오. **메시지 브라우저** 패널이 닫힙니다.
g. 큐를 마우스의 오른쪽 단추로 클릭한 후 **메시지 지우기...**를 선택하십시오. 큐에 있는 메시지가 지워집니다.
h. 큐를 마우스의 오른쪽 단추로 클릭한 후 **삭제...**를 선택하십시오. 확인 패널이 표시되면 **확인**을 클릭하십시오. 큐가 삭제됩니다.

7. MSCS 클러스터 관리자를 사용하여 큐 관리자를 오프라인으로 가져오고 다시 온라인으로 되돌릴 수 있는지 테스트하십시오.
8. 장애 복구를 시뮬레이션하십시오.

   MSCS 클러스터 관리자에서 큐 관리자를 포함하는 그룹에서 마우스의 오른쪽 단추를 클릭하고 `Move Group`을 선택하십시오. 이를 수행하는 데에는 몇 분 정도 소요될 수 있습니다. (다른 때 큐 관리자를 다른 노드로 신속하게 이동하려고 하는 경우 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』의 프로시저에 따르십시오.) 또한 마우스의 오른쪽 단추를 클릭하고 `Initiate Failure`를 선택할 수 있습니다. 이 조치(로컬 재시작 또는 장애 복구)는 현재 상태 및 구성 설정에 따라 다릅니다.

Windows 2003

Windows 2003에서 큐 관리자를 MSCS 제어 하에 배치하려면 다음 프로시저를 사용하십시오.

1. 큐 관리자를 호스팅하는 클러스터 노드 컴퓨터에 로그인하거나 클러스터 관리 권한이 있는 사용자로 원격 워크스테이션에 로그인하고 큐 관리자를 호스팅하는 클러스터 노드에 연결하십시오.
2. MSCS 클러스터 관리자를 시작하십시오.
3. 클러스터에 대한 연결을 여십시오.
4. 큐 관리자 자원을 포함시키기 위해 사용될 MSCS 그룹을 작성하십시오. 이름과 관련된 큐 관리자를 명백히 알 수 있는 방식으로 그룹의 이름을 지정하십시오. 각 그룹은 383 페이지의 『MSCS에서 다중 큐 관리자 사용』에 설명된 대로 다중 큐 관리자를 포함할 수 있습니다.

   남아 있는 모든 단계에서 그룹을 사용하십시오.
5. 큐 관리자가 사용하는 각각의 SCSI 논리 드라이브에 대해 자원 인스턴스를 작성하십시오.

   한 개의 드라이브를 사용하여 로그 및 큐 파일 둘 다 저장하거나 이러한 파일을 여러 드라이브 전체에 분할할 수 있습니다. 둘 중 어느 경우든 각 큐 관리자에 자체 공유 디스크가 있는 경우 이 큐 관리자가 사용하는 모든 드라이브가 이 큐 관리자에 대해 독점적이어야 합니다. 즉, 그 외 어떤 것도 해당드라이브에 종속되지 않아야 합니다. 또한 큐 관리자가 사용하는 모든 드라이브에 대해 자원 인스턴스를 작성해야 합니다.

   드라이브에 대한 자원 유형은 사용하고 있는 SCSI 지원에 따라 다릅니다. SCSI 어댑터 지시사항을 참조하십시오. 이미 각 공유 드라이브에 대한 그룹 및 자원이 있을 수 있습니다. 이 경우 각 드라이브에 대해 자원 인스턴스를 작성하지 않아도 됩니다. 이를 현재 그룹에서 큐 관리자에 대해 작성된 그룹으로 이동하십시오.

   각 드라이브 자원에 대해 가능한 소유자를 두 노드로 설정하십시오. 종속 자원을 없음으로 설정하십시오.
6. IP 주소에 대한 자원 인스턴스를 작성하십시오.

   IP 주소 자원(자원 유형 *IP* 주소)을 작성하십시오. 이 주소는 클라이언트 및 다른 큐 관리자가 가상 큐 관리자에 연결하는 데 사용될 미사용 IP 주소여야 합니다. 이 IP 주소는 두 노드 중 하나의 정상(정적) 주소가 아니며 두 노드 사이에서 플로트하는 추가 주소입니다. MSCS가 이 주소의 라우팅을 핸들링하지만 주소에 도달할 수 있는지는 확인하지 **않습니다**.
7. 큐 관리자에 대한 자원 인스턴스를 작성하십시오.

   자원 유형 *IBM MQ MSCS*를 작성하십시오. 마법사는 다음을 포함하여 다양한 항목을 프롬프트합니다.

   - `Name` ; 어떤 큐 관리자를 나타내는 이름인지 식별하기 쉬운 이름을 선택하십시오.

- `Add to group`; 작성한 그룹을 사용합니다.
- `Run in a separate Resource Monitor`; 더 나은 격리를 위해
- `Possible owners`; 두 노드로 설정합니다.
- `Dependencies`; 이 큐 관리자의 드라이브 및 IP 주소를 추가하십시오.

  경고: 이러한 종속항목 추가에 실패하는 것은 IBM MQ가 장애 복구 중 큐 관리자 상태를 잘못된 클러스터 디스크에 기록하려고 시도하는 것을 의미합니다. 다수의 프로세스가 동시에 이 디스크에 기록하려고 시도할 수 있기 때문에 일부 IBM MQ 프로세스는 실행이 차단될 수 있습니다.
- `Parameters`; 다음과 같습니다.
  - `QueueManagerName`(필수); 이 자원이 제어할 큐 관리자의 이름입니다. 이 큐 관리자는 로컬 컴퓨터에 있어야 합니다.
  - `PostOnlineCommand`(선택적); 큐 관리자 자원이 그 상태를 오프라인에서 온라인으로 변경할 때마다 실행할 프로그램을 지정할 수 있습니다. 자세한 정보는 395 페이지의 『MSCS의 PostOnlineCommand 및 PreOfflineCommand』의 내용을 참조하십시오.
  - `PreOfflineCommand`(선택적); 큐 관리자 자원이 그 상태를 온라인에서 오프라인으로 변경할 때마다 실행할 프로그램을 지정할 수 있습니다. 자세한 정보는 395 페이지의 『MSCS의 PostOnlineCommand 및 PreOfflineCommand』의 내용을 참조하십시오.

    참고: *looksAlive* 폴 간격은 기본값 5,000밀리초로 설정됩니다. *isAlive* 폴 견각은 기본값 30,000밀리초로 설정됩니다. 이러한 기본값은 자원 정의를 완료한 후에만 수정할 수 있습니다. 추가 세부사항은 392 페이지의 『MSCS에서 looksAlive 및 isAlive 폴링』의 내용을 참조하십시오.

8. 선택적으로 선호 노드를 설정하십시오(그러나 395 페이지의 『MSCS의 선호 노드 사용』의 주석 참고).
9. 자원 복구 정책(그룹에 대한 특성에 정의됨)은 기본적으로 실제 값으로 설정되지만, 자원 장애 복구 및 그룹 장애 복구를 제어하는 임계값 및 기간을 조정하여 큐 관리자에 놓여진 로드에 일치시킬 수 있습니다.
10. MSCS 클러스터 관리자에서 큐 관리자를 온라인으로 설정하고 테스트 워크로드를 부과하여 큐 관리자를 테스트하십시오. 테스트 큐 관리자를 실험하고 있다면 IBM MQ 탐색기를 사용하십시오. 예를 들면, 다음과 같습니다.
    a. 큐 트리 노드를 마우스 오른쪽 단추로 누른 후 **새로 작성 > 로컬 큐 ...**를 선택하십시오. 그리고 큐에 이름을 제공한다.
    b. **완료**를 클릭하십시오. 큐가 작성되고 컨텐츠 보기에 표시됩니다.
    c. 마우스의 오른쪽 단추로 큐를 클릭한 후 **테스트 메시지 넣기...**를 선택하십시오. **테스트 메시지 넣기** 패널이 표시됩니다.
    d. 일부 메시지 텍스트를 입력한 다음 **테스트 메시지 넣기**를 클릭하고 패널을 닫으십시오.
    e. 큐에서 마우스의 오른쪽 단추를 클릭한 다음 **메시지 찾아보기...**를 선택하십시오. **메시지 브라우저** 패널이 표시됩니다.
    f. 메시지가 큐에 있는지 확인한 다음 **닫기**를 클릭하십시오. **메시지 브라우저** 패널이 닫힙니다.
    g. 큐를 마우스의 오른쪽 단추로 클릭한 후 **메시지 지우기...**를 선택하십시오. 큐에 있는 메시지가 지워집니다.
    h. 큐를 마우스의 오른쪽 단추로 클릭한 후 **삭제...**를 선택하십시오. 확인 패널이 표시되면 **확인**을 클릭하십시오. 큐가 삭제됩니다.
11. MSCS 클러스터 관리자를 사용하여 큐 관리자를 오프라인으로 가져오고 다시 온라인으로 되돌릴 수 있는지 테스트하십시오.
12. 장애 복구를 시뮬레이션하십시오.

    MSCS 클러스터 관리자에서 큐 관리자를 포함하는 그룹에서 마우스의 오른쪽 단추를 클릭하고 `Move Group`을 선택하십시오. 이를 수행하는 데에는 몇 분 정도 소요될 수 있습니다. (다른 때 큐 관리자를 다른 노드로 신속하게 이동하려고 하는 경우 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』의 프로시저에 따르십시오.) 또한 마우스의 오른쪽 단추를 클릭하고 `Initiate Failure`를 선택할 수 있습니다. 이 조치(로컬 재시작 또는 장애 복구)는 현재 상태 및 구성 설정에 따라 다릅니다.

Windows *MSCS*에서 *looksAlive* 및 *isAlive* 폴링

looksAlive 및 *isAlive*는 MSCS가 자원 유형 제공 라이브러리 코드로 콜백하고 자원이 그 작업 상태를 판별하기 위해 검사를 수행하도록 요청하는 간격입니다. 이는 결국 MSCS가 자원의 장애를 복구하려고 시도하는지 여부를 판별합니다.

looksAlive 간격이 경과될 때마다(기본값 5000ms) 큐 관리자 자원이 호출되어 그 상태가 만족인지 판별하기 위한 자체 검사를 수행합니다.

isAlive 간격이 경과될 때마다(기본값 30000ms) 그에 대한 큐 관리자 자원이 다시 호출되어 자원이 올바르게 작동하는지 판별하기 위한 다른 검사를 수행합니다. 이 변수를 사용하면 두 레벨의 자원 유형 검사가 가능합니다.

1. 자원이 작동하는 것으로 표시되는 지 여부를 설정하기 위한 *looksAlive* 상태 검사입니다.
2. 큐 관리자 자원이 활성인지 판별하는 보다 중요한 *isAlive* 검사입니다.

큐 관리자 자원이 활성이 아닌 것으로 판별되는 경우 MSCS는 다른 고급 MSCS 옵션에 따라 자원 및 연관된 종속 자원에 대한 장애 복구를 클러스터의 다른 노드로 트리거합니다. 자세한 정보는 MSCS 문서를 참조하십시오.

Windows *MSCS* 제어에서 큐 관리자 제거

MSCS 제어에서 큐 관리자를 제거하고 이를 수동 관리로 리턴할 수 있습니다.

유지보수 조작을 위해 MSCS 제어에서 큐 관리자를 제거할 필요는 없습니다. MSCS 클러스터 관리자를 사용하여 큐 관리자를 임시로 오프라인으로 가져와서 유지보수 조작을 수행할 수 있습니다. MSCS 제어에서 큐 관리자를 제거하는 것은 좀 더 영구적인 변경이며 MSCS가 더 이상 큐 관리자의 제어를 받지 않도록 결정한 경우에만 수행하십시오.

제거할 큐 관리자가 TSL 연결을 사용하는 경우 IBM MQ 탐색기 또는 MQSC 명령 `ALTER QMGR`을 사용하여 로컬 디렉토리의 TLS 키 저장소 파일을 가리키도록 큐 관리자 속성 SSLKEYR을 수정해야 합니다.

프로시저는 다음과 같습니다.

1. 392 페이지의 『MSCS에서 큐 관리자를 오프라인으로 가져오기』에 설명한 대로 MSCS 클러스터 관리자를 사용하여 큐 관리자 자원을 오프라인으로 설정합니다.
2. 자원 인스턴스를 파기하십시오. 이는 큐 관리자를 영구 삭제하지는 않습니다.
3. 선택적으로 큐 관리자 파일을 공유 드라이브에서 로컬 드라이브로 다시 마이그레이션하십시오. 이를 수행하려면 392 페이지의 『MSCS 스토리지에서 큐 관리자 리턴』의 내용을 참조하십시오.
4. 큐 관리자를 테스트하십시오.

MSCS에서 큐 관리자를 오프라인으로 가져오기

큐 관리자를 MSCS에서 오프라인으로 가져오려면 다음 단계를 수행하십시오.

1. MSCS 클러스터 관리자를 시작하십시오.
2. 클러스터에 대한 연결을 여십시오.
3. `Groups`를 선택하거나 Windows 2012를 사용하는 경우 `Role` 를 선택하고 이동할 큐 관리자를 포함하는 그룹을 여십시오.
4. 큐 관리자 자원을 선택하십시오.
5. 마우스의 오른쪽 단추를 클릭하고 `Offline`을 선택하십시오.
6. 완료될 때까지 대기하십시오.

MSCS 스토리지에서 큐 관리자 리턴

이 프로시저는 큐 관리자가 해당 컴퓨터의 로컬 드라이브로 돌아가도록 구성합니다. 즉, 이 큐 관리자는 정상 IBM MQ 큐 관리자가 됩니다. 이렇게 하려면 공유 디스크에서 로그 파일 및 데이터 파일을 이동합니다. 예를 들어, 기존 큐 관리자에는 `E:\WebSphere MQ\log\`*`QMname`* 및 `E:\WebSphere MQ\qmgrs\`*`QMname`*과(와) 같은 경로가 있을 수 있습니다. 직접 파일을 이동하려고 시도하지 마십시오. IBM MQ MSCS 지원의 일부로 제공된 **hamvmqm** 유틸리티 프로그램을 사용하십시오.

1. 큐 파일 및 로그 파일의 전체 매체 백업을 작성하고 안전한 위치에 백업을 저장하십시오(이 조치가 중요한 이유는 394 페이지의 『큐 관리자 로그 파일』 참조).

2. 사용할 로컬 드라이브를 결정하고 이 드라이가 큐 관리자 로그 파일 및 데이터(큐) 파일을 저장하기에 충분한 용량을 가지는지 확인하십시오.
3. 파일이 현재 상주하는 공유 디스크가 큐 관리자 로그 및 데이터 파일을 이동할 클러스터 노드에서 온라인인지 확인하십시오.
4. 다음과 같이 유틸리티 프로그램을 실행하여 큐 관리자를 이동하십시오.

   ```
   hamvmqm /m qmname /dd " c:\
   IBM MQ " /ld "c:\
   IBM MQ \log"
   ```

   *qmname*은 큐 관리자 이름으로, *c*는 로컬 디스크 드라이브 문자로, *IBM MQ*는 선택한 디렉토리(아직 없는 경우 디렉토리가 작성됨)로 대체합니다.
5. 큐 관리자가 작동하는지 확인하려면 큐 관리자를 테스트하십시오(385 페이지의 『MSCS 스토리지로 큐 관리자 이동』에 설명된 대로).

Windows *MSCS* 사용의 힌트 및 팁

이 절은 MSCS에 대한 IBM MQ 지원을 효과적으로 사용하는 데 유용한 몇 가지 일반적인 정보를 포함합니다.

이 절은 MSCS에 대한 IBM MQ 지원을 효과적으로 사용하는 데 유용한 몇 가지 일반적인 정보를 포함합니다.

한 시스템에서 다른 시스템으로 큐 관리자의 장애를 복구하는 데 걸리는 시간은? 이는 큐 관리자에 대한 워크로드의 양 및 트래픽 혼합(예: 동기점 내에서 지속 적 트래픽의 양 및 실패 전 커미트된 트래픽 양)에 따라 매우 달라집니다. IBM 테스트에서는 약 1분의 장애 복구 및 페일백 시간이 제공되었습니다. 이는 초경량으로 로드된 큐 관리자이며 실제 시간은 로드에 따라 상당히 달라집니다.

Windows *MSCS*가 작동하는지 확인

다음 단계에 따라 실행 중인 MSCS 클러스터가 있는지 확인합니다.

384 페이지의 『MSCS에서 사용할 큐 관리자 작성』부터 시작하는 태스크 설명에서는 자원을 작성하고 마이그레이션하며 영구 삭제할 수 있는 실행 중인 MSCS 클러스터가 있는 것으로 가정합니다. 그러한 클러스터가 있는지 확인하려면 다음을 수행하십시오.

1. MSCS 클러스터 관리자를 사용하여 그룹을 작성하십시오.
2. 해당 그룹 내에서 시스템 클럭을 지정하여 일반 애플리케이션 자원의 인스턴스를 작성하십시오(경로 이름 `C:\winnt\system32\clock.exe` 및 `C:\`의 작업 디렉토리).
3. 자원을 온라인으로 가져올 수 있는지, 이 자원이 포함된 그룹을 다른 노드로 이동할 수 있는지 및 이 자원을 오프라인으로 가져올 수 있는지 확인하십시오.

Windows 수동 시동 및 *MSCS*

MSCS에서 관리하는 큐 관리자에 대해 시동 속성을 수동으로 설정해야 합니다. 이 설정은 IBM MQ MSCS 지원이 큐 관리자를 즉시 시작하지 않고도 MQSeries 서비스를 재시작할 수 있도록 합니다.

IBM MQ MSCS 지원은 모니터링 및 제어를 수행할 수 있도록 서비스를 재시작할 수 있어야 하지만 그 자체는 큐 관리자가 실행되고 있는 제어 및 시스템에 남아 있어야 합니다. 자세한 정보는 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』의 내용을 참조하십시오.

Windows *MSCS* 및 큐 관리자

MSCS 사용 시 큐 관리자에 관한 고려사항

다른 노드에서 일치하는 큐 관리자 작성

IBM MQ로 작업하도록 클러스터링하려면 노드 A에 있는 각 노드 B에 동일한 큐 관리자가 필요합니다. 그러나 두 번째 항목을 명시적으로 작성할 필요는 없습니다. 한 노드에서 큐 관리자를 작성하거나 준비하고, 이를 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』에 설명된 대로 다른 노드로 이동하면 이 큐 관리자는 해당 노드에서 완전하게 복제됩니다.

기본 큐 관리자

MSCS 제어 아래 기본 큐 관리자를 사용하지 마십시오. 큐 관리자에는 큐 관리자를 기본값으로 설정하는 특성이 없으며 IBM MQ는 자체 개별 레코드를 보관합니다. 장애 복구 시 기본값이 되도록 설정된 큐 관리자를 다른 컴퓨터로 이동하는 경우 다른 컴퓨터에서는 기본값이 되지 않습니다. 모든 애플리케이션이 이름별로 특정 큐 관리자를 참조하게 하십시오.

큐 관리자 삭제

큐 관리자가 노드를 이동한 경우 그 세부사항은 두 컴퓨터의 레지스트리에 존재합니다. 이를 삭제하려는 경우 한 컴퓨터에서는 평소대로 삭제한 다음 395 페이지의 『MSCS 유틸리티 프로그램에 대한 지원』에서 설명된 대로 유틸리티를 실행하여 다른 컴퓨터에서 레지스트리를 정리하십시오.

기존 큐 관리자에 대한 지원

큐 관리자 로그 파일 및 큐 파일을 두 시스템 간 공유 SCSI에 있는 디스크에 넣을 수 있다면 MSCS 제어 아래 기존 큐 관리자를 넣을 수 있습니다(382 페이지의 그림 73 참조). MSCS 자원이 작성되는 동안 큐 관리자를 잠시 오프라인으로 가져올 수 있습니다.

새 큐 관리자를 작성하려는 경우 MSCS와는 별도로 작성하고, 테스트한 다음 MSCS 제어 아래 넣으십시오. 다음을 참조하십시오.

- 384 페이지의 『MSCS에서 사용할 큐 관리자 작성』
- 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』
- 386 페이지의 『MSCS 제어 아래 큐 관리자 넣기』

관리할 큐 관리자를 MSCS에 알림

MSCS 클러스터 관리자를 통해 MSCS 제어 아래 배치할 큐 관리자를 선택하여 그러한 큐 관리자 각각에 대한 자원 인스턴스를 작성합니다. 이 프로세스는 해당 인스턴스가 관리하게 할 큐 관리자를 선택할 수 있는 자원 목록을 제시합니다.

큐 관리자 로그 파일

MSCS 스토리지로 큐 관리자 이동할 때 해당 로그 및 데이터 파일을 공유 디스크로 이동합니다(예는 385 페이지의 『MSCS 스토리지로 큐 관리자 이동』 참조).

이동하기 전에 큐 관리자를 확실히 닫고 데이터 파일 및 로그 파일의 전체 백업을 수행하는 것이 좋습니다.

다중 큐 관리자

IBM MQ MSCS 지원을 사용하여 각 시스템에서 큐 관리자를 실행하고 개별 큐 관리자를 MSCS 제어 아래 배치할 수 있습니다.

Windows 클러스터 관리에 항상 *MSCS* 사용
제어 명령 또는 IBM MQ Explorer를 사용하여 MSCS의 제어를 받는 큐 관리자에서 직접 조작 시작 및 중지를 수행하려고 하지 마십시오. 대신, MSCS 클러스터 관리자를 사용하거나 큐 관리자를 온라인으로 가져오거나 오프라인으로 하십시오.

MSCS 클러스터 관리자를 사용하면 실제로 MSCS의 제어 밖에서 큐 관리자를 시작한 경우 큐 관리자가 오프라인인 MSCS 보고로 인해 발생할 수 있는 혼동을 부분적으로 방지합니다. 또한 MSCS를 사용하지 않고 큐 관리자를 중지하면 MSCS에 의해 실패로 감지되어 다른 노드로의 장애 복구가 시작됩니다.

Windows *MSCS*에서 활성/활성 모드로 작업
MSCS 클러스터에 있는 두 컴퓨터 모두 활성/활성 모드로 큐 관리자를 실행할 수 있습니다. 대기 역할을 하는 완전히 유휴 상태인 시스템이 반드시 있어야 하는 것은 아닙니다(그러나 가능하며 필요한 경우에는 활성/수동 모드).

워크로드를 실행하기 위해 두 시스템 모두 사용하려는 경우 각 시스템에 전체 클러스터 워크로드를 만족스러운 수준의 성능으로 실행하기에 충분한 용량(프로세서, 메모리, 보조 스토리지)을 제공하십시오.

참고: MSCS를 Microsoft Transaction Server(COM+)와 함께 사용하는 경우 활성/활성 모드를 사용할 수 **없습니다**. 이는 IBM MQ를 MSCS 및 COM+와 함께 사용하기 위해서입니다.

- IBM MQ COM+ 지원을 사용하는 애플리케이션 컴포넌트는 COM+의 일부인 DTC(Distributed Transaction Coordinator)와 같은 컴퓨터에서 실행되어야 합니다.
- 큐 관리자도 같은 컴퓨터에서 실행되어야 합니다.
- DTC가 MSCS 자원으로 구성되어야 하고 따라서 언제든지 클러스터 내 컴퓨터 중 하나에서만 실행될 수 있습니다.

Windows *MSCS의 PostOnlineCommand 및 PreOfflineCommand*

이러한 명령을 사용하여 IBM MQ MSCS 지원을 다른 시스템과 통합합니다. 이러한 명령을 사용하여 몇 가지 제한과 함께 IBM MQ 명령을 발행할 수 있습니다.

매개변수에 이러한 명령을 `IBM MQ MSCS`유형의 자원에 지정하십시오. 이러한 명령을 사용하여 IBM MQ MSCS 지원과 다른 시스템 또는 프로시저를 통합할 수 있습니다. 예를 들어, 메일 메시지를 송신하거나, 페이지를 활성화하거나, 다른 모니터링 시스템이 캡처할 일부 다른 양식의 경보를 생성하는 프로그램의 이름을 지정할 수 있습니다.

자원이 오프라인에서 온라인으로 변경될 때 PostOnlineCommand가 호출됩니다. PreOfflineCommand는 온라인에서 오프라인으로 변경 시 호출됩니다. 호출되는 경우 이러한 명령은 기본적으로 Windows 시스템 디렉토리에서 실행됩니다. IBM MQ은(는) 32비트 자원 모니터 프로세스를 사용하므로 Windows 64 비트 시스템에서는 `\Windows\system32` 디렉토리가 아니라 `\Windows\SysWOW64` 디렉토리입니다. 자세한 정보는 Windows x64 환경에서의 파일 경로 재지정에 대한 Microsoft 문서를 참조하십시오. 두 명령 모두 MSCS 클러스터 서비스를 실행하는 데 사용되는 사용자 계정에서 실행되며 비동기로 호출되고 IBM MQ MSCS 지원은 이들 명령이 완료될 때까지 대기하지 않고 작업을 계속합니다. 이는 추가 클러스터 조작을 차단하거나 지연시킬 수 있는 위험을 제거합니다.

또한 이러한 명령을 사용하여 IBM MQ 명령(예: 요청자 채널 재시작)을 발행할 수 있습니다. 그러나 이러한 명령은 큐 관리자의 상태가 변하는 시점에 실행되므로 장기 실행 기능을 수행하는 것을 목적으로 하지 않으며 큐 관리자의 현재 상태에 대해 가정하지 말아야 합니다. 큐 관리자가 온라인 설정된 후 즉시 관리자가 오프라인 명령을 발행하는 것이 가능할 수 있습니다.

큐 관리자의 상태에 종속되는 프로그램을 실행하려는 경우 해당 프로그램을 큐 관리자 자원과 같은 MSCS 그룹에 배치하고 큐 관리자 자원에 종속되도록 설정하여 `MSCS Generic Application` 자원 유형의 인스턴스를 작성하는 것을 고려해 보십시오.

Windows *MSCS의 선호 노드 사용*

MSCS에서 활성/활성 모드를 사용할 때 각 큐 관리자에 대해 선호 노드를 구성하는 것이 유용할 수 있습니다. 그러나 일반적으로 선호 노드를 설정하지 않고 수동 페일백에 의존하는 것이 더 좋습니다.

상대적으로 stateless한 일부 다른 자원과 다르게 큐 관리자는 한 노드에서 다른 노드로 장애 복구(또는 페일백)하는 동안 약간의 시간이 걸릴 수 있습니다. 불필요한 정지를 피하려면 큐 관리자를 복구된 노드로 페일백하기 전에 복구된 노드를 테스트하십시오. 이는 `immediate` 페일백 설정이 사용되지 않도록 팝니다. 하루 중 특정 시간대에 발생하도록 페일백을 구성할 수 있습니다.

가장 안전한 경로는 노드가 완전히 복구되었다고 확신할 때 큐 관리자를 필요한 노드에 수동으로 이동하는 것일 수 있습니다. 이 경우 `preferred node` 옵션의 사용을 피할 수 있습니다.

Windows *MSCS에서 설치할 때 COM+ 오류*

새로 설치된 MSCS 클러스터에서 IBM MQ를 설치할 때 애플리케이션 이벤트 로그에 보고된 소스 COM+ 및 이벤트 ID 4691에 관한 오류를 찾을 수 있습니다.

이는 MSDTC(Microsoft Distributed Transaction Coordinator)가 MSCS(Microsoft Cluster Server) 환경에서 실행되도록 구성된 경우 이러한 환경에서 IBM MQ를 실행하려고 하는 것을 의미합니다. 클러스터된 환경에서 MSDTC를 구성하는 것에 대한 정보는 Microsoft 문서를 참조하십시오.

Windows *MSCS* 유틸리티 프로그램에 대한 지원

명령 프롬프트에서 실행할 수 있는 MSCS 유틸리티 프로그램에 대한 IBM MQ 지원의 목록입니다.

MSCS에 대한 IBM MQ 지원은 다음 유틸리티 프로그램을 포함합니다.

자원 유형 등록/등록 해제

`haregtyp.exe`

IBM MQ MSCS 자원 유형을 등록 해제한 후에는 해당 유형의 자원을 더 이상 작성할 수 없습니다. MSCS에서는 클러스터 내에 해당 자원 유형의 인스턴스가 아직 있을 경우 이 자원 유형을 등록 해제할 수 없습니다.

1. 392 페이지의 『MSCS에서 큐 관리자를 오프라인으로 가져오기』에 설명된 대로 MSCS 클러스터 관리자를 사용하여 MSCS 제어 하에 실행되고 있는 큐 관리자를 오프라인으로 가져와서 해당 큐 관리자를 중지하십시오.
2. MSCS 클러스터 관리자를 사용하여 자원 인스턴스를 삭제하십시오.
3. 명령 프롬프트에서 다음 명령을 입력하여 자원 유형을 등록 해제하십시오.

```
haregtyp /u
```

유형을 등록(또는 나중에 등록 해제)하려는 경우 명령 프롬프트에 다음 명령을 입력하십시오.

```
haregtyp /r
```

MSCS 라이브러리를 성공적으로 등록한 후에는 IBM MQ 설치 후 아직 시스템을 다시 시작하지 않은 경우 시스템을 다시 시작해야 합니다.

MSCS 스토리지로 큐 관리자 이동

`hamvmqm.exe`

385 페이지의 『MSCS 스토리지로 큐 관리자 이동』를 참조하십시오.

노드에서 큐 관리자 삭제

`hadltmqm.exe`

클러스터에 있는 큐 관리자를 한 노드에서 다른 노드로 이동했지만 이제 영구 삭제하려고 하는 경우를 고려해 보십시오. IBM MQ 탐색기를 사용하여 현재 위치한 노드에서 이 큐 관리자를 삭제하십시오. 이 큐 관리자에 대한 레지스트리 항목이 여전히 다른 컴퓨터에 남아 있습니다. 이러한 레지스트리 항목을 삭제하려면 해당 컴퓨터의 프롬프트에 다음 명령을 입력하십시오.

```
hadltmqm /m qmname
```

여기서, qmname은 제거할 큐 관리자의 이름입니다.

설정 세부사항 확인 및 저장

`amqmsysn.exe`

이 유틸리티는 IBM 지원을 호출할 경우 요청되는 것과 같은 IBM MQ MSCS 지원 설정의 전체 세부사항을 보여주는 대화 상자를 표시합니다. 세부사항을 파일에 저장하는 옵션이 있습니다.

Multi 다중 인스턴스 큐 관리자

다중 인스턴스 큐 관리자는 여러 서버에서 구성된 동일한 큐 관리자의 인스턴스입니다. 큐 관리자의 한 인스턴스는 활성 인스턴스로 정의되고 다른 인스턴스는 대기 인스턴스로 정의됩니다. 활성 인스턴스가 실패하면 다중 인스턴스 큐 관리자가 대기 서버에서 자동으로 시작됩니다.

다중 인스턴스 큐 관리자 구성 예

397 페이지의 그림 74은 큐 관리자 QM1의 다중 인스턴스 구성 예를 보여줍니다. IBM MQ는 두 서버에 설치되며 그 중 하나는 예비 서버입니다. 하나의 큐 관리자 QM1이 작성되었습니다. QM1의 한 인스턴스는 활성이고 한 서버에서 실행 중입니다. QM1의 다른 인스턴스는 다른 서버에서 대기 상태로 실행 중이고 활성 처리를 수행하지 않지만 활성 인스턴스가 실패할 경우 QM1의 활성 인스턴스에서 인수할 준비를 하고 있습니다.

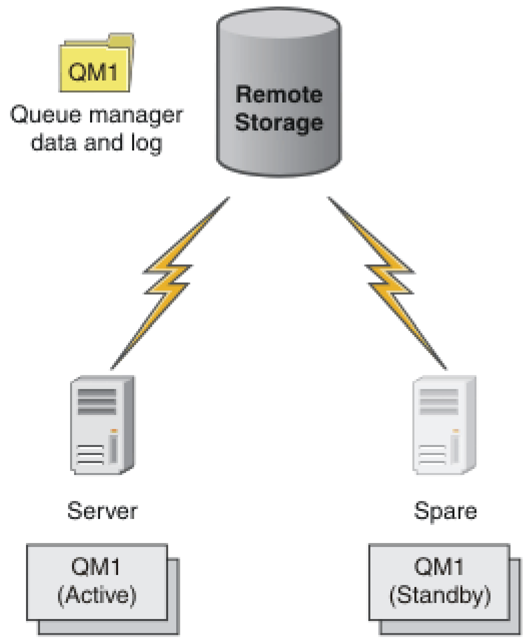


그림 *74.* 다중 인스턴스 큐 관리자

큐 관리자를 다중 인스턴스 큐 관리자로 사용하려고 할 때 **crtmqm** 명령을 사용하여 서버 중 하나에서 단일 큐 관리자를 작성하고 해당 큐 관리자 데이터 및 로그를 공유 네트워크 스토리지에 배치하십시오. 다른 서버에서는 큐 관리자를 다시 작성하는 대신 **addmqinf** 명령을 사용하여 네트워크 스토리지의 큐 관리자 데이터 및 로그에 대한 참조를 작성하십시오.

이제 두 서버 중 하나에서 큐 관리자를 실행할 수 있습니다. 각 서버는 동일한 큐 관리자 데이터 및 로그를 참조합니다. 큐 관리자는 한 개만 있으며 한 번에 한 서버에서만 활성입니다.

큐 관리자는 단일 인스턴스 큐 서버 또는 다중 인스턴스 큐 관리자로서 실행할 수 있습니다. 두 가지 경우 모두 큐 관리자의 한 인스턴스만 실행하고 요청을 처리합니다. 다른 점은 다중 인스턴스 큐 관리자로서 실행할 때 큐 관리자의 활성 인스턴스를 실행하고 있지 않은 서버가 대기 인스턴스로 실행되며 활성 서버가 실패할 경우 자동으로 활성 인스턴스에서 인수할 준비를 합니다.

두 서버에서 큐 관리자를 시작하는 순서로 어느 인스턴스가 먼저 활성화되는지에 대해 제어할 수 있습니다. 큐 관리자 데이터에 대해 읽기/쓰기 잠금을 확보하는 첫 번째 인스턴스가 활성 인스턴스가 됩니다.

활성 인스턴스가 시작된 후에는 제어를 대기로 전송하기 위한 전환 옵션을 사용하여 활성 인스턴스를 중지시켜 이 활성 인스턴스를 다른 서버로 스왑할 수 있습니다.

QM1의 활성 인스턴스는 실행 중일 때 공유 큐 관리자 데이터 및 로그 폴더에 대해 독점 액세스를 가집니다. QM1의 대기 인스턴스는 활성 인스턴스가 실패했을 때 이를 감지하고 활성 인스턴스가 됩니다. 활성 인스턴스가 남긴 상태의 QM1 데이터 및 로그를 인수하고 클라이언트 및 채널로부터의 재연결을 승인합니다.

활성 인스턴스는 대기 인수를 발생시키는 다양한 이유로 인해 실패할 수 있습니다.

- 활성 큐 관리자 인스턴스를 호스팅하는 서버의 실패
- 활성 큐 관리자 인스턴스를 호스팅하는 서버와 파일 시스템 간의 연결 실패
- IBM MQ에서 감지되어 큐 관리자를 종료하는 큐 관리자 프로세스의 무응답

큐 관리자 구성 정보를 다중 서버에 추가하고 활성/대기 쌍으로 실행할 두 개의 서버를 선택할 수 있습니다. 총 인스턴스 수는 2개로 제한됩니다. 두 개의 대기 인스턴스와 한 개의 활성 인스턴스를 가질 수 없습니다.

고가용성 솔루션을 빌드하는 데 필요한 추가 컴포넌트

다중 인스턴스 큐 관리자는 고가용성 솔루션의 한 부분입니다. 유용한 고가용성 솔루션을 빌드하려면 몇 가지 추가 컴포넌트가 필요합니다.

- 활성 큐 관리자 인스턴스 실행을 인수하는 컴퓨터로 IBM MQ 연결을 전송하기 위한 클라이언트 및 채널 다시 연결
- 올바르게 잠금을 관리하고 매체 및 파일 서버 실패에 대한 보호를 제공하는 고성능 공유 네트워크 파일 시스템 (NFS).

  중요사항: NFS 드라이브에서 유지보수를 수행하려면 사용자 환경에서 실행 중인 모든 다중 인스턴스 큐 관리자를 중지해야 합니다. NFS 실패가 발생할 경우 복구하기 위한 큐 관리자 구성 백업이 있는지 확인하십시오.
- 기본 인프라에서 단일 실패 지점을 제거하기 위한 복원 네트워크 및 전원 공급 장치
- 장애 복구를 허용하는 애플리케이션. 특히 트랜잭션 애플리케이션의 작동 및 IBM MQ 큐를 찾아보는 애플리케이션에 대해 주의를 기울일 필요가 있습니다.
- 활성 및 대기 인스턴스가 실행 중인지 확인하고 실패한 활성 인스턴스를 재시작하기 위한 활성 및 대기 인스턴스 모니터링 및 관리. 다중 인스턴스 큐 관리자가 자동으로 재시작되더라도 대기 인스턴스가 실행 중이고 인수할 준비가 되었는지 및 실패한 인스턴스가 새 대기 인스턴스로서 온라인 상태로 되었는지 확인해야 합니다.

IBM MQ MQI clients 및 채널은 대기 큐 관리자가 활성이 되면 자동으로 다시 연결됩니다. 다시 연결 및 고가용성 솔루션의 기타 컴포넌트에 대한 자세한 정보는 관련 주제에서 찾을 수 있습니다. 자동 클라이언트 다시 연결은 IBM MQ classes for Java에서 지원되지 않습니다.

지원 플랫폼

IBM WebSphere MQ 7.0.1 이상에서 지원하는 z/OS 이외 플랫폼에서 다중 인스턴스 큐 관리자를 작성할 수 있습니다.

자동 클라이언트 다시 연결은 IBM WebSphere MQ 7.0.1 이상에서 MQI 클라이언트에 대해 지원됩니다.

다중 인스턴스 큐 관리자 작성

한 서버에서 큐 관리자를 작성하고 다른 한 서버에서 IBM MQ를 구성하여 다중 인스턴스 큐 관리자를 작성합니다. 다중 인스턴스 큐 관리자는 큐 관리자 데이터 및 로그를 공유합니다.

다중 인스턴스 큐 관리자를 작성하는 데 있어 대부분의 수고가 필요한 부분은 공유 큐 관리자 데이터 및 로그 파일을 설정하는 태스크입니다. 네트워크 스토리지에서 공유 디렉토리를 작성해야 하고 작성한 디렉토리를 네트워크 공유를 사용하는 다른 서버가 사용할 수 있도록 설정해야 합니다. 이러한 태스크는 UNIX and Linux 시스템의 루트와 같은 관리 권한이 있는 사람이 수행해야 합니다. 단계는 다음과 같습니다.

1. 데이터 및 로그 파일에 대한 공유를 작성하십시오.
2. 서버에서 큐 관리자를 작성하십시오.
3. 첫 번째 서버에서 **dspmqinf** 명령을 실행하여 큐 관리자 구성 데이터를 수집하고 이를 클립보드에 복사하십시오.
4. 복사한 데이터로 **addmqinf** 명령을 실행하여 두 번째 서버에서 큐 관리자 구성을 작성하십시오.

두 번째 서버에서 큐 관리자를 다시 작성하기 위해 **crtmqm**을 실행하지 않습니다.

파일 액세스 제어

다른 모든 서버에서 사용자 및 그룹 mqm이 공유에 액세스할 수 있음에 주의해야 합니다.

UNIX and Linux에서는 mqm의 `uid` 및 `gid`를 모든 시스템에서 동일하게 작성해야 합니다. mqm의 공통 `uid` 및 `gid`를 설정하기 위해 각 시스템에서 `/etc/passwd`를 편집한 다음 시스템을 다시 시작해야 합니다.

Microsoft Windows에서 큐 관리자 프로세스를 실행하는 사용자 ID에는 큐 관리자 데이터 및 로그 파일이 포함된 디렉토리에 대한 전체 제어 권한이 있어야 합니다. 이 권한을 다음 두 가지 방식으로 구성할 수 있습니다.

1. 대체 보안 프린시펄로 글로벌 그룹을 사용하여 큐 관리자를 작성하십시오. 큐 관리자 데이터 및 로그 파일이 포함된 디렉토리에 대한 전체 제어 액세스 권한을 글로벌 그룹에 부여하십시오. 424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』의 내용을 참조하십시오. 큐 관리자를 실행하는 사용자 ID가 글로벌 그룹의 멤버가 되도록 하십시오. 로컬 사용자를 글로벌 그룹의 멤버로 설정할 수 없으므로 큐 관리자 프로세스는 도메인 사용자 ID로 실행되어야 합니다. 도메인 사용자 ID는 로컬 그룹 mqm의 멤버여야 합니다. 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작

성』 태스크는 이 방식으로 보안 설정된 파일을 사용하여 다중 인스턴스 큐 관리자를 설정하는 방법을 보여줍니다.

2. 로컬 mqm 그룹이 도메인 범위 "도메인 로컬"을 가지도록 도메인 제어기에서 큐 관리자를 작성하십시오. 도메인 로컬 mqm으로 파일 공유를 보안 설정하고 같은 도메인 로컬 mqm 그룹에 있는 큐 관리자의 모든 인스턴스에서 큐 관리자 프로세스를 실행하십시오. 415 페이지의 『Windows 도메인 제어기에서 다중 인스턴스 큐 관리자 작성』 태스크는 이 방식으로 보안 설정된 파일을 사용하여 다중 인스턴스 큐 관리자를 설정하는 방법을 보여줍니다.

구성 정보

각 서버에 대한 IBM MQ 큐 관리자 구성 정보를 수정하여 필요한 수만큼 큐 관리자 인스턴스를 구성하십시오. 각 서버에는 같은 버전의 IBM MQ가 호환 가능한 수정사항 레벨로 설치되어 있어야 합니다. **dspmqinf** 및 **addmqinf** 명령은 추가 큐 관리자 인스턴스를 구성하는 것을 지원합니다. 또는 `mqs.ini` 및 `qm.ini` 파일을 직접 편집할 수 있습니다. 436 페이지의 『Linux에서 다중 인스턴스 큐 관리자 작성』, 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』 및 415 페이지의 『Windows 도메인 제어기에서 다중 인스턴스 큐 관리자 작성』 주제는 다중 인스턴스 큐 관리자를 구성하는 방법을 보여주는 예입니다.

Windows, UNIX and Linux 시스템에서 단일 `mqs.ini` 파일을 네트워크 공유에 배치하고 이 파일을 가리키도록 **AMQ_MQS_INI_LOCATION** 환경 변수를 설정하여 공유할 수 있습니다.

제한

1. 동일한 큐 관리자의 다중 인스턴스를 같은 운영 체제 아키텍처 및 엔디언을 가지는 서버에서만 구성하십시오. 예를 들어, 두 시스템 모두 32비트거나 64비트여야 합니다.
2. 모든 IBM MQ 설치는 릴리스 레벨 7.0.1 이상이어야 합니다.
3. 일반적으로 활성 및 대기 설치는 같은 유지보수 레벨로 유지보수됩니다. 각 업그레이드에 대해 유지보수 지시사항을 참조하여 모든 설치를 함께 업그레이드해야 하는지 여부를 확인하십시오.

   활성 및 수동 큐 관리자의 유지보수 레벨이 동일해야 함에 유의하십시오.
4. 같은 IBM MQ 사용자, 그룹 및 액세스 제어 메커니즘으로 구성되는 큐 관리자 사이에서만 큐 관리자 데이터 및 로그를 공유합니다. IBM i 예를 들어 Linux 서버에서의 네트워크 공유 설정에는 UNIX and Linux 큐 관리자에 대한 별도의 큐 관리자 데이터 및 로그가 포함될 수 있지만 IBM i에서 사용하는 큐 관리자 데이터는 포함될 수 없습니다.

   IBM i 공유가 서로 다른 경우, IBM i 및 UNIX 시스템용 동일한 네트워크 스토리지에서 다중 공유를 작성할 수 있습니다. 소유자에 따라 다른 공유를 제공할 수 있습니다. The restriction is a consequence of the different names used for the IBM MQ users and groups between UNIX and IBM i. 사용자 및 그룹이 같은 `uid` 및 `gid`를 가질 수 있다는 사실로 제한이 완화되지는 않습니다.
5. UNIX and Linux 시스템에서 소프트 마운트가 아닌 인터럽트 가능한 하드 마운트로 네트워크 스토리지에서 공유 파일 시스템을 구성하십시오. 인터럽트 가능한 하드 마운트는 큐 관리자가 시스템 호출에 의해 인터럽트될 때까지 큐 관리자가 정지하도록 강제 실행합니다. 소프트 마운트는 서비스 실패 후의 데이터 일관성을 보장하지 않습니다.
6. 공유 로그 및 데이터 디렉토리는 FAT 또는 NFSv3 파일 시스템에 저장할 수 없습니다. Windows에서 다중 인스턴스 큐 관리자의 경우 네트워크 스토리지는 Windows 네트워크에서 사용되는 CIFS(Common Internet File System) 프로토콜에 의해 액세스 가능해야 합니다.
7. z/OS z/OS는 다중 인스턴스 큐 관리자를 지원하지 않습니다. 큐 공유 그룹을 사용하십시오.

   다시 연결 가능한 클라이언트는 z/OS 큐 관리자에서 작동합니다.

Windows *Windows* 도메인 및 다중 인스턴스 큐 관리자
Windows의 다중 인스턴스 큐 관리자에서는 해당 데이터 및 로그가 공유되어야 합니다. 공유는 여러 서버 또는 워크스테이션에서 실행 중인 큐 관리자의 모든 인스턴스가 액세스할 수 있어야 합니다. 큐 관리자 및 공유를 Windows 도메인의 부분으로 구성하십시오. 큐 관리자는 도메인 워크스테이션 또는 서버 또는 도메인 제어기에서 실행할 수 있습니다.

다중 인스턴스 큐 관리자를 구성하기 전에 427 페이지의 『Windows에서 비공유 큐 관리자 데이터와 로그 디렉토리 및 파일 보안 설정』 및 424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』의 내용을 읽고 큐 관리자 데이터 및 로그 파일에 대한 액세스를 제어하는 방법을 검토하십시오. Windows 도메인에서 다중 인스턴스 큐 관리자의 공유 디렉토리 설정으로 바로 이동하려는 경우 이 주제는 교육적입니다. 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』의 내용을 참조하십시오.

도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 실행

IBM WebSphere MQ 7.1에서 다중 인스턴스 큐 관리자는 워크스테이션 또는 도메인의 멤버인 서버에서 실행됩니다. IBM WebSphere MQ 7.1 이전에서는 다중 인스턴스 큐 관리자가 도메인 제어기에서만 실행되었습니다. 400 페이지의 『도메인 제어기에서 다중 인스턴스 큐 관리자 실행』의 내용을 참조하십시오. Windows에서 다중 인스턴스 큐 관리자를 실행하려면 도메인 제어기, 파일 서버 및 같은 도메인에 연결된 같은 큐 관리자를 실행하는 두 개의 워크스테이션 또는 서버가 필요합니다.

도메인 내 임의의 서버 또는 워크스테이션에서 다중 인스턴스 큐 관리자를 실행할 수 있게 한 변경은 이제 추가 보안 그룹으로 큐 관리자를 작성할 수 있다는 것입니다. 추가 보안 그룹은 **crtmqm** 명령에서 -a 매개변수로 전달됩니다. 이 그룹으로 큐 관리자 데이터 및 로그를 포함하는 디렉토리를 보안 설정합니다. 큐 관리자 프로세스를 실행하는 사용자 ID는 이 그룹의 멤버여야 합니다. 큐 관리자가 디렉토리에 액세스하면 Windows에서는 사용자 ID가 디렉토리에 액세스하기 위해 가지고 있는 권한을 검사합니다. 그룹 및 사용자 ID 도메인 범위를 둘 다 지정하면 큐 관리자 프로세스를 실행하는 사용자 ID는 글로벌 그룹의 신임 정보를 가집니다. 큐 관리자가 다른 서버에서 실행되고 있는 경우 큐 관리자 프로세스를 실행하는 사용자 ID는 같은 신임 정보를 가질 수 있습니다. 이 사용자 ID는 같을 필요는 없습니다. 이 사용자 ID는 로컬 mqm 그룹의 멤버일 뿐만 아니라 대체 보안 그룹의 멤버여야 합니다.

다중 인스턴스 큐 관리자를 작성하는 태스크는 한 가지 변경을 제외하고 IBM WebSphere MQ 7.0.1에서와 같습니다. **crtmqm** 명령의 매개변수에 추가 보안 그룹 이름을 추가해야 합니다. 이 태스크는 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』에 설명되어 있습니다.

도메인 및 도메인 서버와 워크스테이션을 구성하는 데에는 여러 단계가 필요합니다. Windows에서 해당 데이터 및 로그 디렉토리에 대한 큐 관리자의 액세스에 권한을 부여하는 방법에 대해 이해해야 합니다. 해당 로그 및 데이터 파일에 액세스하기 위한 권한이 큐 관리자 프로세스에 부여되는 방식에 대해 확신할 수 없으면 427 페이지의 『Windows에서 비공유 큐 관리자 데이터와 로그 디렉토리 및 파일 보안 설정』 주제를 읽어보십시오. 이 주제는 필요한 단계를 이해하는 데 유용한 두 개의 태스크를 포함합니다. 이 태스크는 429 페이지의 『로컬 mqm 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기』 및 432 페이지의 『대체 로컬 보안 그룹이 권한을 부여한 데이터 및 로그 파일 읽기 및 쓰기』입니다. 다른 주제인 424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』에서는 큐 관리자 데이터 및 로그 파일을 포함하는 공유 디렉토리를 대체 보안 그룹을 사용하여 보안 설정하는 방식을 설명합니다. 이 주제는 Windows 도메인 설정, 파일 공유 작성, IBM MQ for Windows 설치 및 공유를 사용하도록 큐 관리자 구성의 네 개 태스크를 포함합니다. 이러한 태스크는 다음과 같습니다.

1. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』.
2. 407 페이지의 『Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치』.
3. 409 페이지의 『Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성』.
4. 412 페이지의 『대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기』.

도메인을 사용하여 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』 태스크를 수행할 수 있습니다. 이러한 태스크를 수행하여 지식을 프로덕션 도메인에 전송하기 전에 다중 인스턴스 큐 관리자 설정에 대해 알아보십시오.

도메인 제어기에서 다중 인스턴스 큐 관리자 실행

IBM WebSphere MQ 7.0.1에서 다중 인스턴스 큐 관리자는 도메인 제어기에서만 실행되었습니다. 큐 관리자 데이터는 도메인 mqm 그룹을 사용하여 보안 설정할 수 있습니다. 424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』 주제에서 설명한 대로 워크스테이션 또는 서버에서 로컬 mqm 그룹으로 보안 설정된 디렉토리는 공유할 수 없습니다. 그러나 도메인 제어기에서는 모든 그룹 및 프린시펄에 도메인 범위가 있습니다. 도메인 컨트롤러에 IBM MQ for Windows 를 설치하는 경우, 큐 관리자 데이터 및 로그 파일은 공유할 수 있는 도메인 mqm 그룹으로 보안됩니다. 415 페이지의 『Windows 도메인 제어기에서 다중 인스턴스 큐 관리자 작성』 태스크의 단계에 따라 도메인 제어기에서 다중 인스턴스 큐 관리자를 구성하십시오.

관련 정보
권한 부여 및 액세스 제어 관리
Windows Server 클러스터를 도메인 제어기로 사용하는 방법

Windows *Windows*의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성

이 예에서는 Windows 도메인의 일부인 워크스테이션 또는 서버의 Windows에서 다중 인스턴스 큐 관리자를 설정하는 방법을 보여줍니다. 이 서버가 반드시 도메인 제어기일 필요는 없습니다. 이 설정에서는 프로덕션 스케일이 아닌 관련된 개념을 보여줍니다. 이 예는 Windows Server 2008을 기반으로 합니다. 다른 버전의 Windows Server에서는 단계가 다를 수 있습니다.

프로덕션 스케일 구성에서는 기존 도메인에 맞게 구성을 조정해야 할 수 있습니다. 예를 들어, 여러 공유에 권한을 부여하고 큐 관리자를 실행하는 사용자 ID를 그룹화할 수 있도록 여러 도메인 그룹을 정의할 수 있습니다.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: Windows Server 2008 도메인 제어기. *Sun*, *mars* 및 *venus*가 들어 있는 *wmq.example.com* 도메인을 소유합니다. 설명을 위해 파일 서버로도 사용됩니다.

mars
: 첫 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

venus
: 두 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

Windows에서는 큐 관리자 데이터 및 로그 파일을 저장하려는 파일 시스템을 확인할 필요가 없습니다. 확인 프로시저인 공유 파일 시스템 작동 확인은 UNIX and Linux에 적용됩니다. Windows에서는 검사가 항상 성공적입니다.

다음 태스크에 있는 단계를 수행하십시오. 태스크는 도메인 컨트롤러 및 도메인을 작성하고, 하나의 서버에 IBM MQ for Windows을(를) 설치하며, 데이터 및 로그 파일에 대한 파일 공유를 작성합니다. 기존 도메인 제어기를 구성하고 있는 경우 새 Windows Server 2008의 단계를 시험해 보는 것이 유용할 수 있습니다. 사용자 도메인에 단계를 적응시킬 수 있습니다.

1. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』.
2. 407 페이지의 『Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치』.
3. 409 페이지의 『Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성』.
4. 412 페이지의 『대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기』.

이 태스크 정보

이 태스크는 큐 관리자의 인스턴스를 실행하기 위해 도메인에서 도메인 제어기 및 두 개의 서버를 구성하는 태스크 시퀀스 중 하나입니다. 이 태스크에서는 큐 관리자 *QMGR*의 또 다른 인스턴스를 실행하도록 두 번째 서버 *venus*를 구성합니다. 이 태스크의 단계에 따라 큐 관리자 *QMGR*의 두 번째 인스턴스를 작성하고 인스턴스가 작동하는지 테스트합니다.

이 태스크는 이전 절에 있는 네 개의 태스크와는 별도입니다. 이 태스크에는 단일 인스턴스 큐 관리자를 다중 인스턴스 큐 관리자로 변환하는 단계가 포함됩니다. 다른 모든 단계는 단일 또는 다중 인스턴스 큐 관리자와 공통입니다.

프로시저

1. IBM MQ for Windows를 실행하도록 두 번째 서버를 구성하십시오.

a) 407 페이지의 『Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치』 태스크의 단계를 수행하여 두 번째 도메인 서버를 작성하십시오. 이 태스크 순서에서 두 번째 서버를 *venus*라고 합니다.

팁: 두 서버 각각에서 IBM MQ에 대해 동일한 설치 기본값을 사용하여 두 번째 설치를 작성하십시오. 기본값이 다를 경우 IBM MQ 구성 파일 `mqs.ini`의 *QMGR* **QueueManager** 스탠자에서 `Prefix` 및 `InstallationName` 변수를 조정해야 할 수 있습니다. 이 변수는 각 서버에서 설치 및 큐 관리자마다 다를 수 있는 경로를 나타냅니다. 모든 서버에서 경로가 동일하게 유지되는 경우 다중 인스턴스 큐 관리자 구성이 더 간단해집니다.

2. *venus*에서 두 번째 *QMGR* 인스턴스를 작성하십시오.

a) *mars*의 *QMGR*이 존재하지 않는 경우 412 페이지의 『대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기』 태스크를 수행하십시오.

b) *venus*에서 `Prefix` 및 `InstallationName` 매개변수 값이 올바른지 확인하십시오.

*mars*에서 **dspmqinf** 명령을 실행하십시오.

```
dspmqinf QMGR
```

시스템이 다음과 같이 응답합니다.

```
QueueManager:
Name=QMGR
Directory=QMGR
Prefix=C:\ProgramData\IBM\MQ
DataPath=\\sun\wmq\data\QMGR
InstallationName=Installation1
```

c) **QueueManager** 스탠자의 기계 판독 가능(machine-readable) 양식을 클립보드에 복사하십시오.

*mars*에서 `-o command` 매개변수와 함께 **dspmqinf** 명령을 다시 실행하십시오.

```
dspmqinf -o command QMGR
```

시스템이 다음과 같이 응답합니다.

```
addmqinf -s QueueManager -v Name=QMGR
-v Directory=QMGR -v Prefix="C:\ProgramData\IBM\MQ"
-v DataPath=\\sun\wmq\data\QMGR
```

d) *venus*에서 클립보드의 **addmqinf** 명령을 실행하여 *venus*에 큐 관리자 인스턴스를 작성하십시오.

`Prefix` 또는 `InstallationName` 매개변수의 차이를 수용할 수 있도록 필요한 경우 명령을 조정하십시오.

```
addmqinf -s QueueManager -v Name=QMGR
-v Directory=QMGR -v Prefix="C:\ProgramData\IBM\MQ"
-v DataPath=\\sun\wmq\data\QMGR
```

```
IBM MQ configuration information added.
```

3. *venus*에서 대기 인스턴스를 허용하여 큐 관리자 *QMGR*을 시작하십시오.

a) *mars*에서 *QMGR*이 중지되었는지 확인하십시오.

*mars*에서 **dspmq** 명령을 실행하십시오.

```
dspmq -m QMGR
```

시스템 응답은 큐 관리자가 중지된 방식에 따라 다릅니다. 예를 들면, 다음과 같습니다.

```
C:\Users\Administrator>dspmq -m QMGR
QMNAME(QMGR) STATUS(Ended immediately)
```

b) *venus*에서 **strmqm** 명령을 실행하여 *QMGR*을 시작하고 대기를 허용하십시오.

```
strmqm -x QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using 7.1.0.0.
```

결과

다중 인스턴스 큐 관리자 스위치를 테스트하려면 다음 단계를 수행하십시오.

1. *mars*에서 **strmqm** 명령을 실행하여 *QMGR*을 시작하고 대기를 허용하십시오.

```
strmqm -x QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
A standby instance of queue manager 'QMGR' has been started.
The active instance is running elsewhere.
```

2. *venus*에서 **endmqm** 명령을 실행하십시오.

```
endmqm -r -s -i QMGR
```

*venus*에 대해 시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended, permitting switchover to
a standby instance.
```

*mars*에 대한 시스템 응답은 다음과 같습니다.

```
dspmq
QMNAME(QMGR) STATUS(Running as standby)
C:\Users\wmquser2>dspmq
QMNAME(QMGR) STATUS(Running as standby)
```

```
C:\Users\wmquser2>dspmq
QMNAME(QMGR)   STATUS(Running)
```

다음에 수행할 작업

샘플 프로그램을 사용하여 다중 인스턴스 큐 관리자를 확인하려면 422 페이지의 『Windows에서 다중 인스턴스 큐 관리자 확인』의 내용을 참조하십시오.

Windows *Windows*에서 *Active Directory* 및 *DNS* 도메인 작성

이 태스크에서는 *sun*이라고 하는 Windows 2008도메인 제어기에서 도메인 *wmq.example.com*을 작성합니다. 한 명의 사용자와 적절한 권한을 사용하여 도메인에서 `Domain mqm` 글로벌 그룹을 구성합니다.

프로덕션 스케일 구성에서는 기존 도메인에 맞게 구성을 조정해야 할 수 있습니다. 예를 들어, 여러 공유에 권한을 부여하고 큐 관리자를 실행하는 사용자 ID를 그룹화할 수 있도록 여러 도메인 그룹을 정의할 수 있습니다.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: Windows Server 2008 도메인 제어기. *Sun*, *mars* 및 *venus*가 들어 있는 *wmq.example.com* 도메인을 소유합니다. 설명을 위해 파일 서버로도 사용됩니다.

mars
: 첫 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

venus
: 두 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

1. 태스크 단계는 설치되었지만 어떠한 역할도 구성되지 않은 Windows Server 2008과 일치합니다. 기존 도메인 제어기를 구성하고 있는 경우 새 Windows Server 2008의 단계를 시험해 보는 것이 유용할 수 있습니다. 사용자 도메인에 단계를 적응시킬 수 있습니다.

이 태스크 정보

이 태스크에서는 새 도메인 제어기에서 Active Directory 및 DNS 도메인을 작성합니다. 그런 다음 도메인에 조인한 다른 서버 및 워크스테이션에서 IBM MQ를 설치할 준비가 되도록 구성합니다. Windows 도메인을 작성하기 위해 Active Directory를 설치하고 구성하는 것에 익숙하지 않은 경우 이 태스크를 따르십시오. 다중 인스턴스 큐 관리자 구성을 작성하기 위해서는 Windows 도메인을 작성해야 합니다. 이 태스크는 Windows 도메인을 구성하기 위한 최적의 방법을 안내하기 위한 것은 아닙니다. 프로덕션 환경에서 다중 인스턴스 큐 관리자를 배치하려면 Windows 문서를 참조해야 합니다.

태스크 중 다음 단계를 수행하십시오.

1. Active Directory 설치
2. 도메인 추가
3. DNS에 도메인 추가
4. 글로벌 그룹 `Domain mqm`을 작성하여 적절한 권한 부여
5. 사용자를 추가하고 이 사용자를 글로벌 그룹 `Domain mqm`의 멤버로 설정합니다.

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

이 태스크에서는 도메인 제어기 호스트 이름이 *sun*이고, 두 IBM MQ 서버는 *mars* 및 *venus*입니다. 도메인은 *wmq.example.com*이라고 합니다. 태스크에서 이탤릭체로 표시된 이름은 모두 사용자가 선택한 이름으로 바꿀 수 있습니다.

프로시저

1. 로컬 또는 `Workgroup` 관리자로 도메인 컨트롤러 (*sun*) 에 로그온하십시오.

   서버가 도메인 제어기로 이미 구성되어 있는 경우 도메인 관리자로 로그온해야 합니다.
2. Active Directory 도메인 서비스 마법사를 실행하십시오.
   a) **시작** > **실행...** `dcpromo` 를 입력하고 **확인**을 클릭하십시오.

   Active Directory 2진 파일이 아직 설치되지 않은 경우 Windows에서 파일을 자동으로 설치합니다.
3. 마법사의 첫 번째 창에서 **고급 모드 설치 사용** 선택란을 지워진 상태로 두십시오. **다음** > **다음**을 클릭하고 **새 포리스트에서 새 도메인 작성** > **다음**을 클릭하십시오.
4. **포리스트 루트 도메인의 FQDN** 필드에 *wmq.example.com* 을 입력하십시오. **다음**을 클릭하십시오.
5. 포레스트 기능 레벨 설정 창에서 **포레스트 기능 레벨** > **다음**목록에서 **Windows Server 2003**이상을 선택하십시오.

   IBM MQ에서 지원되는 Windows Server의 가장 오래된 레벨은 Windows Server 2003입니다.
6. 옵션: 도메인 기능 레벨 설정 창에서 **도메인 기능 레벨** > **다음**의 목록에서 **Windows Server 2003**이상을 선택하십시오.

   이 단계는 포리스트 가동 레벨을 **Windows Server 2003**으로 설정한 경우에만 필요합니다.
7. **DNS 서버**가 추가 옵션으로 선택된 상태로 추가 도메인 제어기 옵션 창이 열립니다. **다음** 및 **예**를 클릭하여 경고 창을 지우십시오.

   팁: DNS 서버가 이미 설치된 경우에는 이 옵션이 표시되지 않습니다. 이 태스크를 정확히 수행하려면 이 도메인 제어기에서 모든 역할을 제거하고 다시 시작하십시오.
8. `Database`, `Log Files` 및 `SYSVOL` 디렉토리를 변경하지 않은 채로 두십시오. **다음**을 클릭하십시오.
9. 디렉토리 서비스 복원 모드 관리자 비밀번호 창의 **비밀번호** 및 **비밀번호 확인** 필드에 비밀번호를 입력하십시오. **다음** > **다음**을 클릭하십시오. 마법사 창을 종료하려면 **완료 시 시스템 다시 시작**을 선택하십시오.
10. 도메인 제어기 시스템이 다시 시작되면 *wmq*`\Adminstrator`로 로그온하십시오.

    서버 관리자가 자동으로 시작됩니다.
11. *wmq.example.com*`\Users` 폴더를 여십시오.
    a) **서버 관리자** > **역할** > **Active Directory 도메인 서비스** > ***wmq.example.com*** > **사용자**를 여십시오.
12. **사용자** > **새로 작성** > **그룹**에서 마우스의 오른쪽 단추를 클릭하십시오.
    a) **그룹 이름** 필드에 그룹 이름을 입력하십시오.

    참고: 선호되는 그룹 이름은 `Domain mqm`입니다. 표시된 그대로 입력하십시오.

    - 그룹 `Domain mqm`을 호출하면 도메인 워크스테이션 또는 서버에서의 "IBM MQ 준비" 마법사 작동이 수정됩니다. 이로 인해 " IBM MQ 준비 " 마법사가 자동으로 `Domain mqm` 그룹을 도메인에 있는 새 IBM MQ 의 각 새 설치에 있는 로컬 mqm 그룹에 추가합니다.
    - 도메인에서 `Domain mqm` 글로벌 그룹 없이 워크스테이션 또는 서버를 설치할 수 있습니다. 이를 수행하는 경우, `Domain mqm` 그룹과 동일한 특성을 가진 그룹을 정의해야 합니다. IBM MQ가 도메인에 설치되는 모든 경우에 해당 그룹 또는 해당 그룹의 구성원인 사용자를 로컬 mqm 그룹의 구성원으로 설정해야 합니다. 여러 그룹에 도메인 사용자를 배치할 수 있습니다. 각 도메인 그룹이 별도로 관리하려는 설치 세트에 해당하는 다중 도메인 그룹을 작성하십시오. 관리하는 설치에 따라 도메인 사용자를 여러 도메인 그룹으로 분할하십시오. 각 도메인 그룹 또는 그룹을 다른 IBM MQ 설치의 로컬 mqm 그룹에 추가하십시오. 도메인 그룹에서 특정 로컬 mqm 그룹의 구성원인 도메인 사용자만이 해당 설치를 위한 큐 관리자를 작성하고 관리하고 실행할 수 있습니다.
    - 도메인의 워크스테이션 또는 서버에 IBM MQ 를 설치할 때 사용자가 지정하는 도메인 사용자는 `Domain mqm` 그룹의 구성원이거나 `Domain mqm` 그룹과 동일한 특성을 사용하여 정의한 대체 그룹의 구성원이어야 합니다.

    b) **글로벌**을 **그룹 범위**로 클릭한 상태로 두거나 **범용**으로 변경하십시오. **보안**을 **그룹 유형**으로 클릭한 상태로 두십시오. **확인**을 클릭하십시오.
13. **허용 그룹 멤버십 읽기** 및 **허용 groupMembershipSAM 읽기** 권한을 `Domain mqm` 글로벌 그룹의 권한에 추가하십시오.

a) 서버 관리자 조치 막대에서 **보기 > 고급 기능**을 클릭하십시오.
b) 서버 관리자 탐색 트리에서 **사용자**를 클릭하십시오.
c) 사용자 창에서 **Domain mqm > 특성**을 마우스 오른쪽 단추로 클릭하십시오.
d) **보안 > 고급 > 추가...**를 클릭하십시오. `Domain mqm`을 입력하고 **이름 확인 > 확인**을 클릭하십시오.

   이름 필드에는 `Domain mqm (`*`domain name`*`\Domain mqm)` 문자열이 미리 입력되어 있습니다.
e) **특성**을 클릭하십시오. **적용 대상** 목록에서 **하위 사용자 오브젝트**를 선택하십시오.
f) **권한** 목록에서 **그룹 멤버십 읽기** 및 **groupMembershipSAM 읽기 허용** 선택란을 선택하고 **확인 > 적용 > 확인 > 확인**을 클릭하십시오.

14. `Domain mqm` 글로벌 그룹에 둘 이상의 사용자를 추가하십시오.

    예제에서 하나의 사용자 (*wmquser1*) 가 IBM MQ 서비스를 실행하고 다른 사용자 (*wmquser2*) 가 대화식으로 사용됩니다.
    도메인 구성에서 대체 보안 그룹을 사용하는 큐 관리자를 작성하려면 도메인 사용자가 필요합니다. 관리자는 **`crtmqm`** 명령을 실행할 수 있는 권한이 있지만 사용자 ID가 관리자가 되는 것만으로는 충분하지 않습니다. 관리자가 될 수 있는 도메인 사용자는 로컬 mqm 그룹의 구성원은 물론 대체 보안 그룹의 구성원이어야 합니다.
    이 예제에서는 `Domain mqm` 글로벌 그룹의 *wmquser1* 및 *wmquser2* 구성원이 되어야 합니다. "IBM MQ 준비" 마법사는 마법사가 실행되는 로컬 mqm 그룹의 멤버로서 `Domain mqm`을 자동 구성합니다.
    단일 컴퓨터에서 IBM MQ를 설치할 때마다 IBM MQ 서비스를 실행하려면 다른 사용자를 제공해야 합니다. 다른 컴퓨터에서 동일한 사용자를 재사용할 수 있습니다.

a) 서버 관리자 탐색 트리에서 **사용자 > 새로 작성 > 사용자**를 클릭하십시오.
b) 새 오브젝트-사용자 창에서 **사용자 로그온 이름** 필드에 *wmquser1* 을 입력하십시오. **이름** 필드에 *WebSphere* 을 입력하고 **성** 필드에 *MQ1* 을 입력하십시오. **다음**을 클릭하십시오.
c) **비밀번호** 및 **비밀번호 확인** 필드에 비밀번호를 입력하고 **다음 로그온 시 사용자가 비밀번호를 변경해야 함** 선택란을 지우십시오. **다음 > 마침**을 클릭하십시오.
d) 사용자 창에서 ***WebSphere MQ* > 그룹에 추가...**를 마우스 오른쪽 단추로 클릭하십시오. `Domain mqm`을 입력하고 **이름 확인 > 확인 > 확인**을 클릭하십시오.
e) a - d 단계를 반복하여 *WebSphere MQ2* 를 *wmquser2* 로 추가하십시오.

15. IBM MQ를 서비스로 실행하십시오.

    IBM MQ를 서비스로 실행한 다음 도메인 사용자(도메인 관리자로부터 확보함)에게 서비스로 실행할 수 있는 액세스 권한을 부여해야 하는 경우 다음 프로시저를 수행하십시오.

a) **시작 > 실행...**을 클릭하십시오.

   명령 `secpol.msc`를 입력하고 **확인**을 클릭하십시오.
b) **보안 설정 > 로컬 정책 > 사용자 권한 지정**을 여십시오.

   정책 목록에서 **서비스로서 로그온 > 특성**을 마우스 오른쪽 단추로 클릭하십시오.
c) **사용자 또는 그룹 추가...**를 클릭하십시오.

   도메인 관리자에서 확보한 사용자 이름을 입력하고 **이름 확인**을 클릭하십시오.
d) Windows 보안 창에서 프롬프트가 표시되면, 충분한 권한을 가진 계정 사용자 또는 관리자의 사용자 이름 및 비밀번호를 입력한 후 **확인 > 적용 > 확인**을 클릭하십시오.

   로컬 보안 정책 창을 닫으십시오.

참고: Windows Server 2008 및 Windows Server 2012에서 UAC(User Account Control)는 기본적으로 사용을 설정됩니다.

UAC 기능은 사용자가 관리자 그룹의 구성원인 경우에도 특정 운영 체제 기능에 대해 사용자가 수행할 수 있는 조치를 제한합니다. 이러한 제한을 극복하려면 적절한 단계를 거쳐야 합니다.

다음에 수행할 작업

다음 407 페이지의 『Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치』로 이동하십시오.

관련 태스크

Windows Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치

Windows Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성

Windows 대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기

Windows *Windows* 도메인의 서버 또는 워크스테이션에 *IBM MQ* 설치

이 태스크에서는 *wmq.example.com* Windows 도메인의 서버 또는 워크스테이션에 IBM MQ을(를) 설치하고 구성합니다.

프로덕션 스케일 구성에서는 기존 도메인에 맞게 구성을 조정해야 할 수 있습니다. 예를 들어, 여러 공유에 권한을 부여하고 큐 관리자를 실행하는 사용자 ID를 그룹화할 수 있도록 여러 도메인 그룹을 정의할 수 있습니다.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: Windows Server 2008 도메인 제어기. *Sun*, *mars* 및 *venus*가 들어 있는 *wmq.example.com* 도메인을 소유합니다. 설명을 위해 파일 서버로도 사용됩니다.

mars
: 첫 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

venus
: 두 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

1. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』의 단계를 수행하여 도메인 *wmq.example.com*에 대해 도메인 제어기 *sun*을 작성하십시오. 이탤릭체로 표시된 이름을 구성에 맞게 변경하십시오.
2. IBM MQ 를 실행할 수 있는 기타 Windows 버전은 Windows 시스템의 하드웨어 및 소프트웨어 요구사항 을 참조하십시오.

이 태스크 정보

이 태스크에서 *wmq.example.com* 도메인의 멤버로 *mars*라고 하는 Windows Server 2008을 구성합니다. IBM MQ를 설치하고 *wmq.example.com* 도메인의 멤버로 실행되도록 설치를 구성합니다.

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

이 태스크에서는 도메인 제어기 호스트 이름이 *sun*이고, 두 IBM MQ 서버는 *mars* 및 *venus*입니다. 도메인은 *wmq.example.com*이라고 합니다. 태스크에서 이탤릭체로 표시된 이름은 모두 사용자가 선택한 이름으로 바꿀 수 있습니다.

프로시저

1. *sun.wmq.example.com* 도메인 제어기를 *mars*에 DNS 서버로 추가하십시오.
   a) *mars*에서 *mars*\Administrator 로 로그온한 후 **시작**을 클릭하십시오.
   b) **네트워크** > **특성** > **네트워크 연결 관리**에서 마우스의 오른쪽 단추를 클릭하십시오.
   c) 네트워크 어댑터에서 마우스의 오른쪽 단추를 클릭하고 **특성**을 클릭하십시오.

      시스템이 로컬 영역 연결 특성 창으로 응답하고 연결에 사용되는 항목을 나열합니다.

d) 로컬 영역 연결 특성 창의 항목 목록에서 **Internet Protocol 버전 4** 또는 **Internet Protocol 버전 6** 을 선택하십시오. **특성** > **고급 ...** 을 클릭하십시오. **DNS** 탭을 클릭하십시오.

e) DNS 서버 주소 아래에서 **추가...**를 클릭하십시오.

f) DNS 서버이기도 한 도메인 제어기의 IP 주소를 입력하고 **추가**를 클릭하십시오.

g) **DNS 접미부 추가** > **추가...**를 클릭하십시오.

h) *wmq.example.com*을 입력하고 **추가**를 클릭하십시오.

i) **이 연결의 DNS 접미부** 필드에 *wmq.example.com*을 입력하십시오.

j) **DNS에 이 연결의 주소 등록** 및 **DNS 등록에 이 연결의 접미부 사용**을 선택하십시오. **확인** > **확인** > **닫기**를 클릭하십시오.

k) 명령 창을 열고 **ipconfig /all** 명령을 입력하여 TCP/IP 설정을 검토하십시오.

2. *mars*에서 컴퓨터를 *wmq.example.com* 도메인에 추가하십시오.

a) **시작**을 클릭하십시오.

b) **컴퓨터** > **특성**에서 마우스 오른쪽의 단추를 클릭하십시오. 컴퓨터 이름, 도메인 및 작업 그룹 설정 부분에서 **설정 변경**을 클릭하십시오.

c) 시스템 특성 창에서 **변경...**을 클릭하십시오.

d) 도메인을 클릭하고 *wmq.example.com*을 입력한 후 **확인**을 클릭하십시오.

e) 컴퓨터가 도메인에 조인하도록 허용하는 권한을 가진 도메인 제어기 관리자의 **사용자 이름** 및 **비밀번호**를 입력하고 **확인**을 클릭하십시오.

f) "*wmq.example.com* 도메인 시작" 메시지에 대한 응답으로 **확인** > **확인** > **닫기** > **지금 다시 시작**을 클릭하십시오.

3. 컴퓨터가 *wmq.example.com* 도메인의 멤버인지 확인하십시오.

a) *sun*에서 *wmq*\Administrator로서 도메인 제어기에 로그온하십시오.

b) **서버 관리자** > **Active Directory 도메인 서비스** > ***wmq.example.com*** > **컴퓨터**를 열고 *mars*가 컴퓨터 창에서 올바로 나열되는지 확인하십시오.

4. *mars*에 IBM MQ for Windows를 설치하십시오.

IBM MQ for Windows 설치 마법사 실행에 대한 자세한 정보는 Windows에서 IBM MQ 서버 설치를 참조하십시오.

a) *mars*에서 로컬 관리자(*mars*\Administrator)로 로그온하십시오.

b) IBM MQ for Windows 설치 매체에서 **Setup** 명령을 실행하십시오.

IBM MQ 런치패드 애플리케이션이 시작됩니다.

c) **소프트웨어 요구사항**을 클릭하여 필수 소프트웨어가 설치되는지 확인하십시오.

d) **네트워크 구성** > **예**를 클릭하여 도메인 사용자 ID를 구성하십시오.

404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』 태스크에서는 이 태스크 세트에 대한 도메인 ID를 구성합니다.

e) **IBM MQ 설치**를 클릭하고 설치 언어를 선택한 다음 IBM MQ 설치 프로그램 시작을 클릭하십시오.

f) 라이센스 계약을 확인하고 **다음** > **다음** > **설치**를 클릭하여 기본 구성을 승인하십시오. 설치가 완료될 때까지 대기한 후 **마침**을 클릭하십시오.

설치 이름을 변경하거나, 다른 컴포넌트를 설치하거나, 큐 관리자 데이터 및 로그를 위한 다른 디렉토리를 구성하거나, 다른 디렉토리에 설치하려고 할 수 있습니다. 이 경우 **일반 설치**가 아닌 **사용자 설치**를 클릭하십시오.

IBM MQ가 설치되었으며 설치 프로그램이 "Prepare IBM MQ " 마법사를 시작합니다.

중요사항: 아직 마법사를 실행하지 마십시오.

5. **Run as a service** 권한을 사용하여 IBM MQ 서비스를 실행할 사용자를 구성하십시오.

로컬 mqm 그룹을 구성할지, Domain mqm 그룹을 구성할지 또는 권한을 사용하여 IBM MQ 서비스를 실행할 사용자를 구성할지 선택하십시오. 이 예에서는 사용자에게 권한을 제공합니다.

a) **시작** > **실행 ...**을 클릭하십시오. **secpol.msc** 명령을 입력하고 **확인**을 클릭하십시오.

b) **보안 설정** > **로컬 정책** > **사용자 권한 지정**을 여십시오. 정책 목록에서 **서비스로서 로그온** > **특성**을 마우스 오른쪽 단추로 클릭하십시오.

c) **사용자 또는 그룹 추가...**를 클릭하십시오. 및 *wmquser1* 를 입력하고 **이름 확인** 을 클릭하십시오.

d) 도메인 관리자의 사용자 이름과 비밀번호를 입력하고 *wmq*\Administrator를 클릭하고 **확인** > **적용** > **확인**을 클릭하십시오. 로컬 보안 정책 창을 닫으십시오.

6. "IBM MQ 준비 " 마법사를 실행하십시오.

   "IBM MQ 준비 " 마법사 실행에 대한 추가 정보는 IBM MQ 준비 마법사로 IBM MQ 구성을 참조하십시오.

   a) IBM MQ 설치 프로그램은 "IBM MQ 준비"를 자동으로 실행합니다.

   마법사를 수동으로 시작하려면 **시작** > **모든 프로그램** > **IBM MQ** 폴더에서 " IBM MQ 준비 " 의 바로 가기를 찾으십시오. 다중 설치 구성에서 IBM MQ의 설치에 해당하는 바로 가기를 선택하십시오.

   b) **다음**을 클릭하고 "네트워크에 Windows 2000 이상 도메인 제어기가 있는지 식별"에 대한 응답으로 **예**가 클릭된 상태로 두십시오.

   c) Windows 도메인 사용자 창에 대한 IBM MQ for Windows의 첫 번째 구성에서 **예** > **다음**을 클릭하십시오.

   d) Windows 도메인 사용자에 대한 IBM MQ for Windows 두 번째 구성 창에서 **도메인** 필드에 *wmq* 를 입력하십시오. **사용자 이름** 필드에 *wmquser1* 를 입력하고 **비밀번호** 필드에 비밀번호를 설정한 경우 비밀번호를 입력하십시오. **다음**을 클릭하십시오.

   이 마법사는 *wmquser1*으로 IBM MQ를 구성하고 시작합니다.

   e) 마지막 마법사 페이지에서 필요에 따라 선택란을 선택하거나 선택 취소하고 **마침**을 클릭하십시오.

다음에 수행할 작업

1. 429 페이지의 『로컬 mqm 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기』 태스크를 실행하여 설치 및 구성이 올바로 작동하는지 확인하십시오.
2. 다중 인스턴스 큐 관리자의 로그 파일 및 데이터를 저장하도록 파일 공유를 구성하려면 409 페이지의 『Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성』 태스크를 수행하십시오.

관련 태스크

Windows Windows에서 Active Directory 및 DNS 도메인 작성

Windows Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성

Windows 대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기

관련 정보

IBM MQ Windows 서비스에 필요한 사용자 권한

Windows *Windows*에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다.

프로덕션 스케일 구성에서는 기존 도메인에 맞게 구성을 조정해야 할 수 있습니다. 예를 들어, 여러 공유에 권한을 부여하고 큐 관리자를 실행하는 사용자 ID를 그룹화할 수 있도록 여러 도메인 그룹을 정의할 수 있습니다.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun

Windows Server 2008 도메인 제어기. *Sun*, *mars* 및 *venus*가 들어 있는 *wmq.example.com* 도메인을 소유합니다. 설명을 위해 파일 서버로도 사용됩니다.

mars

첫 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

venus

두 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

1. 문서에 설명된 그대로 이 태스크를 수행하려면 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』 태스크의 단계를 수행하여 도메인 제어기 *sun*에서 도메인 *sun.wmq.example.com*을 작성합니다. 이탤릭체로 표시된 이름을 구성에 맞게 변경하십시오.

이 태스크 정보

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

이 태스크에서 데이터 및 로그 디렉토리를 포함하는 공유 및 이 공유에 액세스 권한을 부여하기 위한 글로벌 그룹을 작성합니다. 공유에 권한을 부여하는 글로벌 그룹의 이름을 해당 -a 매개변수로 **crtmqm** 명령에 전달합니다. 글로벌 그룹은 이 공유의 사용자를 다른 공유 사용자로부터 분리할 수 있는 유연성을 제공합니다. 이 유연성이 필요하지 않은 경우 새 글로벌 그룹을 작성하는 대신, Domain mqm 그룹으로 공유에 대한 권한을 부여합니다.

이 태스크를 공유하는 데 사용되는 글로벌 그룹은 *wmqha*라고 하며, 공유는 *wmq*라고 합니다. 이러한 공유는 Windows 도메인 *wmq.example.com*에서 도메인 제어기 *sun*에서 정의됩니다. 공유에는 글로벌 그룹 *wmqha*에 대한 전체 제어 권한이 있습니다. 태스크에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

이 태스크에서는 도메인 제어기가 파일 서버와 같은 서버입니다. 실제 애플리케이션에서는 성능 및 가용성을 위해 서로 다른 서버 사이에서 디렉토리 및 파일 서비스를 분리하십시오.

큐 관리자 실행에 사용되는 사용자 ID를 두 그룹의 멤버가 되도록 구성해야 합니다. IBM MQ 서버 및 *wmqha* 글로벌 그룹의 로컬 mqm 그룹의 구성원이어야 합니다.

이 태스크 세트에서 큐 관리자가 서비스로 실행되면 사용자 ID *wmquser1*로 실행됩니다. 따라서 *wmquser1*은 *wmqha*의 멤버여야 합니다. 큐 관리자가 대화식으로 실행되는 경우 사용자 ID *wmquser2*로 실행됩니다. 따라서 *wmquser2*는 *wmqha*의 멤버여야 합니다. *wmquser1* 및 *wmquser2*는 모두 글로벌 그룹 Domain mqm의 멤버입니다. Domain mqm 은 *mars* 및 *venus* IBM MQ 서버의 로컬 mqm 그룹의 구성원입니다. 따라서 *wmquser1* 및 *wmquser2* 는 두 IBM MQ 서버 모두에서 로컬 mqm 그룹의 구성원입니다.

프로시저

1. 도메인 제어기 *sun.wmq.example.com*에 도메인 관리자로 로그온하십시오.
2. 글로벌 그룹 *wmqha*를 작성하십시오.
   a) **서버 관리자** > **역할** > **Active Directory 도메인 서비스** > ***wmq.example.com*** > **사용자**를 여십시오.
   b) *wmq.example.com*\Users 폴더를 여십시오.
   c) **사용자** > **새로 작성** > **그룹**에서 마우스의 오른쪽 단추를 클릭하십시오.
   d) **그룹 이름** 필드에 *wmqha* 을 입력하십시오.
   e) **그룹 범위**로 **글로벌**을 클릭된 상태로 두고 **그룹 유형**으로 **보안**을 클릭된 상태로 두십시오. **확인**을 클릭하십시오.
3. 도메인 사용자 *wmquser1* 및 *wmquser2*를 글로벌 그룹 *wmqha*에 추가하십시오.
   a) 서버 관리자 탐색 트리에서 **사용자**를 클릭한 후 사용자 목록에서 ***wmqha*** > **특성**을 마우스 오른쪽 단추로 클릭하십시오.
   b) *wmqha* 특성 창에서 멤버 탭을 클릭하십시오.
   c) **추가 ...** 를 클릭하십시오. *wmquser1* ; *wmquser2* 를 입력하고 **이름 확인** > **확인** > **적용** > **확인**을 클릭하십시오.
4. 큐 관리자 데이터 및 로그 파일이 포함될 디렉토리 트리를 작성하십시오.

a) 명령 프롬프트를 여십시오.

b) 명령을 입력하십시오.

```
md c:\wmq\data, c:\wmq\logs
```

5. *c:\wmq* 디렉토리 및 공유에 대한 전체 제어 권한을 보유하도록 글로벌 그룹 *wmqha*에 권한을 부여하십시오.

a) Windows 탐색기에서 ***c:\wmq*** > **특성**을 마우스 오른쪽 단추로 클릭하십시오.

b) **보안** 탭을 클릭하고 **고급** > **편집...**을 클릭하십시오.

c) **이 오브젝트의 소유자로부터 상속 가능한 권한을 포함함** 선택란을 지우십시오. Windows 보안 창에서 **복사**를 클릭하십시오.

d) **권한 항목** 목록에서 사용자에 대한 행을 선택하고 **제거**를 클릭하십시오. **권한 항목**의 목록에서 SYSTEM, 관리자 및 CREATOR OWNER에 대한 행은 그대로 두십시오.

e) **추가 ...**를 클릭하십시오. 글로벌 그룹 *wmqha*의 이름을 입력하십시오. **이름 확인** > **확인**을 클릭하십시오.

f) wmq 창에 있는 권한 항목의 **권한** 목록에서 **전체 제어**를 선택하십시오.

g) **확인** > **적용** > **확인** > **확인** > **확인**을 클릭하십시오.

h) Windows 탐색기에서 ***c:\wmq*** > **공유...**를 마우스 오른쪽 단추로 클릭하십시오.

i) **고급 공유 ...** 를 클릭하십시오. **이 폴더 공유** 선택란을 선택하십시오. 공유 이름은 *wmq*로 두십시오.

j) **권한** > **추가 ...**를 클릭하십시오. 글로벌 그룹 *wmqha*의 이름을 입력하십시오. **이름 확인** > **확인**을 클릭하십시오.

k) **그룹 또는 사용자 이름** 목록에서 *wmqha*를 선택하십시오. ***wmqha* 에 대한 권한** 목록에서 **전체 제어** 선택란을 선택하십시오. **적용**을 누르십시오.

l) **그룹 또는 사용자 이름** 목록에서 *Administrators*를 선택하십시오. **관리자 에 대한 권한** 목록에서 **전체 제어** 선택란을 선택하십시오. **적용** > **알았다** > **알았다** > **닫기**을 누르십시오.

다음에 수행할 작업

각 IBM MQ 서버로부터 공유 디렉토리에 파일을 작성하고 읽을 수 있는지 확인하십시오. IBM MQ 서비스 사용자 ID, *wmquser1* 및 대화식 사용자 ID *wmquser2*를 확인하십시오.

1. If you are using remote desktop, you must add *wmq**wmquser1* and *wmquser2* to the local group `Remote Desktop Users` on *mars*.

   a. *mars*에 *wmq*\`Administrator`로서 로그온하십시오.

   b. **`lusrmgr.msc`** 명령을 실행하여 로컬 사용자 및 그룹 창을 여십시오.

   c. **그룹**을 클릭하십시오. **원격 데스크탑 사용자** > **특성** > **추가...**에서 마우스의 오른쪽 단추를 클릭하십시오. *wmquser1 ; wmquser2* 를 입력하고 **이름 확인**을 클릭하십시오.

   d. 도메인 관리자의 사용자 이름과 비밀번호를 입력하고 *wmq*\`Administrator`를 입력하고 **확인** > **적용** > **확인**을 클릭하십시오.

   e. 로컬 사용자 및 그룹 창을 닫으십시오.

2. *wmq**wmquser1*으로 *mars*에 로그온하십시오.

   a. Windows 탐색기 창을 열고 *sun**wmq*에 입력하십시오.

   시스템은 *sun.wmq.example.com*에서 *wmq* 공유를 열어 응답하고 데이터 및 로그 디렉토리를 나열합니다.

   b. 데이터 서브디렉토리에서 파일을 작성하고, 일부 컨텐츠를 추가하고, 읽은 다음 삭제하여 *wmquser1*의 권한을 검사하십시오.

3. *wmq**wmquser2*로 *mars*에 로그온하고 검사를 반복하십시오.

4. 공유 데이터 및 로그 디렉토리를 사용하기 위한 큐 관리자를 작성하려면 다음 태스크를 수행하십시오. 412 페이지의 『대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기』의 내용을 참조하십시오.

관련 태스크

Windows Windows에서 Active Directory 및 DNS 도메인 작성

Windows Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치

Windows 대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기

Windows 대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기

이 태스크에서는 **crtmqm** 명령에서 -a 플래그를 사용하는 방법을 보여줍니다. -a 플래그는 대체 보안 그룹을 사용하여 원격 파일 공유에 있는 해당 로그 및 데이터 파일에 대한 액세스 권한을 큐 관리자에게 제공합니다.

프로덕션 스케일 구성에서는 기존 도메인에 맞게 구성을 조정해야 할 수 있습니다. 예를 들어, 여러 공유에 권한을 부여하고 큐 관리자를 실행하는 사용자 ID를 그룹화할 수 있도록 여러 도메인 그룹을 정의할 수 있습니다.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: Windows Server 2008 도메인 제어기. *Sun*, *mars* 및 *venus*가 들어 있는 *wmq.example.com* 도메인을 소유합니다. 설명을 위해 파일 서버로도 사용됩니다.

mars
: 첫 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

venus
: 두 번째 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

다음 태스크에 있는 단계를 수행하십시오. 태스크는 도메인 컨트롤러 및 도메인을 작성하고, 하나의 서버에 IBM MQ for Windows을(를) 설치하며, 데이터 및 로그 파일에 대한 파일 공유를 작성합니다. 기존 도메인 제어기를 구성하고 있는 경우 새 Windows Server 2008의 단계를 시험해 보는 것이 유용할 수 있습니다. 사용자 도메인에 단계를 적응시킬 수 있습니다.

1. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』.
2. 407 페이지의 『Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치』.
3. 409 페이지의 『Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성』.

이 태스크 정보

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

이 태스크에서는 이 태스크에서는 파일 서버의 원격 디렉토리에 해당 데이터 및 로그를 저장하는 큐 관리자를 작성합니다. 이 예에서는 파일 서버가 도메인 제어기와 동일한 서버입니다. 데이터 및 로그 폴더를 포함하는 디렉토리는 글로벌 그룹 wmqha에 제공된 전체 제어 권한과 공유됩니다.

프로시저

1. 로컬 관리자(*mars*\Administrator)로 도메인 서버(*mars*)에 로그온하십시오.
2. 명령 창을 여십시오.
3. IBM MQ 서비스를 재시작하십시오.

   서비스를 실행하는 사용자 ID가 그에 대해 구성한 추가 보안 신임 정보를 획득할 수 있도록 서비스를 재시작해야 합니다.

다음 명령을 입력하십시오.

```
endmqsvc
strmqsvc
```

시스템이 다음과 같이 응답합니다.

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' ended successfully.
```

및

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' started successfully.
```

4. 큐 관리자를 작성하십시오.

```
crtmqm -a wmq\wmqha -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\sun\wmq\data -ld \\sun\wmq\logs
QMGR
```

글로벌 그룹 *"wmq\wmqha"*의 전체 도메인 이름을 지정하여 대체 보안 그룹 *wmqha*의 *wmq* 도메인을 지정해야 합니다.

공유 *sun**wmq*의 UNC(Universal Naming Convention) 이름을 입력하고 맵핑된 드라이브 참조를 사용하지 마십시오.

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager created.
Directory '\\sun\wmq\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
기본 오브젝트 통계: 74개가 작성되었습니다. 0을 대체했습니다.
Completing setup.
Setup completed.
```

다음에 수행할 작업

메시지를 큐에 넣거나 가져와서 큐 관리자를 테스트하십시오.

1. 큐 관리자를 시작하십시오.

```
strmqm QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using 7.1.0.0.
```

2. 테스트 큐를 작성하십시오.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

시스템이 다음과 같이 응답합니다.

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.


1 : define qlocal(QTEST)
AMQ8006: IBM MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. 샘플 프로그램 **amqsput**을 사용하여 테스트 메시지를 넣으십시오.

```
echo 'A test message' | amqsput QTEST QMGR
```

시스템이 다음과 같이 응답합니다.

```
Sample AMQSPUT0 start
target queue is QTEST
Sample AMQSPUT0 end
```

4. 샘플 프로그램 **amqsget**을 사용하여 테스트 메시지를 가져오십시오.

```
amqsget QTEST QMGR
```

시스템이 다음과 같이 응답합니다.

```
Sample AMQSGET0 start
message A test message
Wait 15 seconds ...
no more messages
Sample AMQSGET0 end
```

5. 큐 관리자를 중지합니다.

```
endmqm -i QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended.
```

6. 큐 관리자를 삭제합니다.

```
dltmqm QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' deleted.
```

7. 작성한 디렉토리를 삭제하십시오.

팁: 명령이 각 파일 또는 디렉토리를 삭제하도록 프롬프트 표시되는 것을 막으려면 /Q 옵션을 명령에 추가하십시오.

```
del /F /S C:\wmq\*.*
rmdir /S C:\wmq
```

관련 태스크

Windows Windows에서 Active Directory 및 DNS 도메인 작성

Windows Windows 도메인의 서버 또는 워크스테이션에 IBM MQ 설치

Windows Windows에서 큐 관리자 데이터 및 로그 파일을 위한 공유 디렉토리 작성

Windows *Windows* 도메인 제어기에서 다중 인스턴스 큐 관리자 작성

이 예에서는 도메인 제어기의 Windows에서 다중 인스턴스 큐 관리자를 설정하는 방법을 보여줍니다. 이 설정에서는 프로덕션 스케일이 아닌 관련된 개념을 보여줍니다. 이 예는 Windows Server 2008을 기반으로 합니다. 다른 버전의 Windows Server에서는 단계가 다를 수 있습니다.

구성에서는 미니 도메인 또는 "domainlet"의 개념을 사용합니다. 도메인 제어기로서 Windows 2000, Windows Server 2003 및 Windows Server 2008 클러스터 노드를 참조하십시오. 기존 도메인에 다중 인스턴스 큐 관리자를 추가하려면 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』의 내용을 참조하십시오.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: 첫 번째 도메인 제어기로 사용되는 Windows Server 2008 서버입니다. 이는 *sun*, *earth* 및 *mars*를 포함하는 *wmq.example.com* 도메인을 정의합니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

earth
: 두 번째 도메인 제어기 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

mars
: 파일 서버로 사용되는 Windows Server 2008입니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

1. Windows에서는 큐 관리자 데이터 및 로그 파일을 저장하려는 파일 시스템을 확인할 필요가 없습니다. 확인 프로시저인 공유 파일 시스템 작동 확인은 UNIX and Linux에 적용됩니다. Windows에서는 검사가 항상 성공적입니다.
2. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』의 단계를 수행하여 첫 번째 도메인 제어기를 작성하십시오.
3. 418 페이지의 『두 번째 Windows 도메인 제어기를 예제 도메인에 추가』의 단계를 수행하여 두 번째 도메인 제어기를 추가하고 두 도메인 제어기 모두에 IBM MQ for Windows를 설치하고 설치를 확인하십시오.
4. 420 페이지의 『예제 도메인에서 Windows 도메인 제어기에 IBM MQ 설치』의 단계를 수행하여 두 도메인 제어기에 IBM MQ를 설치하십시오.

이 태스크 정보

같은 도메인의 파일 서버에서 큐 관리자 로그 및 데이터 디렉토리에 대한 공유를 작성하십시오. 그 다음에는 도메인 제어기 중 하나에서 파일 공유를 사용하는 다중 인스턴스 큐 관리자의 첫 번째 인스턴스를 작성하십시오. 다른 도메인 제어기에서 다른 인스턴스를 작성하고 마지막으로 구성을 확인하십시오. 도메인 제어기에서 파일 공유를 작성할 수 있습니다.

샘플에서 *sun*은 첫 번째 도메인 제어기이고 *earth*는 두 번째 도메인제어기이고 *mars*는 파일 서버입니다.

프로시저

1. 큐 관리자 데이터 및 로그 파일을 포함할 디렉토리를 작성하십시오.
   a) *mars*에서 다음 명령을 입력하십시오.

      ```
      md c:\wmq\data , c:\wmq\logs
      ```

2. 큐 관리자 데이터 및 로그 파일을 포함할 디렉토리를 공유하십시오.

   큐 관리자를 작성하는 데 사용하는 사용자 ID 및 도메인 로컬 그룹 mqm에 대한 전체 제어 액세스를 허용해야 합니다. 이 예에서 `Domain Administrators`의 멤버인 사용자 ID에는 큐 관리자를 작성하기 위한 권한이 있습니다.

   파일 공유는 도메인 제어기와 같은 도메인에 있는 서버에 있어야 합니다. 이 예에서 서버 *mars*는 도메인 제어기와 같은 도메인에 있습니다.

   a) Windows 탐색기에서 ***c:\wmq*** > **특성**을 마우스 오른쪽 단추로 클릭하십시오.
   b) **보안** 탭을 클릭하고 **고급** > **편집...**을 클릭하십시오.
   c) **이 오브젝트의 소유자로부터 상속 가능한 권한을 포함함** 선택란을 지우십시오. Windows 보안 창에서 **복사**를 클릭하십시오.
   d) **권한 항목** 목록에서 사용자에 대한 행을 선택하고 **제거**를 클릭하십시오. **권한 항목**의 목록에서 SYSTEM, 관리자 및 CREATOR OWNER에 대한 행은 그대로 두십시오.
   e) **추가 ...**를 클릭하십시오. 도메인 로컬 그룹 *mqm*의 이름을 입력하십시오. **이름 확인**을 클릭하십시오.
   f) Windows 보안 창에 대한 응답으로 `Domain Administrator`의 이름 및 비밀번호를 입력하고 **확인** > **확인**을 클릭하십시오.
   g) wmq 창에 있는 권한 항목의 **권한** 목록에서 **전체 제어**를 선택하십시오.
   h) **확인** > **적용** > **확인** > **확인** > **확인**을 클릭하십시오.
   i) e - h단계를 반복하여 `Domain Administrators`를 추가하십시오.
   j) Windows 탐색기에서 ***c:\wmq*** > **공유...**를 마우스 오른쪽 단추로 클릭하십시오.
   k) **고급 공유 ...** 를 클릭하십시오. **이 폴더 공유** 선택란을 선택하십시오. 공유 이름은 *wmq*로 두십시오.
   l) **권한** > **추가 ...**를 클릭하십시오. 도메인 로컬 그룹 *mqm* ; `Domain Administrators`의 이름을 입력하십시오. **이름 확인**을 클릭하십시오.
   m) Windows 보안 창에 대한 응답으로 `Domain Administrator`의 이름 및 비밀번호를 입력하고 **확인** > **확인**을 클릭하십시오.

3. 첫 번째 도메인 제어기 *sun*에서 큐 관리자 *QMGR*을 작성하십시오.

   ```
   crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\mars\wmq\data -ld \\mars\wmq\logs QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager created.
   Directory '\\mars\wmq\data\QMGR' created.
   The queue manager is associated with installation 'Installation1'.
   Creating or replacing default objects for queue manager 'QMGR'.
   Default objects statistics : 74 created. 0 replaced. 0 failed.
   Completing setup.
   Setup completed.
   ```

4. 대기 인스턴스를 허용하며, *sun*에서 큐 관리자를 시작하십시오.

   ```
   strmqm -x QMGR
   ```

   시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation 'Installation1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using 7.1.0.0.
```

5. *earth*에서 *QMGR*의 두 번째 인스턴스를 작성하십시오.

   a) Prefix 및 InstallationName 매개변수의 값이 *earth*에 대해 올바른지 확인하십시오.

      *sun*에서 **dspmqinf** 명령을 실행하십시오.

      ```
      dspmqinf QMGR
      ```

      시스템이 다음과 같이 응답합니다.

      ```
      QueueManager:
      Name=QMGR
      Directory=QMGR
      Prefix=C:\ProgramData\IBM\MQ
      DataPath=\\mars\wmq\data\QMGR
      InstallationName=Installation1
      ```

   b) **QueueManager** 스탠자의 기계 판독 가능(machine-readable) 양식을 클립보드에 복사하십시오.

      *sun*에서 -o command 매개변수와 함께 **dspmqinf** 명령을 다시 실행하십시오.

      ```
      dspmqinf -o command QMGR
      ```

      시스템이 다음과 같이 응답합니다.

      ```
      addmqinf -s QueueManager -v Name=QMGR
      -v Directory=QMGR -v Prefix="C:\ProgramData\IBM\MQ"
      -v DataPath=\\mars\wmq\data\QMGR
      ```

   c) *earth*에서 클립보드로부터 **addmqinf** 명령을 실행하여 *earth*에서 큐 관리자의 인스턴스를 작성하십시오.

      Prefix 또는 InstallationName 매개변수의 차이를 수용할 수 있도록 필요한 경우 명령을 조정하십시오.

      ```
      addmqinf -s QueueManager -v Name= QMGR
      -v Directory= QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"
      -v DataPath=\\mars\wmq\data\QMGR
      ```

      ```
      IBM MQ configuration information added.
      ```

6. *earth*에서 큐 관리자의 대기 인스턴스를 시작하십시오.

   ```
   strmqm -x QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' starting.
   ```

```
The queue manager is associated with installation 'Installation1'.
A standby instance of queue manager 'QMGR' has been started. The active
instance is running elsewhere.
```

결과

*sun*에서 *earth*로 큐 관리자가 전환되었는지 확인하십시오.

1. *sun*에서 다음 명령을 실행하십시오.

   ```
   endmqm -i -r -s QMGR
   ```

   시스템이 *sun*에 대해 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ended, permitting switchover to
   a standby instance.
   ```

2. *earth*에서 다음 명령을 반복적으로 입력하십시오.

   ```
   dspmq
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   QMNAME(QMGR) STATUS(Running as standby)
   QMNAME(QMGR) STATUS(Running as standby)
   QMNAME(QMGR) STATUS(Running)
   ```

다음에 수행할 작업

샘플 프로그램을 사용하여 다중 인스턴스 큐 관리자를 확인하려면 422 페이지의 『Windows에서 다중 인스턴스 큐 관리자 확인』의 내용을 참조하십시오.

관련 태스크

418 페이지의 『두 번째 Windows 도메인 제어기를 예제 도메인에 추가』
420 페이지의 『예제 도메인에서 Windows 도메인 제어기에 IBM MQ 설치』

관련 정보

도메인 제어기로서의 Windows 2000, Windows Server 2003 및 Windows Server 2008 클러스터 노드

Windows 두 번째 *Windows* 도메인 제어기를 예제 도메인에 추가

두 번째 도메인 제어기를 *wmq.example.com* 도메인에 추가하여 도메인 제어기 및 파일 서버에서 다중 인스턴스 큐 관리자를 실행하기 위한 Windows 도메인을 구성합니다.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: 첫 번째 도메인 제어기로 사용되는 Windows Server 2008 서버입니다. 이는 *sun*, *earth* 및 *mars*를 포함하는 *wmq.example.com* 도메인을 정의합니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

earth
: 두 번째 도메인 제어기 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

mars
: 파일 서버로 사용되는 Windows Server 2008입니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

1. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』의 단계를 수행하여 도메인 *wmq.example.com*에 대해 도메인 제어기 *sun*을 작성하십시오. 이탤릭체로 표시된 이름을 구성에 맞게 변경하십시오.
2. 기본 작업 그룹 `WORKGROUP`에 있는 서버에 Windows Server 2008을 설치하십시오. 예를 들어, 서버의 이름은 *earth*로 지정됩니다.

이 태스크 정보

이 태스크에서는 *earth*라고 하는 Windows Server 2008을 *wmq.example.com* 도메인의 두 번째 도메인 제어기로 구성합니다.

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

프로시저

1. 도메인 제어기 *sun.wmq.example.com*을 *earth*에 DNS 서버로 추가하십시오.
   a) *earth*에서 *earth*`\Administrator` 로 로그온한 후 **시작**을 클릭하십시오.
   b) **네트워크 > 특성 > 네트워크 연결 관리**에서 마우스의 오른쪽 단추를 클릭하십시오.
   c) 네트워크 어댑터에서 마우스의 오른쪽 단추를 클릭하고 **특성**을 클릭하십시오.

      시스템이 로컬 영역 연결 특성 창으로 응답하고 연결에 사용되는 항목을 나열합니다.
   d) 로컬 영역 연결 특성 창의 항목 목록에서 **Internet Protocol 버전 4** 또는 **Internet Protocol 버전 6** 을 선택하십시오. **특성 > 고급 ...** 을 클릭하십시오. **DNS** 탭을 클릭하십시오.
   e) DNS 서버 주소 아래에서 **추가...**를 클릭하십시오.
   f) DNS 서버이기도 한 도메인 제어기의 IP 주소를 입력하고 **추가**를 클릭하십시오.
   g) **DNS 접미부 추가 > 추가...**를 클릭하십시오.
   h) *wmq.example.com*을 입력하고 **추가**를 클릭하십시오.
   i) **이 연결의 DNS 접미부** 필드에 *wmq.example.com*을 입력하십시오.
   j) **DNS에 이 연결의 주소 등록** 및 **DNS 등록에 이 연결의 접미부 사용**을 선택하십시오. **확인 > 확인 > 닫기**를 클릭하십시오.
   k) 명령 창을 열고 **`ipconfig /all`** 명령을 입력하여 TCP/IP 설정을 검토하십시오.
2. 로컬 또는 `Workgroup` 관리자로 도메인 컨트롤러 (*sun*) 에 로그온하십시오.

   서버가 도메인 제어기로 이미 구성되어 있는 경우 도메인 관리자로 로그온해야 합니다.
3. Active Directory 도메인 서비스 마법사를 실행하십시오.
   a) **시작 > 실행...** `dcpromo` 를 입력하고 **확인**을 클릭하십시오.

   Active Directory 2진 파일이 아직 설치되지 않은 경우 Windows에서 파일을 자동으로 설치합니다.
4. *earth*를 *wmq.example.com* 도메인의 두 번째 도메인 제어기로 구성하십시오.
   a) 마법사의 첫 번째 창에서 **고급 모드 설치 사용** 선택란을 지워진 상태로 두십시오. **다음 > 다음**을 클릭하고 **기존 도메인에 도메인 제어기 추가 > 다음**를 클릭하십시오.
   b) **이 포리스트에 있는 도메인의 이름 입력 ...** 에 *wmq* 를 입력하십시오. 필드. **대체 신임 정보** 단일 선택 단추가 클릭되면 **설정...**을 클릭하십시오. 도메인 관리자의 이름 및 비밀번호를 입력하고 **확인 > 다음 > 다음 > 다음**을 클릭하십시오.

c) 추가 도메인 제어기 옵션 창에서 선택된 **DNS 서버** 및 **글로벌 카탈로그** 옵션을 승인하고 **다음** > **다음**을 클릭하십시오.

d) 디렉토리 서비스 복원 모드 관리자 비밀번호에서 **Password**및 **비밀번호 확인**을 입력하고 **다음** > **다음**을 클릭하십시오.

e) **네트워크 신인 정보**에 대해 프롬프트 표시되면 도메인 관리자의 비밀번호를 입력하십시오. 마법사 창을 종료하려면 **완료 시 시스템 다시 시작**을 선택하십시오.

f) 잠시 후에 DNS 위임에 관한 **DCPromo** 오류가 포함된 창이 열릴 수 있습니다. **확인**을 클릭하십시오. 서버가 다시 시작됩니다.

결과

*earth*가 다시 시작되면 도메인 관리자로 로그온하십시오. wmq.example.com 도메인이 *earth*로 복제되었는지 확인하십시오.

다음에 수행할 작업

IBM MQ 설치를 계속하십시오. 420 페이지의 『예제 도메인에서 Windows 도메인 제어기에 IBM MQ 설치』의 내용을 참조하십시오.

관련 태스크

Windows 예제 도메인에서 Windows 도메인 제어기에 IBM MQ 설치

404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』

Windows 예제 도메인에서 *Windows* 도메인 제어기에 *IBM MQ* 설치

wmq.example.com 도메인에 있는 두 도메인 컨트롤러에 IBM MQ을(를) 설치하고 구성하십시오.

여기에 첫 번째 단락과 요약에 사용되는 간단한 설명을 적으십시오.

이 구성 예는 다음 세 개의 서버로 구성됩니다.

sun
: 첫 번째 도메인 제어기로 사용되는 Windows Server 2008 서버입니다. 이는 *sun*, *earth* 및 *mars*를 포함하는 *wmq.example.com* 도메인을 정의합니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 한 인스턴스를 포함합니다.

earth
: 두 번째 도메인 제어기 IBM MQ 서버로 사용되는 Windows Server 2008입니다. 이 서버는 *QMGR*이라고 하는 다중 인스턴스 큐 관리자의 두 번째 인스턴스를 포함합니다.

mars
: 파일 서버로 사용되는 Windows Server 2008입니다.

예에서 이탤릭체로 표시된 이름을 선택한 이름으로 바꾸십시오.

시작하기 전에

1. 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』의 단계를 수행하여 도메인 *wmq.example.com*에 대해 도메인 제어기 *sun*을 작성하십시오. 이탤릭체로 표시된 이름을 구성에 맞게 변경하십시오.
2. 418 페이지의 『두 번째 Windows 도메인 제어기를 예제 도메인에 추가』의 단계를 수행하여 도메인 *wmq.example.com*에 대해 두 번째 도메인 제어기 *earth*를 작성하십시오. 이탤릭체로 표시된 이름을 구성에 맞게 변경하십시오.
3. IBM MQ 를 실행할 수 있는 기타 Windows 버전은 Windows 시스템의 하드웨어 및 소프트웨어 요구사항 을 참조하십시오.

이 태스크 정보

wmq.example.com 도메인에 있는 두 도메인 컨트롤러에 IBM MQ을(를) 설치하고 구성하십시오.

프로시저

1. *sun* 및 *earth*에서 IBM MQ를 설치하십시오.

   IBM MQ for Windows 설치 마법사 실행에 대한 자세한 정보는 Windows에서 IBM MQ 서버 설치를 참조하십시오.

   a) *sun* 및 *earth*에서 둘 다 도메인 관리자로 로그온하십시오. `wmq\Administrator`

   b) IBM MQ for Windows 설치 매체에서 **Setup** 명령을 실행하십시오.

      IBM MQ 런치패드 애플리케이션이 시작됩니다.

   c) **소프트웨어 요구사항**을 클릭하여 필수 소프트웨어가 설치되는지 확인하십시오.

   d) **네트워크 구성** > **아니오**를 클릭하십시오.

      이 설치에 대해 도메인 사용자 ID를 구성하거나 구성하지 않을 수 있습니다. 작성되는 사용자 ID는 도메인 로컬 사용자 ID입니다.

   e) **IBM MQ 설치**를 클릭하고 설치 언어를 선택한 다음 IBM MQ 설치 프로그램 시작을 클릭하십시오.

   f) 라이센스 계약을 확인하고 **다음** > **다음** > **설치**를 클릭하여 기본 구성을 승인하십시오. 설치가 완료될 때까지 대기한 후 **마침**을 클릭하십시오.

      설치 이름을 변경하거나, 다른 컴포넌트를 설치하거나, 큐 관리자 데이터 및 로그를 위한 다른 디렉토리를 구성하거나, 다른 디렉토리에 설치하려는 경우 **일반 설치**가 아닌 **사용자 설치**를 클릭하십시오.

      IBM MQ가 설치되었으며 설치 프로그램이 "Prepare IBM MQ " 마법사를 시작합니다.

   IBM MQ for Windows 설치는 도메인 로컬 그룹 `mqm` 및 도메인 그룹 `Domain mqm`을 구성합니다. 이 구성에서는 `Domain mqm`을 `mqm`의 멤버로 설정합니다. 같은 도메인에 있는 후속 도메인 제어기는 `mqm` 및 `Domain mqm` 그룹을 공유합니다.

2. *earth* 및 *sun* 모두에서 "IBM MQ 준비" 마법사를 실행하십시오.

   "IBM MQ 준비" 마법사 실행에 대한 추가 정보는 IBM MQ 준비 마법사로 IBM MQ 구성을 참조하십시오.

   a) IBM MQ 설치 프로그램은 "IBM MQ 준비"를 자동으로 실행합니다.

      마법사를 수동으로 시작하려면 **시작** > **모든 프로그램** > **IBM MQ** 폴더에서 " IBM MQ 준비 " 의 바로 가기를 찾으십시오. 다중 설치 구성에서 IBM MQ의 설치에 해당하는 바로 가기를 선택하십시오.

   b) **다음** 을 클릭하고 질문에 대한 응답으로 **아니오** 를 클릭하십시오. "네트워크에 Windows 2000 또는 이후 도메인 컨트롤러가 있는지 식별"[1].

   c) 마지막 마법사 페이지에서 필요에 따라 선택란을 선택하거나 선택 취소하고 **마침**을 클릭하십시오.

   "IBM MQ 준비" 마법사는 첫 번째 도메인 제어기에서는 도메인 로컬 사용자 `MUSR_MQADMIN`을 작성하고, 두 번째 도메인 제어기에서는 다른 도메인 로컬 사용자 `MUSR_MQADMIN1`을 작성합니다. 이 마법사는 `MUSR_MQADMIN` 또는 `MUSR_MQADMIN1`을 서비스에 로그온하는 사용자로 사용하여 각 제어기에서 IBM MQ 서비스를 작성합니다.

3. 큐 관리자 작성 권한이 있는 사용자를 정의하십시오.

   사용자는 로컬로 로그온할 수 있는 권한을 가져야 하고 도메인 로컬 `mqm`그룹의 멤버여야 합니다. 도메인 제어기에서 도메인 사용자는 로컬로 로그온할 수 있는 권한을 가지지 않지만 관리자에게는 해당 권한이 있습니다. 기본적으로 이러한 두 속성 모두 가지는 사용자는 없습니다. 이 태스크에서는 도메인 관리자를 도메인 로컬 `mqm` 그룹에 추가하십시오.

   a) **서버 관리자** > **역할** > **Active Directory 도메인 서비스** > ***wmq.example.com*** > **사용자**를 여십시오.

   b) **도메인 관리** > **그룹에 추가 ...** 를 마우스 오른쪽 단추로 클릭하십시오. 및 `mqm` 를 입력하십시오. **확인 이름** > **확인** > **확인** 을 누르십시오.

[1] 도메인에 대한 설치를 구성할 수 있습니다. 도메인 제어기의 모든 사용자 및 그룹에는 도메인 범위가 있기 때문에 이 옵션은 별다른 차이를 만들지 않습니다. 도메인에 없는 것처럼 IBM MQ을(를) 설치하는 것이 더 간단합니다.

결과

1. "IBM MQ 준비"가 도메인 사용자 MUSR_MQADMIN를 작성했는지 확인하십시오.
   a. **서버 관리자** > **역할** > **Active Directory 도메인 서비스** > ***wmq.example.com*** > **사용자**를 여십시오.
   b. 마우스 오른쪽 단추 누르기 **MUSR_MQADMIN** > **특성 ...** > **구성원**및 `Domain users` 및 mqm의 구성원임을 확인하십시오.
2. MUSR_MQADMIN에 서비스로 실행할 수 있는 권한이 있는지 확인하십시오.
   a. **시작** > **실행 ...**을 클릭하십시오. **secpol.msc** 명령을 입력하고 **확인**을 클릭하십시오.를 클릭하십시오.
   b. **보안 설정** > **로컬 정책** > **사용자 권한 지정**을 여십시오. 정책 목록에서 **서비스로서 로그온** > **특성**을 마우스 오른쪽 단추로 클릭하십시오.를 수행하고 MUSR_MQADMIN이 서비스로 로그온할 수 있는 권한을 가진 것으로 나열되는지 확인하십시오. **확인**을 클릭하십시오.

다음에 수행할 작업

1. 429 페이지의 『로컬 mqm 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기』 태스크를 실행하여 설치 및 구성이 올바로 작동하는지 확인하십시오.
2. 415 페이지의 『Windows 도메인 제어기에서 다중 인스턴스 큐 관리자 작성』 태스크로 돌아가서 도메인 제어기에서 다중 인스턴스 큐 관리자를 구성하는 태스크를 완료하십시오.

관련 태스크

Windows 두 번째 Windows 도메인 제어기를 예제 도메인에 추가

관련 정보

IBM MQ Windows 서비스에 필요한 사용자 권한

Windows *Windows*에서 다중 인스턴스 큐 관리자 확인

샘플 프로그램 **amqsghac**, **amqsphac** 및 **amqsmhac**를 사용하여 다중 인스턴스 큐 관리자 구성을 확인합니다. 이 주제에서는 Windows Server 2003에서 다중 인스턴스 큐 관리자 구성을 확인하는 구성 예를 제공합니다.

고가용성 샘플 프로그램은 자동 클라이언트 다시 연결을 사용합니다. 연결된 큐 관리자가 실패하면 클라이언트가 같은 큐 관리자 그룹에 있는 큐 관리자에 다시 연결하려고 시도합니다. 샘플 고가용성 샘플 프로그램에 대한 설명에서는 단순성을 위해 단일 인스턴스 큐 관리자를 사용한 클라이언트 다시 연결을 보여줍니다. 동일한 샘플을 다중 인스턴스 큐 관리자와 사용하여 다중 인스턴스 큐 관리자 구성을 확인할 수 있습니다.

이 예에서는 415 페이지의 『Windows 도메인 제어기에서 다중 인스턴스 큐 관리자 작성』에 설명된 다중 인스턴스 구성을 사용합니다. 이 구성을 사용하여 다중 인스턴스 큐 관리자가 대기 인스턴스로 전환하는지 확인하십시오. **endmqm** 명령을 사용하여 큐 관리자를 중지하고 전환 옵션인 `-s`를 사용하십시오. 클라이언트 프로그램이 새 큐 관리자 인스턴스에 다시 연결하여 잠깐의 지연 후 새 인스턴스에 대한 작업을 계속합니다.

클라이언트는 Windows 7 서비스 팩 1을 실행 중인 400MB VMware 이미지에 설치됩니다. 보안상의 이유로 멀티 인스턴스 큐 관리자를 실행 중인 도메인 서버와 동일한 VMware 호스트 전용 네트워크에 연결됩니다. 이 클라이언트는 구성을 단순화하기 위해 클라이언트 연결 테이블을 포함하는 /MQHA 폴더를 공유합니다.

IBM MQ Explorer를 사용한 장애 복구 확인

장애 복구를 확인하기 위해 샘플 애플리케이션을 사용하기 전에 각 서버에서 IBM MQ Explorer를 실행하십시오. **리모트 큐 관리자 추가 > 멀티 인스턴스 큐 관리자에 직접 연결** 마법사를 사용하여 각 탐색기에 큐 관리자 인스턴스 모두를 추가하십시오. 대기를 허용하여 두 인스턴스 모두 실행되고 있는지 확인하십시오. 활성 인스턴스로 VMware 이미지를 실행하는 창을 닫아서 가상으로 서버의 전원을 끄거나 활성 인스턴스를 중지하여 대기 인스턴스로의 전환을 허용하고 다시 연결 가능한 클라이언트가 다시 연결할 수 있게 하십시오.

⚠ **주의:** 서버의 전원을 끌 경우에는 해당 서버가 MQHA 폴더를 호스팅하는 서버가 아니어야 합니다!

참고: **대기 인스턴스로 전환 허용** 옵션은 **큐 관리자 중지** 대화 상자에서 사용할 수 없습니다. 이 옵션은 큐 관리자가 단일 인스턴스 큐 관리자로 실행되고 있기 때문에 누락되었습니다. **대기 인스턴스 허용** 옵션 없이 큐 관리자를 시작했습니다. 큐 관리자 중지 요청이 거부된 경우 **자세한 내용** 창을 확인하십시오. 실행 중인 대기 인스턴스가 없을 수 있습니다.

샘플 프로그램을 사용하여 장애 복구 확인

활성 인스턴스를 실행할 서버 선택

서버 중 MQHA 디렉토리 또는 파일 시스템을 호스팅할 서버를 선택했을 수 있습니다. 활성 서버를 실행 중인 VMware 창을 닫아 장애 복구를 테스트할 계획인 경우 이 서버는 MQHA를 호스팅하는 서버가 아니어야 합니다!

활성 큐 관리자 인스턴스를 실행하는 서버에서

1. *ipaddr1* 및 *ipaddr2*를 수정하고 다음 명령을 `N:\hasample.tst.`에 저장하십시오.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN)  TRPTYPE(TCP) +
MCAUSER(' ') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(' ipaddr1 (1414), ipaddr2 (1414)') QMNAME(QM1) REPLACE
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

참고: **MCAUSER** 매개변수를 공백으로 둠으로써 클라이언트 사용자 ID가 서버로 송신됩니다. 이 클라이언트 사용자 ID에는 서버에 대한 올바른 권한이 있어야 합니다. 대안은 서버에서 구성한 사용자 ID에 대한 SVRCONN 채널에서 **MCAUSER** 매개변수를 설정하는 것입니다.

2. `N:\` 경로에서 명령 프롬프트를 열고 다음 명령을 실행하십시오.

```
runmqsc -m QM1 < hasample.tst
```

3. **runmqsc** 명령의 출력을 검사하여 리스너가 실행 중이고 큐 관리자 제어를 가지는지 확인하십시오.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

또는 TCPIP 리스너가 실행 중이고 `Control = Queue Manager`를 가지는 IBM MQ Explorer를 사용하여 확인하십시오.

클라이언트에서

1. 서버의 공유 디렉토리 `C:\MQHA`를 클라이언트의 `N:\`로 맵핑하십시오.
2. `N:\` 경로로 명령 프롬프트를 여십시오. 서버의 클라이언트 채널 정의 테이블(CCDT)을 가리키도록 환경 변수 `MQCHLLIB`를 다음과 같이 설정하십시오.

```
SET MQCHLLIB=N:\data\QM1\@ipcc
```

3. 명령 프롬프트에 다음 명령을 입력하십시오.

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

참고: 문제점이 있는 경우 이유 코드가 콘솔에 인쇄되도록 명령 프롬프트에서 애플리케이션을 시작하거나 `N:\data\QM1\errors` 폴더에서 AMQERR01.LOG 파일을 보십시오.

활성 큐 관리자 인스턴스를 실행하는 서버에서

1. 다음 중 하나입니다.
   - 활성 서버 인스턴스로 VMware 이미지를 실행하는 창을 닫으십시오.
   - IBM MQ Explorer를 사용하여 활성 큐 관리자 인스턴스를 중지하여 대기 인스턴스로의 전환을 허용하고 다시 연결 가능 클라이언트에 다시 연결하도록 지시하십시오.

2. 세 클라이언트는 결국 연결이 끊어진 것을 감지한 후 다시 연결합니다. 이 구성에서 서버 창을 닫으면 세 연결 모두 재설정되는 데 약 7분 정도 소요됩니다. 일부 연결은 다른 연결 전에 정상적으로 재설정됩니다.

결과

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsghac TARGET QM1
Sample AMQSGHAC start
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

Windows *Windows*에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안

이 주제에서는 글로벌 대체 보안 그룹을 사용하여 큐 관리자 데이터 및 로그 파일의 공유 위치를 보안 설정하는 방법을 설명합니다. 서로 다른 서버에서 실행되는 여러 큐 관리자 인스턴스 사이에서 위치를 공유할 수 있습니다.

일반적으로 큐 관리자 데이터 및 로그 파일에 대한 공유 위치는 설정하지 않습니다. IBM MQ for Windows를 설치하면 이 설치 프로그램은 해당 서버에서 작성되는 큐 관리자에 대해 사용자가 선택하는 홈 디렉토리를 작성합니다. 이는 로컬 mqm 그룹을 사용하여 디렉토리를 보안 설정하고 IBM MQ 서비스에 대한 사용자 ID가 해당 디렉토리에 액세스할 수 있도록 구성합니다.

보안 그룹으로 공유 폴더를 보안 설정하는 경우 폴더에 대한 액세스가 허용된 사용자에게는 그룹의 신임 정보가 있어야 합니다. 원격 파일 서버의 폴더가 `mars` 서버의 로컬 mqm 그룹으로 보안된다고 가정합니다. 큐 관리자 프로세스를 실행하는 사용자를 `mars`의 로컬 mqm 그룹 멤버로 설정하십시오. 사용자에게는 원격 파일 서버에 있는 폴더의 신임 정보와 일치하는 신임 정보가 있습니다. 이러한 신임 정보를 사용하여 큐 관리자는 폴더에 있는 해당 데이터 및 로그 파일에 액세스할 수 있습니다. 다른 서버에서 큐 관리자 프로세스를 실행하는 사용자는 일치하는 신임 정보가 없는 다른 로컬 mqm 그룹의 멤버입니다. 큐 관리자가 `mars`와는 다른 서버에서 실행되는 경우 `mars`에서 실행될 때 작성되는 데이터 및 로그 파일에 액세스할 수 없습니다. 사용자를 도메인 사용자로 설정하더라도 이 사용자는 다른 신임 정보를 가집니다. 이 사용자가 `mars`의 로컬 mqm 그룹에서 신임 정보를 획득하고 다른 서버에서는 신임 정보 획득을 수행할 수 없기 때문입니다.

큐 관리자에 글로벌 대체 보안 그룹을 제공하면 문제점이 해결됩니다. 425 페이지의 그림 75의 내용을 참조하십시오. 글로벌 그룹으로 원격 폴더를 보안 설정하십시오. *mars*에서 작성 시 큐 관리자에 글로벌 그룹 이름을 전달합니다. **crtmqm** 명령에 -a[r] 매개변수를 사용하여 글로벌 그룹 이름을 대체 보안 그룹으로 전달하십시오. 다른 서버에서 실행하기 위해 큐 관리자를 전송하는 경우 보안 그룹의 이름이 함께 전송됩니다. 이 이름은 SecurityGroup으로서 qm.ini 파일의 **AccessMode** 스탠자로 전송됩니다. 예를 들면, 다음과 같습니다.

```
AccessMode:
SecurityGroup=wmq\wmq
```

qm.ini의 **AccessMode** 스탠자는 RemoveMQMAccess도 포함합니다. 예를 들면, 다음과 같습니다.

```
AccessMode:
RemoveMQMAccess=true|false
```

이 속성이 true 값으로 지정되고 액세스 그룹도 제공된 경우 로컬 mqm 그룹에는 큐 관리자 데이터 파일에 대한 액세스 권한이 부여되지 않습니다.

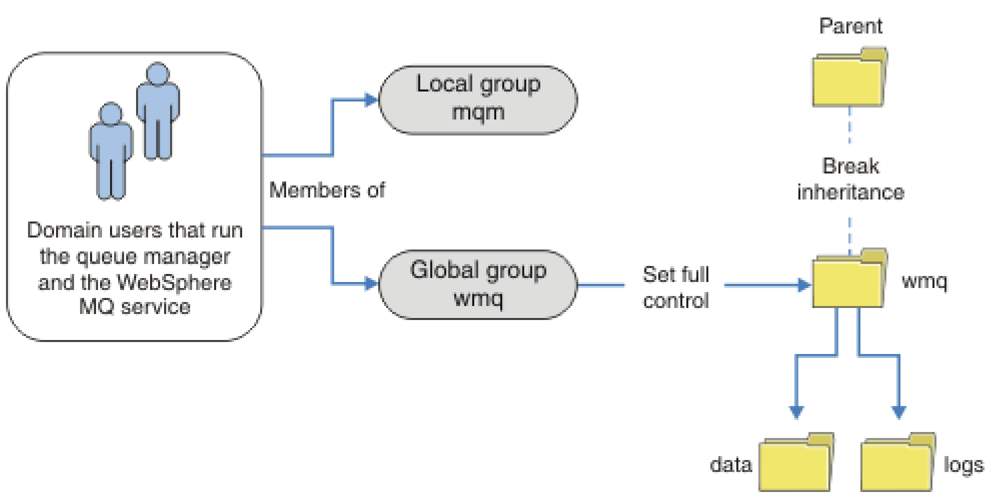


그림 *75.* 대체 글로벌 보안 그룹을 사용하여 큐 관리자 데이터 및 로그 보안 설정*(1)*

큐 관리자 프로세스 실행에 사용하는 사용자 ID가 글로벌 보안 그룹의 신임 정보와 일치하는 신임 정보를 가지도록 하려면 이 사용자 ID에도 글로벌 범위가 있어야 합니다. 로컬 그룹 또는 프린시펄을 글로벌 그룹의 멤버로 설정할 수 없습니다. 425 페이지의 그림 75에 큐 관리자 프로세스를 실행하는 사용자가 도메인 사용자로 표시되어 있습니다.

다수의 IBM MQ 서버를 배치하는 경우 425 페이지의 그림 75에서 사용자의 그룹화는 편리하지 않습니다. 모든 IBM MQ 서버에 대해 사용자를 로컬 그룹에 추가하는 프로세스를 반복해야 합니다. 대신 도메인 컨트롤러에서 Domain mqm 글로벌 그룹을 작성하고 Domain mqm 그룹의 IBM MQ 멤버를 실행하는 사용자를 작성하십시오. 426 페이지의 그림 76의 내용을 참조하십시오. 도메인 설치로서 IBM MQ를 설치하는 경우 "IBM MQ 준비" 마법사는 Domain mqm 그룹을 로컬 mqm 그룹의 멤버로 만듭니다. 동일한 사용자가 전역 그룹 Domain mqm 및 wmq에 있습니다.

팁: 동일한 사용자가 여러 서버에서 IBM MQ를 실행할 수 있지만 개별 서버에서는 IBM MQ를 서비스로 실행하고 대화식으로 실행하기 위해 다른 사용자가 있어야 합니다. 서버의 모든 설치에 대해서도 다른 사용자가 있어야 합니다. 일반적으로 Domain mqm에는 여러 사용자가 포함되어 있습니다.

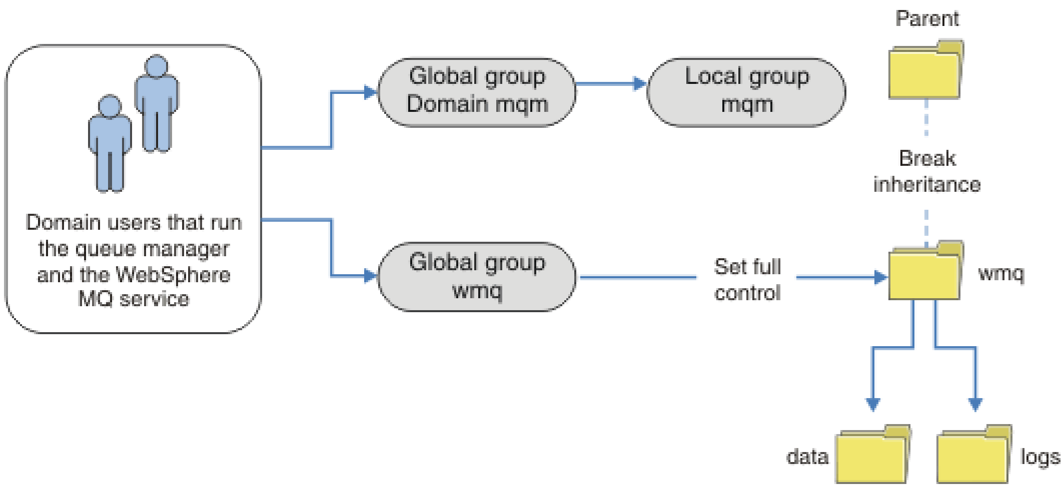


그림 76. 대체 글로벌 보안 그룹을 사용하여 큐 관리자 데이터 및 로그 보안 설정(2)

426 페이지의 그림 76에서 조직은 현재 상태 그대로 불필요하게 복잡합니다. 이 배열에는 동일한 멤버를 포함하는 두 개의 글로벌 그룹이 있습니다. 조직을 단순화하고 하나의 글로벌 그룹만 정의할 수 있습니다(426 페이지의 그림 77 참조).

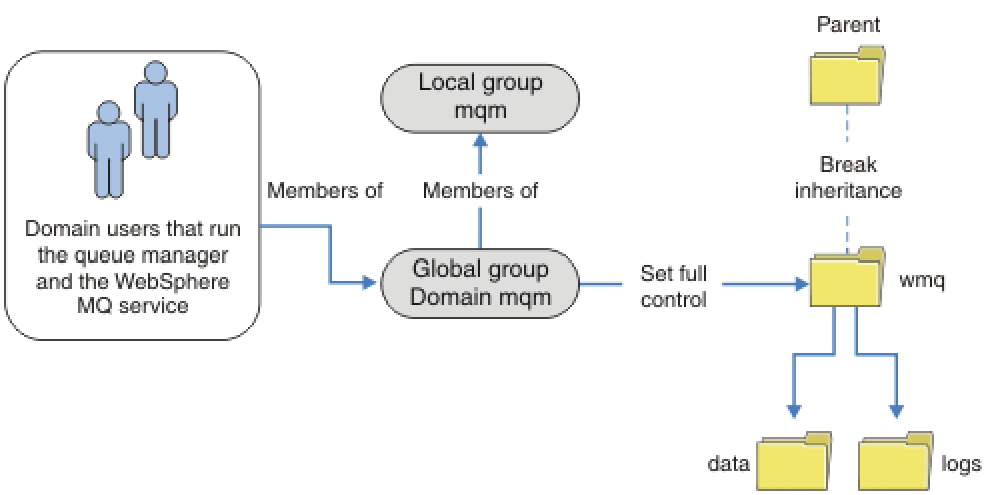


그림 77. 대체 글로벌 보안 그룹을 사용하여 큐 관리자 데이터 및 로그 보안 설정(3)

또는 폴더에 따라 다른 큐 관리자가 액세스할 수 있도록 제한되는 더 세분화된 수준의 액세스 제어가 필요할 수 있습니다. 427 페이지의 그림 78의 내용을 참조하십시오. 427 페이지의 그림 78에서 다른 큐 관리자 로그 및 데이터 파일을 보안 설정하기 위해 두 그룹의 도메인 사용자가개별 글로벌 그룹으로 정의됩니다. 서로 다른 IBM MQ 서버에 있어야 하는 두 개의 다른 로컬 mqm 그룹이 표시됩니다. 이 예에서 큐 관리자는 두 세트에 다른 사용자가 할당되는 두 개의 세트로 파티션됩니다. 이 두 세트는 테스트 및 프로덕션 큐 관리자일 수 있습니다. 대체 보안 그룹을 wmq1 및 wmq2라고 합니다. 글로벌 그룹 wmq1 및 wmq2를 테스트 구획에 있는지 또는 프로덕션 구획에 있는지에 따라 해당하는 큐 관리자에 수동으로 추가해야 합니다. 구성은 두 그룹의 사용자가 있기 때문에 IBM MQ 설치가 426 페이지의 그림 77에서와 같이 로컬 mqm 그룹에 `Domain mqm` 전파되는 것을 이용할 수 없습니다.

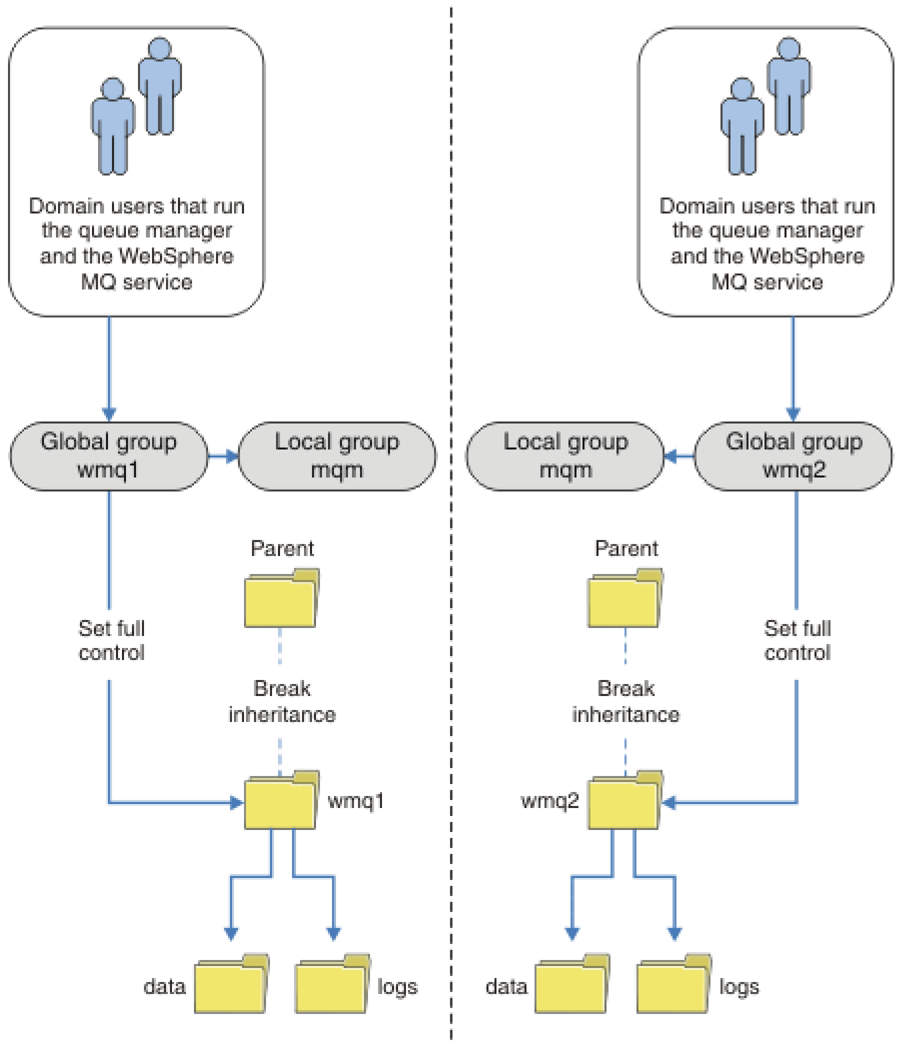


그림 *78.* 대체 글로벌 보안 그룹을 사용하여 큐 관리자 데이터 및 로그 보안 설정*(4)*

두 구획으로 파티션하는 것에 대한 대체 방식은 이를 두 개의 창 도메인에 배치하는 것입니다. 이 경우에 426 페이지의 그림 77에 표시된 보다 단순한 모델 사용으로 돌아갈 수 있습니다.

Windows *Windows*에서 비공유 큐 관리자 데이터와 로그 디렉토리 및 파일 보안 설정

이 주제에서는 로컬 mqm 그룹 및 대체 보안 그룹 둘 다를 사용하여 큐 관리자 데이터 및 로그 파일의 대체 위치를 보안 설정하는 방법을 설명합니다.

일반적으로 큐 관리자 데이터 및 로그 파일에 대한 대체 위치는 설정하지 않습니다. IBM MQ for Windows를 설치하면 이 설치 프로그램은 작성되는 큐 관리자에 대해 사용자가 선택하는 홈 디렉토리를 작성합니다. 이는 로컬 mqm 그룹을 사용하여 디렉토리를 보안 설정하고 IBM MQ 서비스에 대한 사용자 ID가 해당 디렉토리에 액세스할 수 있도록 구성합니다.

두 가지 예에서 IBM MQ에 대한 액세스 제어를 구성하는 방법을 보여줍니다. 이러한 예는 설치 시 작성된 데이터 및 로그 경로에 없는 디렉토리의 해당 데이터 및 로그를 사용하여 큐 관리자를 작성하는 방법을 보여줍니다. 첫 번째 예 429 페이지의 『로컬 mqm 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기』에서는 로컬 mqm 그룹에 의해 권한을 부여하여 큐 및 로그 디렉토리에 대한 액세스를 허용합니다. 두 번째 예 432 페이지의 『대체 로컬 보안 그룹이 권한을 부여한 데이터 및 로그 파일 읽기 및 쓰기』는 디렉토리에 대한 액세스 권한이 대체 보안 그룹에 의해 부여된다는 점에서 다릅니다. 한 서버에서만 실행되는 큐 관리자가 디렉토리에 액세스하는 경우 대체 보안 그룹으로 데이터 및 로그 파일을 보안 설정하면 다른 로컬 그룹 또는 프린시펄을 사용하여 다른 큐 관리자를 보안 설정하는 방법을 선택할 수 있습니다. 다중 인스턴스 큐 관리자와 같이 여러 서버에서 실행되는 큐 관리자가 디렉토리에 액세스하는 경우 대체 보안 그룹으로 데이터 및 로그 파일을 보안 설정하는 방법만

선택할 수 있습니다. 424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』의 내용을 참조하십시오.

큐 관리자 데이터 및 로그 파일의 보안 권한 구성은 Windows에서 공통 구성이 아닙니다. IBM MQ for Windows를 설치할 때 큐 관리자 데이터 및 로그에 대한 디렉토리를 지정하거나 기본 디렉토리를 승인합니다. 설치 프로그램은 로컬 mqm 그룹에 전체 제어 권한을 제공하여 이 그룹으로 이러한 디렉토리를 자동으로 보안 설정합니다. 설치 프로세스는 큐 관리자를 실행하는 사용자 ID가 로컬 mqm그룹의 멤버인지 확인합니다. 액세스 요구사항을 충족시키기 위해 디렉토리에 대한 다른 액세스 제어를 수정할 수 있습니다.

데이터 및 로그 파일 디렉토리를 새 위치로 이동하는 경우 새 위치의 보안을 구성해야 합니다. 큐 관리자를 백업하고 다른 컴퓨터로 복원하는 경우 또는 큐 관리자가 다중 인스턴스 큐 관리자가 되도록 변경하는 경우 디렉토리의 위치를 변경할 수 있습니다. 새 위치에서 큐 관리자 데이터 및 로그 디렉토리를 보안 설정하기 위해 선택할 수 있는 두 가지 방법이 있습니다. 로컬 mqm 그룹에 대한 액세스를 제한하여 디렉토리를 보안 설정하거나 선택한 보안 그룹에 대한 액세스를 제한할 수 있습니다.

로컬 mqm 그룹을 사용하여 디렉토리를 보안 설정하는 방법에 가장 적은 수의 단계가 소요됩니다. 데이터 및 로그 디렉토리에 대한 권한을 설정하여 로컬 mqm 그룹 전체 제어를 허용하십시오. 일반적인 접근법은 기존 권한 세트를 복사하고 상위에서 상속을 제거하는 것입니다. 그런 다음 다른 프린시펄의 권한을 제거하거나 제한할 수 있습니다.

IBM MQ 준비 마법사에서 설정된 서비스에 대해 다른 사용자 ID로 큐 관리자를 실행하는 경우 해당 사용자 ID는 로컬 mqm 그룹의 멤버여야 합니다. 이 429 페이지의 『로컬 mqm 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기』 태스크는 단계를 통해 안내합니다.

대체 보안 그룹을 사용하여 큐 관리자 데이터 및 로그 파일을 보안 설정할 수도 있습니다. 대체 보안 그룹으로 큐 관리자 데이터 및 로그 파일을 보안 설정하는 프로세스에는 428 페이지의 그림 79의 내용을 참조하는 다수의 단계가 포함됩니다. 로컬 그룹 wmq는 대체 보안 그룹의 예입니다.

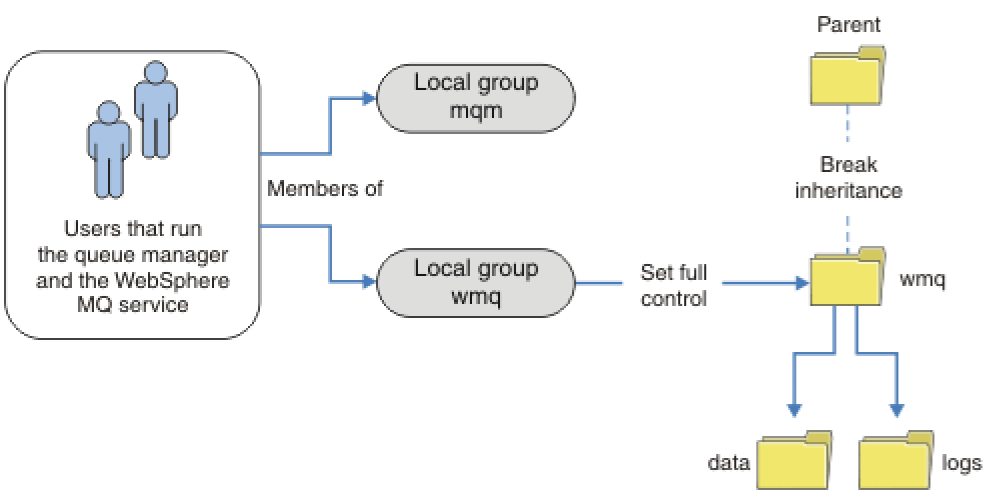


그림 79. 대체 로컬 보안 그룹, *wmq*를 사용하여 큐 관리자 데이터 및 로그 보안

1. 큐 관리자 데이터 및 로그에 대한 개별 디렉토리, 공용 디렉토리 또는 공용 상위 디렉토리를 작성하십시오.
2. 디렉토리에 대한 기존 상속 권한 세트 또는 상위 디렉토리를 복사하고 필요에 따라 수정하십시오.
3. 대체 그룹 wmq에 디렉토리에 대한 전체 제어 권한을 제공하여 큐 관리자 및 로그를 포함할 디렉토리를 보안 설정하십시오.
4. 큐 관리자 프로세스를 실행하는 모든 사용자 ID에 대체 보안 그룹의 신임 정보 또는 프린시펄을 제공하십시오.
   a. 사용자를 대체 보안 프린시펄로 정의하는 경우 이 사용자는 큐 관리자를 실행할 관리자와 같은 관리자여야 합니다. 이 사용자는 로컬 mqm 그룹의 멤버여야 합니다.

b. 로컬 그룹을 대체 보안 그룹으로 정의하는 경우 큐 관리자를 실행할 사용자를 대체 그룹에 추가하십시오. 이 사용자는 또한 로컬 mqm 그룹의 멤버여야 합니다.

c. 글로벌 그룹을 대체 보안 그룹으로 정의하는 경우 424 페이지의 『Windows에서 공유 큐 관리자 데이터 및 로그 디렉토리와 파일 보안』의 내용을 참조하십시오.

5. -a 매개변수로 **crtmqm** 명령에서 대체 보안 그룹 또는 프린시펄을 지정하여 큐 관리자를 작성하십시오.

Windows 로컬 *mqm* 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기

이 태스크에서는 선택한 디렉토리에 저장된 해당 데이터 및 로그 파일을 사용하여 큐 관리자를 작성하는 방법을 보여줍니다. 파일에 대한 액세스는 로컬 mqm 그룹에 의해 보안 설정됩니다. 이 디렉토리는 공유하지 않습니다.

시작하기 전에

1. 기본 설치로 IBM MQ for Windows를 설치하십시오.
2. "IBM MQ 준비 " 마법사를 실행하십시오. 이 태스크에서는 로컬 사용자 ID로 실행하거나 또는 도메인 사용자 ID로 실행하도록 설치를 구성하십시오. 결국 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』의 모든 태스크를 완료하기 위해 설치는 도메인에 대해 구성되어야 합니다.
3. 관리자 권한으로 로그온하여 태스크의 첫 번째 부분을 수행하십시오.

이 태스크 정보

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

Windows에서는 사용자가 선택한 디렉토리의 IBM MQ for Windows에 대한 기본 데이터 및 로그 경로를 작성할 수 있습니다. 이 설치 및 구성 마법사는 로컬 mqm 그룹 및 큐 관리자 프로세스를 실행하는 사용자 ID에 디렉토리에 대한 액세스 권한을 자동으로 제공합니다. 큐 관리자 데이터 및 로그 파일에 대해 다른 디렉토리를 지정하는 큐 관리자를 작성하는 경우 이 디렉토리에 대한 전체 제어 권한을 구성해야 합니다.

이 예에서는 로컬 mqm 그룹 권한을 디렉토리 *c:\wmq*에 제공함으로써 큐 관리자에게 해당 데이터 및 로그 파일에 대한 전체 제어를 제공합니다.

crtmqm 명령은 IBM MQ 서비스를 사용하여 워크스테이션이 시작될 때 자동으로 시작되는 큐 관리자를 작성합니다.

이 태스크는 설명용이며 변경할 수 있는 특정 값을 사용합니다. 변경할 수 있는 값은 이탤릭체로 표시됩니다. 태스크 종료 시에 사용자가 작성한 모든 변경사항을 제거하려면 지시사항을 따르십시오.

프로시저

1. 명령 프롬프트를 여십시오.
2. 명령을 입력하십시오.

```
md c:\wmq\data, c:\wmq\logs
```

3. 디렉토리에 대해 로컬 mqm 그룹 읽기 및 쓰기 액세스를 허용하는 권한을 설정하십시오.

```
cacls c:\wmq/T /E /G mqm:F
```

시스템이 다음과 같이 응답합니다.

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

4. 옵션: 로컬 mqm 그룹의 멤버인 사용자 ID로 전환하십시오.

관리자로서 계속할 수 있지만 실제 프로덕션 구성에서는 보다 제한적인 권한을 가지는 사용자 ID로 계속할 수 있습니다. 이 사용자 ID는 최소한 로컬 mqm 그룹의 멤버여야 합니다.

IBM MQ 설치가 도메인의 일부로 구성되면 이 사용자 ID를 `Domain mqm` 그룹의 멤버로 설정하십시오. "IBM MQ 준비" 마법사에서는 `Domain mqm` 글로벌 그룹이 로컬 mqm 그룹의 멤버가 되도록 하므로, 사용자 ID를 로컬 mqm 그룹의 멤버로 만들 필요가 없습니다.

5. 큐 관리자를 작성하십시오.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager created.
Directory 'c:\wmq\data\QMGR' created.
The queue manager is associated with installation '1'
Creating or replacing default objects for queue manager 'QMGR'
Default objects statistics : 74 created. 0 replaced.
Completing setup.
Setup completed.
```

6. 큐 관리자에서 작성한 디렉토리가 *c:\wmq* 디렉토리에 있는지 확인하십시오.

```
dir c:\wmq/D /B /S
```

7. 파일에 읽기 및 쓰기 또는 로컬 mqm 그룹에 대한 전체 제어 권한이 있는지 확인하십시오.

```
cacls c:\wmq\*.*
```

다음에 수행할 작업

메시지를 큐에 넣거나 가져와서 큐 관리자를 테스트하십시오.

1. 큐 관리자를 시작하십시오.

```
strmqm QMGR
```

시스템이 다음과 같이 응답합니다.

```
IBM MQ queue manager 'QMGR' starting.
The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log
replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using 7.1.0.0.
```

2. 테스트 큐를 작성하십시오.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

시스템이 다음과 같이 응답합니다.

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)
AMQ8006: IBM MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. 샘플 프로그램 **amqsput**을 사용하여 테스트 메시지를 넣으십시오.

   ```
   echo 'A test message' | amqsput QTEST QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   Sample AMQSPUT0 start
   target queue is QTEST
   Sample AMQSPUT0 end
   ```

4. 샘플 프로그램 **amqsget**을 사용하여 테스트 메시지를 가져오십시오.

   ```
   amqsget QTEST QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   Sample AMQSGET0 start
   message A test message
   Wait 15 seconds ...
   no more messages
   Sample AMQSGET0 end
   ```

5. 큐 관리자를 중지합니다.

   ```
   endmqm -i QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ended.
   ```

6. 큐 관리자를 삭제합니다.

   ```
   dltmqm QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' deleted.
   ```

7. 작성한 디렉토리를 삭제하십시오.

   팁: 명령이 각 파일 또는 디렉토리를 삭제하도록 프롬프트 표시되는 것을 막으려면 /Q 옵션을 명령에 추가하십시오.

   ```
   del /F /S C:\wmq\*.*
   rmdir /S C:\wmq
   ```

관련 개념

399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』

Windows의 다중 인스턴스 큐 관리자에서는 해당 데이터 및 로그가 공유되어야 합니다. 공유는 여러 서버 또는 워크스테이션에서 실행 중인 큐 관리자의 모든 인스턴스가 액세스할 수 있어야 합니다. 큐 관리자 및 공유를 Windows 도메인의 부분으로 구성하십시오. 큐 관리자는 도메인 워크스테이션 또는 서버 또는 도메인 제어기에서 실행할 수 있습니다.

관련 태스크

Windows 대체 로컬 보안 그룹이 권한을 부여한 데이터 및 로그 파일 읽기 및 쓰기
이 태스크에서는 **crtmqm** 명령에서 -a 플래그를 사용하는 방법을 보여줍니다. 이 플래그는 큐 관리자에게 대체 로컬 보안 그룹을 제공하여 해당 로그 및 데이터 파일에 대한 액세스 권한을 제공합니다.

412 페이지의 『대체 글로벌 보안 그룹이 권한을 부여한 공유 데이터 및 로그 파일 읽기 및 쓰기』
401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』

Windows 대체 로컬 보안 그룹이 권한을 부여한 데이터 및 로그 파일 읽기 및 쓰기
이 태스크에서는 **crtmqm** 명령에서 -a 플래그를 사용하는 방법을 보여줍니다. 이 플래그는 큐 관리자에게 대체 로컬 보안 그룹을 제공하여 해당 로그 및 데이터 파일에 대한 액세스 권한을 제공합니다.

시작하기 전에

1. 기본 설치로 IBM MQ for Windows를 설치하십시오.
2. "IBM MQ 준비 " 마법사를 실행하십시오. 이 태스크에서는 로컬 사용자 ID로 실행하거나 또는 도메인 사용자 ID로 실행하도록 설치를 구성하십시오. 결국 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』의 모든 태스크를 완료하기 위해 설치는 도메인에 대해 구성되어야 합니다.
3. 관리자 권한으로 로그온하여 태스크의 첫 번째 부분을 수행하십시오.

이 태스크 정보

이 태스크는 큐 관리자 데이터 및 로그 파일 액세스를 설명하는 관련 태스크 세트 중 하나입니다. 이 태스크는 사용자가 선택한 디렉토리에 저장된 데이터 및 로그 파일을 읽고 쓸 수 있도록 권한이 부여된 큐 관리자를 작성하는 방법을 보여줍니다. 399 페이지의 『Windows 도메인 및 다중 인스턴스 큐 관리자』 태스크를 수반합니다.

Windows에서는 사용자가 선택한 디렉토리의 IBM MQ for Windows에 대한 기본 데이터 및 로그 경로를 작성할 수 있습니다. 이 설치 및 구성 마법사는 로컬 mqm 그룹 및 큐 관리자 프로세스를 실행하는 사용자 ID에 디렉토리에 대한 액세스 권한을 자동으로 제공합니다. 큐 관리자 데이터 및 로그 파일에 대해 다른 디렉토리를 지정하는 큐 관리자를 작성하는 경우 이 디렉토리에 대한 전체 제어 권한을 구성해야 합니다.

이 예에서는 큐 관리자에게 디렉토리에 대한 전체 제어 권한이 있는 대체 보안 로컬 그룹을 제공합니다. 이 대체 보안 그룹은 큐 관리자에게 디렉토리에 있는 파일을 관리할 수 있는 권한을 제공합니다. 대체 보안 그룹의 기본 용도는 대체 보안 글로벌 그룹에 권한을 부여하는 것입니다. 대체 보안 글로벌 그룹을 사용하여 다중 인스턴스 큐 관리자를 설정하십시오. 이 예에서는 사용자가 도메인에 IBM MQ를 설치하지 않고도 대체 보안 그룹의 사용을 익힐 수 있도록 로컬 그룹을 구성합니다. 로컬 그룹을 대체 보안 그룹으로 구성하는 것은 일반적이지 않습니다.

crtmqm 명령은 IBM MQ 서비스를 사용하여 워크스테이션이 시작될 때 자동으로 시작되는 큐 관리자를 작성합니다.

이 태스크는 설명용이며 변경할 수 있는 특정 값을 사용합니다. 변경할 수 있는 값은 이탤릭체로 표시됩니다. 태스크 종료 시에 사용자가 작성한 모든 변경사항을 제거하려면 지시사항을 따르십시오.

프로시저

1. 대체 보안 그룹을 설정하십시오.

   대체 보안 그룹은 일반적으로 도메인 그룹입니다. 이 예에서는 로컬 대체 보안 그룹을 사용하는 큐 관리자를 작성합니다. 로컬 대체 보안 그룹을 사용하여 도메인의 일부가 아닌 IBM MQ 설치에 대한 태스크를 수행할 수 있습니다.

   a) **lusrmgr.msc** 명령을 실행하여 로컬 사용자 및 그룹 창을 여십시오.

   b) 다음에서 마우스의 오른쪽 단추를 클릭하십시오. **그룹 > 새 그룹...**

   c) **그룹 이름** 필드에 *altmqm* 를 입력하고 **작성 > 닫기**를 클릭하십시오.

d) IBM MQ 서비스를 실행하는 사용자 ID를 식별하십시오.

i) **시작** > **실행 ...**을 클릭하십시오. `services.msc` 을 입력하고 **확인**을 클릭하십시오.

ii) 서비스 목록에서 IBM MQ 서비스를 클릭하고 로그온 탭을 클릭하십시오.

iii) 이 사용자 ID를 기억해두고 Services Explorer를 닫으십시오.

e) IBM MQ 서비스를 실행하는 사용자 ID를 *altmqm* 그룹에 추가하십시오. 또한 로그온할 때 사용하는 사용자 ID를 추가하여 큐 관리자를 작성하고 대화식으로 실행하십시오.

Windows는 큐 관리자 프로세스를 실행하는 사용자 ID의 권한을 확인하여 데이터 및 로그 디렉토리에 액세스하기 위한 큐 관리자의 권한을 확인합니다. 이 사용자 ID는 직접적으로 또는 글로벌 그룹을 통해 간접적으로 디렉토리에 권한을 부여한 *altmqm* 그룹의 멤버여야 합니다.

IBM MQ을(를) 도메인의 일부로 설치하고 401 페이지의 『Windows의 도메인 워크스테이션 또는 서버에서 다중 인스턴스 큐 관리자 작성』에서 작업을 하려는 경우 404 페이지의 『Windows에서 Active Directory 및 DNS 도메인 작성』에서 작성되는 도메인 사용자 ID는 *wmquser1* 및 *wmquser2* 입니다.

큐 관리자를 도메인의 일부로 설치하지 않은 경우 IBM MQ 서비스를 실행하는 기본 로컬 사용자 ID는 `MUSR_MQADMIN`입니다. 관리자 권한 없이 이 태스크를 수행하려고 하는 경우 로컬 `mqm` 그룹의 멤버인 사용자를 작성하십시오.

다음 단계에 따라 *wmquser1* 및 *wmquser2*를 *altmqm*에 추가하십시오. 구성이 다른 경우 사용자 ID 및 그룹을 자신의 이름으로 대체하십시오.

i) 그룹 목록의 **altmqm** > **특성** > **추가...**에서 마우스의 오른쪽 단추를 클릭하십시오.

ii) 사용자, 컴퓨터 또는 그룹 선택 창에서 *wmquser1* ; *wmquser2* 을 입력하고 **이름 확인**을 클릭하십시오.

iii) Windows 보안 창에서 도메인 관리자의 이름 및 비밀번호를 입력한 다음 **확인** > **확인** > **적용** > **확인**을 클릭하십시오.

2. 명령 프롬프트를 여십시오.
3. IBM MQ 서비스를 재시작하십시오.

   서비스를 실행하는 사용자 ID가 그에 대해 구성한 추가 보안 신임 정보를 획득할 수 있도록 서비스를 재시작해야 합니다.

   다음 명령을 입력하십시오.

```
endmqsvc
strmqsvc
```

   시스템이 다음과 같이 응답합니다.

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' ended successfully.
```

   및

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
The MQ service for installation 'Installation1' started successfully.
```

4. 명령을 입력하십시오.

```
md c:\wmq\data, c:\wmq\logs
```

5. 로컬 사용자 *user* 읽기 및 쓰기 액세스를 허용하도록 디렉토리에 대한 권한을 설정하십시오.

```
cacls c:\wmq/T /E /G altmqm:F
```

   시스템이 다음과 같이 응답합니다.

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

6. 옵션: 로컬 mqm 그룹의 멤버인 사용자 ID로 전환하십시오.

   관리자로서 계속할 수 있지만 실제 프로덕션 구성에서는 보다 제한적인 권한을 가지는 사용자 ID로 계속할 수 있습니다. 이 사용자 ID는 최소한 로컬 mqm 그룹의 멤버여야 합니다.

   IBM MQ 설치가 도메인의 일부로 구성되면 이 사용자 ID를 `Domain mqm` 그룹의 멤버로 설정하십시오. "IBM MQ 준비" 마법사에서는 `Domain mqm` 글로벌 그룹이 로컬 mqm 그룹의 멤버가 되도록 하므로, 사용자 ID를 로컬 mqm 그룹의 멤버로 만들 필요가 없습니다.

7. 큐 관리자를 작성하십시오.

   ```
   crtmqm -a altmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager created.
   Directory 'c:\wmq1\data\QMGR' created.
   The queue manager is associated with installation '1'
   Creating or replacing default objects for queue manager 'QMGR'
   Default objects statistics : 74 created. 0 replaced.
   Completing setup.
   Setup completed.
   ```

8. 큐 관리자에서 작성한 디렉토리가 *c:\wmq* 디렉토리에 있는지 확인하십시오.

   ```
   dir c:\wmq/D /B /S
   ```

9. 파일에 읽기 및 쓰기 또는 로컬 mqm 그룹에 대한 전체 제어 권한이 있는지 확인하십시오.

   ```
   cacls c:\wmq\*.*
   ```

다음에 수행할 작업

메시지를 큐에 넣거나 가져와서 큐 관리자를 테스트하십시오.

1. 큐 관리자를 시작하십시오.

   ```
   strmqm QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' starting.
   The queue manager is associated with installation '1'.
   5 log records accessed on queue manager 'QMGR' during the log
   replay phase.
   Log replay for queue manager 'QMGR' complete.
   Transaction manager state recovered for queue manager 'QMGR'.
   IBM MQ queue manager 'QMGR' started using 7.1.0.0.
   ```

2. 테스트 큐를 작성하십시오.

   ```
   echo define qlocal(QTEST) | runmqsc QMGR
   ```

   시스템이 다음과 같이 응답합니다.

```
5724-H72 (C) Copyright IBM Corp. 1994, 2023. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QMGR.


1 : define qlocal(QTEST)
AMQ8006: IBM MQ queue created.
One MQSC command read.
No commands have a syntax error.
All valid MQSC commands were processed.
```

3. 샘플 프로그램 **amqsput**을 사용하여 테스트 메시지를 넣으십시오.

   ```
   echo 'A test message' | amqsput QTEST QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   Sample AMQSPUT0 start
   target queue is QTEST
   Sample AMQSPUT0 end
   ```

4. 샘플 프로그램 **amqsget**을 사용하여 테스트 메시지를 가져오십시오.

   ```
   amqsget QTEST QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   Sample AMQSGET0 start
   message A test message
   Wait 15 seconds ...
   no more messages
   Sample AMQSGET0 end
   ```

5. 큐 관리자를 중지합니다.

   ```
   endmqm -i QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' ending.
   IBM MQ queue manager 'QMGR' ended.
   ```

6. 큐 관리자를 삭제합니다.

   ```
   dltmqm QMGR
   ```

   시스템이 다음과 같이 응답합니다.

   ```
   IBM MQ queue manager 'QMGR' deleted.
   ```

7. 작성한 디렉토리를 삭제하십시오.

팁: 명령이 각 파일 또는 디렉토리를 삭제하도록 프롬프트 표시되는 것을 막으려면 /Q 옵션을 명령에 추가하십시오.

```
del /F /S C:\wmq\*.*
rmdir /S C:\wmq
```

관련 태스크

Windows 로컬 mqm 그룹에 의해 권한이 부여된 데이터 및 로그 파일 읽기 및 쓰기
이 태스크에서는 선택한 디렉토리에 저장된 해당 데이터 및 로그 파일을 사용하여 큐 관리자를 작성하는 방법을 보여줍니다. 파일에 대한 액세스는 로컬 mqm 그룹에 의해 보안 설정됩니다. 이 디렉토리는 공유하지 않습니다.

Linux *Linux*에서 다중 인스턴스 큐 관리자 작성
이 예는 Linux에서 다중 인스턴스 큐 관리자를 설정하는 방법을 보여줍니다. 이 설정은 포함된 개념을 설명하기에는 작습니다. 예제는 Linux Red Hat Enterprise 5를 기반으로 합니다. 이 단계는 다른 UNIX 플랫폼에 따라 다릅니다.

이 태스크 정보

이 예제는 Windows 7 서비스 팩 1을 실행하는 3GB RAM이 있는 2GHz 노트북 컴퓨터에 설정됩니다. 두 개의 VMware 가상 머신(Server1 및 Server2)이 640MB 이미지에서 Linux Red Hat Enterprise 5를 실행합니다. Server1은 네트워크 파일 시스템(NFS), 큐 관리자 로그 및 HA 인스턴스를 호스팅합니다. NFS 서버가 큐 관리자 인스턴스 중 하나를 호스팅하는 것은 일반적인 사례가 아닙니다. 이는 예를 단순화하기 위한 것입니다. Server2는 대기 인스턴스로 Server1의 큐 관리자 로그를 마운트합니다. WebSphere MQ MQI 클라이언트는 Windows 7 서비스 팩 1을 실행하고 샘플 고가용성 애플리케이션을 실행하는 추가 400MB VMware 이미지에서 설치됩니다. 모든 가상 시스템은 보안상의 이유로 VMware 호스트 전용 네트워크의 일부로 구성됩니다.

참고: 큐 관리자 데이터만 NFS 서버에 배치해야 합니다. NFS에서는 마운트 명령과 함께 다음 세 가지 옵션을 사용하여 시스템에 보안을 설정하십시오.

- **noexec**
  이 옵션을 사용하면 NFS에서 2진 파일의 실행을 중지하여 원격 사용자가 시스템에서 원하지 않는 코드를 실행하지 못하게 합니다.
- **nosuid**
  이 옵션을 사용하면 set-user-identifier 및 set-group-identifier 비트 사용을 방지하여 원격 사용자가 상위 권한을 얻지 못하게 합니다.
- **nodev**
  이 옵션을 사용하면 문자 및 블록 특수 디바이스의 사용 또는 정의를 중지하여 원격 사용자가 chroot에서 벗어나지 않게 합니다.

프로시저

1. root로 로그인하십시오.
2. IBM MQ 설치 - 개요를 읽고 해당 링크에 따라 IBM MQ을(를) 설치하고 mqm 사용자 및 그룹을 작성한 후 /var/mqm을(를) 정의하십시오.
3. 공유 파일 시스템 작동 확인 태스크를 완료하여 파일 시스템이 다중 인스턴스 큐 관리자를 지원하는지 확인하십시오.
4. Server1에 대해 다음 단계를 완료하십시오.
   a. 공유될 공용 폴더 /MQHA에서 로그 및 데이터 디렉토리를 작성하십시오. 예를 들면, 다음과 같습니다.
      i) **mkdir** */MQHA*
      ii) **mkdir** */MQHA/logs*
      iii) **mkdir** */MQHA/qmgrs*
5. Server2에 대해 다음 단계를 완료하십시오.

a. 공유 파일 시스템을 마운트할 /MQHA 폴더를 작성하십시오. 경로를 Server1에서와 같이 유지하십시오. 예를 들면, 다음과 같습니다.

i) **mkdir** */MQHA*

6. MQHA 디렉토리를 사용자 및 그룹 mqm이 소유하고 사용자 및 그룹에 대해 액세스 권한이 rwx로 설정되는지 확인하십시오. 예를 들어 **ls -al** 에는 drwxrwxr-x mqm mqm 4096 Nov 27 14:38 MQDATA 이 표시됩니다.

   a. **chown -R** mqm:mqm */MQHA*

   b. **chmod -R** ug+rwx */MQHA*

7. 다음 명령을 입력하여 큐 관리자를 작성하십시오. **crtmqm -ld** */MQHA/logs* **-md** */MQHA/qmgrs QM1*

8. 추가[2]/MQHA *(rw,sync,no_wdelay,fsid=0)을 /etc/exports로 지정하십시오.

9. Server1에 대해 다음 단계를 완료하십시오.

   a. NFS 디먼을 시작하십시오. */etc/init.d/* **nfs** start

   b. Server1의 큐 관리자 구성 세부사항을 복사하십시오.

```
dspmqinf -o command QM1
```

   그리고 그 결과를 클립보드에 복사하십시오.

```
addmqinf -s QueueManager
-v Name=QM1
-v Directory=QM1
-v Prefix=/var/mqm
-v DataPath=/MQHA/qmgrs/QM1
```

10. Server2에 대해 다음 단계를 완료하십시오.

   a. **mount -t** nfs4 **-o** hard,intr *Server1:/ /MQHA* 명령을 입력하여 내보낸 파일 시스템 /MQHA 을 마운트하십시오.

   b. 큐 관리자 구성 명령을 Server2에 붙여넣으십시오.

```
addmqinf -s QueueManager
-v Name=QM1
-v Directory=QM1
-v Prefix=/var/mqm
-v DataPath=/MQHA/qmgrs/QM1
```

11. **-x** 매개변수를 사용하여 순서대로 큐 관리자 인스턴스를 시작하십시오. **strmqm -x** QM1.

   큐 관리자 인스턴스를 시작하는 데 사용되는 명령은 **addmqinf** 명령과 동일한 IBM MQ 설치에서 실행해야 합니다. 다른 설치에서 큐 관리자를 시작하고 중지하려면 먼저 **setmqm** 명령을 사용하여 큐 관리자와 연관된 설치를 설정해야 합니다. 자세한 정보는 setmqm을 참조하십시오.

Linux *Linux*에서 다중 인스턴스 큐 관리자 확인

샘플 프로그램 **amqsghac**, **amqsphac** 및 **amqsmhac**를 사용하여 다중 인스턴스 큐 관리자 구성을 확인합니다. 이 주제에서는 Linux Red Hat Enterprise 5에서 다중 인스턴스 큐 관리자 구성을 확인하는 구성 예를 제공합니다.

고가용성 샘플 프로그램은 자동 클라이언트 다시 연결을 사용합니다. 연결된 큐 관리자가 실패하면 클라이언트가 같은 큐 관리자 그룹에 있는 큐 관리자에 다시 연결하려고 시도합니다. 샘플 고가용성 샘플 프로그램에 대한 설명에서는 단순성을 위해 단일 인스턴스 큐 관리자를 사용한 클라이언트 다시 연결을 보여줍니다. 동일한 샘플을 다중 인스턴스 큐 관리자와 사용하여 다중 인스턴스 큐 관리자 구성을 확인할 수 있습니다.

[2] '*'는 읽기/쓰기를 위해 이 마운트 /MQHA에 도달할 수 있는 모든 시스템을 허용합니다. 프로덕션 시스템에 대한 액세스를 제한합니다.

이 예에서는 436 페이지의 『Linux에서 다중 인스턴스 큐 관리자 작성』에 설명된 다중 인스턴스 구성을 사용합니다. 이 구성을 사용하여 다중 인스턴스 큐 관리자가 대기 인스턴스로 전환하는지 확인하십시오. **endmqm** 명령을 사용하여 큐 관리자를 중지하고 전환 옵션인 -s를 사용하십시오. 클라이언트 프로그램이 새 큐 관리자 인스턴스에 다시 연결하여 잠깐의 지연 후 새 인스턴스에 대한 작업을 계속합니다.

예에서는 클라이언트가 Windows 7 서비스 팩 1 시스템에서 실행됩니다. 이 시스템은 다중 인스턴스 큐 관리자를 실행하는 두 VMware Linux 서버를 호스팅합니다.

IBM MQ Explorer를 사용한 장애 복구 확인

장애 복구를 확인하기 위해 샘플 애플리케이션을 사용하기 전에 각 서버에서 IBM MQ Explorer를 실행하십시오. **리모트 큐 관리자 추가 > 멀티 인스턴스 큐 관리자에 직접 연결** 마법사를 사용하여 각 탐색기에 큐 관리자 인스턴스 모두를 추가하십시오. 대기를 허용하여 두 인스턴스 모두 실행되고 있는지 확인하십시오. 활성 인스턴스로 VMware 이미지를 실행하는 창을 닫아서 가상으로 서버의 전원을 끄거나 활성 인스턴스를 중지하여 대기 인스턴스로의 전환을 허용하십시오.

참고: 서버의 전원을 끄는 경우 /MQHA을(를) 호스트하는 서버가 아닌지 확인하십시오.

참고: 대기 인스턴스로 전환 허용 옵션은 **큐 관리자 중지** 대화 상자에서 사용할 수 없습니다. 이 옵션은 큐 관리자가 단일 인스턴스 큐 관리자로 실행되고 있기 때문에 누락되었습니다. **대기 인스턴스 허용** 옵션 없이 큐 관리자를 시작했습니다. 큐 관리자 중지 요청이 거부된 경우 **자세한 내용** 창을 확인하십시오. 이는 실행 중인 대기 인스턴스가 없기 때문일 수 있습니다.

샘플 프로그램을 사용하여 장애 복구 확인

활성 인스턴스를 실행할 서버 선택

서버 중 MQHA 디렉토리 또는 파일 시스템을 호스팅할 서버를 선택했을 수 있습니다. 활성 서버를 실행 중인 VMware 창을 닫아 장애 복구를 테스트할 계획인 경우 이 서버는 MQHA를 호스팅하는 서버가 아니어야 합니다!

활성 큐 관리자 인스턴스를 실행하는 서버에서

참고: MCAUSER가 mqm으로 설정된 SVRCONN 채널을 실행하면 예의 구성 단계 수를 줄이는 데 편리합니다. 다른 사용자 ID를 선택하고 시스템이 예에서 사용된 것과 다르게 설정된 경우 액세스 권한 문제점을 경험할 수 있습니다. 노출된 시스템에서 mqm을 MCAUSER로 사용하지 마십시오. 상당한 보안성 저하를 초래할 수 있습니다.

1. *ipaddr1* 및 *ipaddr2*를 수정하고 다음 명령을 /MQHA/hasamples.tst.에 저장하십시오.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN)  TRPTYPE(TCP) +
MCAUSER('mqm') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(' ipaddr1 (1414), ipaddr2
(1414)') QMNAME(QM1) REPLACE
START  CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
START  LISTENER(LISTENER.TCP)
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

2. 경로 /MQHA로 터미널 창을 열고 다음 명령을 실행하십시오.

```
runmqsc -m QM1 < hasamples.tst
```

3. **runmqsc** 명령의 출력을 검사하여 리스너가 실행 중이고 큐 관리자 제어를 가지는지 확인하십시오.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

또는 TCPIP 리스너가 실행 중이고 Control = Queue Manager를 가지는 IBM MQ Explorer를 사용하여 확인하십시오.

클라이언트에서

1. 서버에서 `/MQHA/qmgrs/QM1.000/@ipcc`의 클라이언트 연결 테이블 AMQCLCHL.TAB을 클라이언트의 `C:\`에 복사하십시오.
2. 경로 `C:\`로 명령 프롬프트를 열고 클라이언트 채널 정의 테이블(CCDT)을 가리키도록 환경 변수 `MQCHLLIB`를 설정하십시오.

```
SET MQCHLLIB=C:\
```

3. 명령 프롬프트에 다음 명령을 입력하십시오.

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsphac SOURCE QM1
```

활성 큐 관리자 인스턴스를 실행하는 서버에서

1. 다음 중 하나입니다.
   - 활성 서버 인스턴스로 VMware 이미지를 실행하는 창을 닫으십시오.
   - IBM MQ Explorer를 사용하여 활성 큐 관리자 인스턴스를 중지하여 대기 인스턴스로의 전환을 허용하고 다시 연결 가능한 클라이언트에 다시 연결하도록 지시하십시오.
2. 세 클라이언트는 결국 연결이 끊어진 것을 감지한 후 다시 연결합니다. 이 구성에서 서버 창을 닫으면 세 연결 모두 재설정되는 데 약 7분 정도 소요됩니다. 일부 연결은 다른 연결 전에 정상적으로 재설정됩니다.

결과

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsghac TARGET QM1
Sample AMQSGHAC start
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

Multi 다중 인스턴스 큐 관리자 삭제

멀티플랫폼에서 다중 인스턴스 큐 관리자를 완전히 삭제하려면 **dltmqm** 명령을 사용하여 큐 관리자를 삭제한 다음 **rmvmqinf** 또는 **dltmqm** 명령을 사용하여 다른 서버에서 인스턴스를 제거하십시오.

dltmqm 명령을 실행하여 다른 서버에서 정의된 인스턴스가 있는 큐 관리자를 이 큐 관리자가 정의되는 서버에서 삭제하십시오. 큐 관리자를 작성한 서버와 같은 서버에서 **dltmqm** 명령을 실행할 필요는 없습니다. 그런 다음 큐 관리자의 정의가 있는 다른 모든 서버에서 **rmvmqinf** 또는 **dltmqm** 명령을 실행하십시오.

중지된 큐 관리자만 삭제할 수 있습니다. 큐 관리자를 삭제할 때 실행 중인 인스턴스가 없고 큐 관리자가 단일 인스턴스 큐 관리자 또는 다중 인스턴스 큐 관리자가 아니며 단순히 원격 공유에 해당 큐 관리자 데이터 및 로그가 있는 큐 관리자입니다. 큐 관리자를 삭제하면 해당 큐 관리자 데이터 및 로그가 삭제되고 **dltmqm** 명령을 발행한 서버의 mqs.ini 파일에서 큐 관리자 스탠자가 제거됩니다. 큐 관리자를 삭제할 때 큐 관리자 데이터 및 로그가 포함된 네트워크 공유에 대한 액세스 권한이 있어야 합니다.

이전에 큐 관리자의 인스턴스를 작성한 다른 서버의 mqs.ini 파일에도 여러 항목이 있습니다. 각 서버에 순서대로 방문하고 **rmvmqinf** *Queue manager stanza name* 명령을 실행하여 큐 관리자 스탠자를 제거해야 합니다.

Linux UNIX UNIX and Linux 시스템에서 네트워크 스토리지에 공용 mqs.ini 파일을 배치했고 각 서버에서 AMQ_MQS_INI_LOCATION 환경 변수를 설정하여 모든 서버에서 이 파일을 참조한 경우 업데이트할 한 개의 mqs.ini 파일만 있기 때문에 해당 서버 중 한 서버에서만 큐 관리자를 삭제해야 합니다.

예

첫 번째 서버
: **dltmqm** QM1

인스턴스가 정의되는 다른 서버
: **rmvmqinf** QM1 또는

: **dltmqm** QM1

Multi 다중 인스턴스 큐 관리자 시작 및 중지

단일 인스턴스 또는 다중 인스턴스 큐 관리자로 멀티플랫폼에 구성된 큐 관리자를 시작 및 중지합니다.

한 쌍의 서버에서 다중 인스턴스 큐 관리자를 정의한 경우 두 서버 중 하나에서 단일 인스턴스 큐 관리자 또는 다중 인스턴스 큐 관리자로 이 큐 관리자를 실행할 수 있습니다.

다중 인스턴스 큐 관리자를 실행하려면 **strmqm** -x *QM1* 명령을 사용하여 서버 중 하나에서 큐 관리자를 시작하십시오. -x 옵션은 인스턴스가 장애를 복구하도록 합니다. 이는 활성 인스턴스가 됩니다. 같은 **strmqm** -x *QM1* 명령을 사용하여 다른 서버에서 대기 인스턴스를 시작하십시오. -x 옵션은 인스턴스가 대기로 시작할 수 있게 합니다.

큐 관리자는 모든 요청을 처리하는 한 개의 활성 인스턴스와 활성 인스턴스가 실패하면 인수할 준비가 되어 있는 대기 인스턴스로 실행되고 있습니다. 활성 인스턴스에는 큐 관리자 데이터 및 로그에 대한 독점 액세스 권한이 부여됩니다. 대기는 큐 관리자 데이터 및 로그에 대한 독점 액세스 권한이 부여될 때까지 대기합니다. 대기에 독점 액세스 권한이 부여되는 경우 활성 인스턴스가 됩니다.

활성 인스턴스에서 **endmqm** -s 명령을 발행하여 수동으로 제어를 대기 인스턴스로 전환할 수도 있습니다. **endmqm** -s명령은 대기를 종료하지 않고 활성 인스턴스를 종료합니다. 큐 관리자 데이터 및 로그에 대한 독점 액세스 잠금이 해제되고 이를 대기가 인수합니다.

여러 서버에서 다중 인스턴스로 구성된 큐 관리자를 단일 인스턴스 큐 관리자로 시작하고 중지할 수도 있습니다. **strmqm** 명령에 -x 옵션을 사용하지 않고 큐 관리자를 시작하면 다른 시스템에서 구성된 큐 관리자의 인스턴스가 대기 인스턴스로 시작되지 않습니다. 다른 인스턴스를 시작하려고 시도하면 큐 관리자 인스턴스가 대기로 실행되는 것이 허용되지 않는다는 응답을 수신합니다.

-s 옵션 없이 **endmqm** 명령을 사용하여 다중 인스턴스 큐 관리자의 활성 인스턴스를 중지하는 경우 활성 및 대기 인스턴스 둘 다 중지됩니다. -x 옵션과 함께 **endmqm** 명령을 사용하여 대기 인스턴스를 중지하면 대기 인스턴스 상태가 중지되고 활성 인스턴스는 실행을 계속합니다. 대기에서 -x 옵션 없이 **endmqm**을 발행할 수 없습니다.

두 개의 큐 관리자 인스턴스만 동시에 실행될 수 있으며 이 중 하나는 활성 인스턴스이고 다른 하나는 대기 인스턴스입니다. 두 인스턴스를 동시에 시작하면 IBM MQ는 어떤 인스턴스가 활성 인스턴스가 되는지에 대해 제어할 수 없고 이는 네트워크 파일 시스템에 의해 판별됩니다. 큐 관리자 데이터에 대한 독점 액세스를 획득하는 첫 번째 인스턴스가 활성 인스턴스가 됩니다.

참고: 실패한 큐 관리자를 재시작하기 전에 큐 관리자의 해당 인스턴스로부터 애플리케이션의 연결을 끊어야 합니다. 이를 수행하지 않으면 큐 관리자가 올바르게 재시작하지 않을 수 있습니다.

Multi 공유 파일 시스템

멀티플랫폼에서 다중 인스턴스 큐 관리자는 네트워크 파일 시스템을 사용하여 큐 관리자 인스턴스를 관리합니다.

다중 인스턴스 큐 관리자는 파일 시스템 잠금과 공유 큐 관리자 데이터 및 로그의 결합을 사용하여 장애 복구를 자동화합니다. 큐 관리자의 한 개의 인스턴스만 공유 큐 관리자 데이터 및 로그에 대한 독점 액세스 권한을 가질 수 있습니다. 액세스하게 되면 활동 인스턴스가 됩니다. 독점 액세스 권한을 가져오는 데 성공하지 못한 다른 인스턴스는 큐 관리자 데이터 및 로그가 사용 가능하게 될 때까지 대기 인스턴스로 대기합니다.

네트워크 파일 시스템은 활성 큐 관리자 인스턴스에 대해 보유하고 있는 잠금 해제를 담당합니다. 활성 인스턴스가 어떤 식으로든 실패하면 네트워크 파일 시스템이 활성 인스턴스에 대해 보유하고 있는 잠금을 해제합니다. 독점 잠금이 해제되자마자 잠금 대기 중인 대기 큐 관리자가 이를 획득하려고 시도합니다. 획득에 성공하면 활성 인스턴스가 되고 공유 큐 시스템에 있는 큐 관리자 데이터 및 로그에 대한 독점 액세스 권한을 가지게 됩니다. 그러면 시작을 계속합니다.

관련 주제 파일 시스템 지원 계획에서는 파일 시스템이 다중 인스턴스 큐 관리자를 지원하도록 설정하고 확인하는 방법을 설명합니다.

다중 인스턴스 큐 관리자는 파일 시스템에서의 실패에 대해 보호하지 않습니다. 데이터를 보호하기 위한 다양한 방법이 있습니다.

- 중복 디스크 어레이(RAID)와 같은 신뢰할 수 있는 스토리지에 투자하고 이를 네트워크 복원성이 있는 네트워크 파일 시스템에 포함시키십시오.
- IBM MQ 선형 로그를 대체 매체에 백업하고 기본 로그 매체가 실패할 경우 대체 매체에 있는 로그를 사용하여 복구하십시오. 백업 큐 관리자를 사용하여 이 프로세스를 관리할 수 있습니다.

Multi 다중 큐 관리자 인스턴스

다중 인스턴스 큐 관리자는 실패 후 큐 관리자 가용성을 복원하는 데 대기 큐 관리자 인스턴스를 사용하기 때문에 복원성이 있습니다.

큐 관리자 인스턴스 복제는 큐 관리자 프로세스의 가용성을 향상시키기 위한 매우 효과적인 방법입니다. 설명만을 위해 단순 가용성 모델을 사용할 때 한 큐 관리자 인스턴스의 신뢰도가 99%(1년 동안, 누적 정지 시간은 3.65일임)인 경우 큐 관리자의 다른 인스턴스를 추가하면 가용성이 99.99%로 증가합니다(1년 동안, 누적 정지 시간은 약 1시간임).

이는 가용성의 실제 숫자 추정치를 제시하기에는 너무 단순한 모델입니다. 가용성을 현실적으로 모델링하려면 평균 실패 간격(MTBF) 및 평균 복구 시간(MTTR), 실패와 복구 시간 사이의 시간 확률 분배에 대한 통계를 수집해야 합니다.

용어 다중 인스턴스 큐 관리자는 큐 관리자 데이터 및 로그를 공유하는 큐 관리자의 활성 인스턴스와 대기 인스턴스의 결합을 나타냅니다. 다중 인스턴스 큐 관리자는 한 서버에 한 개의 활성인 큐 관리자 인스턴스를 두고 활성

인스턴스가 실패할 경우 자동으로 인계받을 준비가 되어 있는 대기 상태의 다른 큐 관리자 인스턴스를 다른 서버에 둠으로써 큐 관리자 프로세스의 실패에 대해 보호합니다.

Multi 장애 복구 또는 전환

대기 큐 관리자 인스턴스는 요청 시(전환) 또는 활성 인스턴스가 실패하는 경우(장애 복구) 활성 인스턴스로부터 인수합니다.

- 전환은 활성 큐 관리자 인스턴스로 발행되는 **endmqm** `-s` 명령에 대한 응답으로 대기 인스턴스가 시작될 때 발생합니다. **endmqm** 매개변수 `-c`, `-i` 또는 `-p`를 지정하여 큐 관리자가 갑자기 중지되는 수준을 제어할 수 있습니다.

  참고: 전환은 대기 큐 관리자 인스턴스가 이미 시작된 경우에만 발생합니다. **endmqm** `-s` 명령은 활성 큐 관리자 잠금을 해제하고 전환을 허용합니다. 이 명령은 대기 큐 관리자 인스턴스를 시작하지 않습니다.
- 장애 복구는 활성 인스턴스가 보유하는 큐 관리자 데이터에 대한 잠금이 해제되면 발생합니다. 이 인스턴스가 예상치 못하게 중지되는 것으로 표시되었기 때문입니다(즉 **endmqm** 명령이 발행되지 않음)

대기 인스턴스가 활성 인스턴스로 인수하면 큐 관리자 오류 로그에 메시지를 기록합니다.

다시 연결 가능한 클라이언트는 큐 관리자가 실패하거나 전환할 때 자동으로 다시 연결됩니다. 클라이언트 다시 연결을 요청하기 위해 `-r` 플래그를 **endmqm** 명령에 포함시키지 않아도 됩니다. 자동 클라이언트 다시 연결은 IBM MQ classes for Java에서 지원되지 않습니다.

장애 복구가 발생했고 대기 인스턴스가 활성이 된 후에도 실패한 인스턴스를 재시작할 수 없는 경우 실패한 인스턴스에 로컬로 연결된 애플리케이션이 실패한 인스턴스와의 연결을 끊었는지 여부를 확인하십시오.

로컬로 연결된 애플리케이션은 실패한 인스턴스가 재시작될 수 있도록 실패한 큐 관리자 인스턴스와의 연결을 끊거나 종료되어야 합니다. 공유 바인딩을 사용하고(기본 설정임) 실패한 인스턴스에 대한 연결을 계속 보유하는 로컬로 연결된 애플리케이션은 인스턴스가 재시작되는 것을 막습니다.

로컬로 연결된 애플리케이션을 종료하거나 로컬 큐 관리자 인스턴스가 실패할 경우 연결을 끊는 것이 불가능한 경우 격리 바인딩 사용을 고려해 보십시오. 격리 바인딩을 사용하는 로컬로 연결된 애플리케이션은 연결을 끊지 않더라도 로컬 큐 관리자가 재시작되는 것을 막지 않습니다.

Multi 채널 및 클라이언트 다시 연결

채널 및 클라이언트 다시 연결은 대기 큐 관리자 인스턴스가 활성이 된 후의 메시지 처리 복원에서 필수적인 부분입니다.

다중 인스턴스 큐 관리자 인스턴스는 여러 네트워크 주소로 서버에서 설치됩니다. 모든 큐 관리자 인스턴스에 대한 연결 정보를 사용하여 IBM MQ 채널 및 클라이언트를 구성해야 합니다. 대기가 인수를 하면 클라이언트 및 채널이 새 네트워크 주소에서 새로 활성이 된 큐 관리자 인스턴스에 자동으로 다시 연결됩니다. 자동 클라이언트 다시 연결은 IBM MQ classes for Java에서 지원되지 않습니다.

이 설계는 HA-CMP 작업과 같은 고가용성 환경 방식과는 다릅니다. HA-CMP는 클러스터에 대한 가상 IP 주소를 제공하고 이 주소를 활성 서버에 전송합니다. IBM MQ 다시 연결은 IP 주소를 변경하거나 다시 라우팅하지 않습니다. 이는 채널 정의 및 클라이언트 연결에서 정의한 네트워크 주소를 사용한 다시 연결을 통해 작동합니다. 관리자로서 채널 정의 및 다중 인스턴스 큐 관리자의 모든 인스턴스에 대한 연결에서 네트워크 주소를 정의해야 합니다. 다중 인스턴스 큐 관리자에 대해 네트워크 주소를 구성하는 최고의 방법은 연결에 따라 다릅니다.

큐 관리자 채널

채널의 `CONNAME` 속성은 연결 이름의 쉼표로 구분된 목록이며 예를 들면, 다음과 같습니다. `CONNAME('127.0.0.1(1234), 192.0.2.0(4321)')`. 연결은 연결이 성공적으로 설정될 때까지 연결에서 지정된 순서대로 시도됩니다. 연결에 실패하면 채널이 다시 연결을 시도합니다.

클러스터 채널

일반적으로 다중 인스턴스 큐 관리자가 클러스터에서 작동하게 하기 위해서 필요한 추가 구성은 없습니다.

큐 관리자가 저장소 큐 관리자에 연결하는 경우 저장소는 큐 관리자의 네트워크 주소를 검색합니다. 큐 관리자에서 `CLUSRCVR` 채널의 `CONNAME`을 참조합니다. `TCPIP`에서 사용자가 `CONNAME`을 생략하거나 공백으로 구성한 경우 큐 관리자는 자동으로 이 옵션을 설정합니다. 대기 인스턴스가 인수하면 해당 `IP` 주소가 이전 활성 인스턴스의 `IP` 주소를 `CONNAME`으로 바꿉니다.

필요한 경우 큐 관리자 인스턴스의 네트워크 주소 목록으로 `CONNAME`을 수동으로 구성할 수 있습니다.

클라이언트 연결

클라이언트 연결에서는 연결 목록 또는 큐 관리자 그룹을 사용하여 대체 연결을 선택할 수 있습니다. 클라이언트는 IBM WebSphere MQ 7.0.1 클라이언트 라이브러리 또는 그 이상과 실행하기 위해 컴파일되어야 합니다. 클라이언트는 IBM WebSphere MQ 7.0.1 이상의 큐 관리자와 연결되어야 합니다.

장애 복구가 발생하면 다시 연결에 시간이 걸립니다. 대기 큐 관리자는 그 시동을 완료해야 합니다. 실패한 큐 관리자와 연결된 클라이언트는 연결 실패를 감지하고 새 클라이언트 연결을 시작해야 합니다. 새 클라이언트 연결이 새로 활성이 된 대기 큐 관리자를 선택하면 클라이언트가 동일한 큐 관리자에 다시 연결됩니다.

클라이언트가 다시 연결 중 MQI 호출 중간에 있게 되면 호출 완료까지 길어진 대기를 허용해야 합니다.

메시지 채널에서 배치 전송 중에 실패가 발생하면 배치가 롤백되고 재시작됩니다.

전환이 장애 복구보다 빠르며 큐 관리자의 한 인스턴스를 중지하고 다른 인스턴스를 시작하는 시간만 소요됩니다. 재실행할 로그 레코드 수가 적은 큐 관리자의 경우 전환은 길어봐야 몇 초 정도 걸릴 수 있습니다. 장애 복구에 걸리는 시간을 추정하려면 실패가 감지되는 데 걸리는 시간을 추가해야 합니다. 감지에는 길어봐야 10초 정도 걸리지만 네트워크 및 파일 시스템에 따라 몇 분이 될 수도 있습니다.

Multi 애플리케이션 복구

애플리케이션 복구는 장애 복구 후 자동화된 애플리케이션 연속 처리입니다. 장애 복구 후 애플리케이션 복구에는 주의 깊은 설계가 필요합니다. 일부 애플리케이션은 장애 복구가 발생했음을 인지해야 합니다.

애플리케이션 복구의 목적은 애플리케이션이 잠깐 동안의 지연 후에 처리를 계속하게 하는 것입니다. 새 처리를 계속하기 전에 애플리케이션은 백아웃하고 실패 중 처리하고 있었던 작업 단위를 다시 제출해야 합니다.

애플리케이션 복구의 문제점은 IBM MQ MQI client와 큐 관리자 간 공유되고 큐 관리자에 저장되는 컨텍스트가 손실되는 것입니다. IBM MQ MQI client는 대부분의 컨텍스트를 저장하지만 신뢰할 수 있는 수준으로 복원할 수 없는 컨텍스트 부분이 있습니다. 다음 절에서는 애플리케이션 복구의 일부 특성 및 그러한 특성이 다중 인스턴스 큐 관리자에 연결된 애플리케이션의 복구에 영향을 주는 방식에 대해 설명합니다.

트랜잭션 가능한 메시징

메시지 전달의 관점에서 장애 복구는 IBM MQ 메시징의 지속적 특성을 변경하지 않습니다. 메시지가 지속적이고 작업 단위 내에서 올바르게 관리되는 경우 장애 복구 중 메시지가 손실되지 않습니다.

트랜잭션 처리의 관점에서 트랜잭션은 백아웃되거나 장애 복구 후 커미트됩니다.

커미트되지 않은 트랜잭션은 롤백됩니다. 장애 복구 후 다시 연결 가능한 애플리케이션은 트랜잭션이 실패했음을 나타내는 `MQRC_BACKED_OUT` 이유 코드를 수신합니다. 그런 다음 트랜잭션을 다시 재시작해야 합니다.

커미트된 트랜잭션은 2단계 커미트의 두 번째 단계에 도달한 트랜잭션이거나 `MQCMIT`를 시작한 1단계(메시지만 있음) 트랜잭션입니다.

큐 관리자가 트랜잭션 코디네이터이고 `MQCMIT`가 실패 전 해당 2단계 커미트의 두 번째 단계를 시작한 경우에 트랜잭션이 성공적으로 완료됩니다. 완료는 큐 관리자의 제어를 받으며 큐 관리자가 다시 실행될 때 계속됩니다. 다시 연결 가능한 애플리케이션에서 `MQCMIT` 호출은 정상적으로 완료됩니다.

메시지만 포함되는 1단계 커미트에서 커미트 처리를 시작한 트랜잭션은 일단 다시 실행되면 큐 관리자의 제어 하에 정상적으로 완료됩니다. 다시 연결 가능한 애플리케이션에서 `MQCMIT`가 정상적으로 완료됩니다.

다시 연결 가능한 클라이언트는 큐 관리자의 제어 하에 1단계 트랜잭션을 트랜잭션 코디네이터로 사용할 수 있습니다. 확장된 트랜잭션 클라이언트는 다시 연결을 지원하지 않습니다. 트랜잭션 가능한 클라이언트가 연결할 때 다시 연결이 요청되면 연결에 성공하지만 다시 연결되는 기능은 없습니다. 이 연결은 다시 연결 불가능한 것처럼 작동합니다.

애플리케이션 재시작 또는 계속

장애 복구는 애플리케이션을 인터럽트합니다. 실패 후 애플리케이션은 처음부터 재시작하거나 인터럽트 후 처리를 계속할 수 있습니다. 후자를 자동 클라이언트 다시 연결이라고 합니다. 자동 클라이언트 다시 연결은 IBM MQ classes for Java에서 지원되지 않습니다.

IBM MQ MQI client 애플리케이션을 사용하여 클라이언트를 자동으로 다시 연결하도록 연결 옵션을 설정해야 합니다. 이 옵션은 `MQCNO_RECONNECT` 또는 `MQCNO_RECONNECT_Q_MGR`입니다. 옵션을 설정하지 않으면 클

라이언트가 자동으로 다시 연결을 시도하지 않으며 큐 관리자 실패는 MQRC_CONNECTION_BROKEN을 클라이언트에 리턴합니다. 새 MQCONN 또는 MQCONNX 호출을 발행하여 클라이언트가 새 연결을 시도하고 시작하도록 설계할 수 있습니다.

서버 프로그램은 재시작되어야 하고 큐 관리자 또는 서버가 실패했을 때 처리하고 있었던 시점에서 큐 관리자에 의해 자동으로 다시 연결되어야 합니다. IBM MQ 서버 프로그램은 일반적으로 다중 인스턴스 큐 관리자 인스턴스가 실패할 때 대기 큐 관리자 인스턴스에서 재시작되지 않습니다.

다음 두 가지 방식으로 IBM MQ 서버 프로그램이 대기 서버에서 재시작하도록 자동화할 수 있습니다.

1. 서버 애플리케이션을 큐 관리자 서비스로 패키지화하십시오. 서버 애플리케이션은 대기 큐 관리자를 재시작할 때 재시작됩니다.
2. 대기 큐 관리자 인스턴스가 시작할 때 작성한 장애 복구 로그 메시지 등으로 트리거되는 자체 장애 복구 논리를 작성하십시오. 그러면 애플리케이션 인스턴스가 시작 후 MQCONN 또는 MQCONNX를 호출하여 큐 관리자에 대한 연결을 작성해야 합니다.

장애 복구 감지

일부 애플리케이션은 장애 복구를 인지해야 하고 나머지 애플리케이션을 인지하지 않아도 됩니다. 다음 두 가지 예를 고려하십시오.

1. 메시징 채널을 통해 메시지를 가져오거나 수신하는 메시징 애플리케이션에서는 일반적으로 채널의 다른 측 큐 관리자가 실행되도록 요구하지 않습니다. 이 애플리케이션은 채널의 다른 측에 있는 큐 관리자가 대기 인스턴스에서 재시작되는 경우 영향을 받을 가능성이 거의 없습니다.
2. An IBM MQ MQI client application processes persistent message input from one queue and puts persistent message responses onto another queue as part of a single unit of work: if it handles an MQRC_BACKED_OUT reason code from MQPUT, MQGET or MQCMIT within sync point by restarting the unit of work, then no messages are lost. 또한 이 애플리케이션은 연결 실패를 처리하기 위한 특수 처리를 수행하지 않아도 됩니다.

그러나 두 번째 예에서는 애플리케이션이 MQGET 옵션인 MQGMO_MSG_UNDER_CURSOR을 사용하여 처리할 메시지를 선택하기 위해 큐를 찾아본다고 가정하십시오. 다시 연결은 찾아보기 커서를 재설정하고 MQGET 호출은 올바른 메시지를 리턴하지 않습니다. 이 예에서 애플리케이션은 장애 복구가 발생했음을 인지해야 합니다. 또한 커서 아래 메시지에 대해 다른 MQGET을 발행하기 전에 이 애플리케이션은 이 브라우저 커서를 복원해야 합니다.

브라우저 커서 손실은 애플리케이션 컨텍스트가 다음 다시 연결을 변경하는 방식의 한 가지 예입니다. 다른 경우는 445 페이지의 『자동으로 다시 연결된 클라이언트의 복구』에 설명되어 있습니다.

장애 복구 후 IBM MQ MQI client 애플리케이션에 대한 세 가지 대체 설계 패턴이 있습니다. 이들 중 하나만 장애 복구를 감지하지 않아도 됩니다.

다시 연결 안함

이 패턴에서는 연결이 끊긴 경우 애플리케이션이 현재 연결에 대한 모든 처리를 중지합니다. 처리를 계속하려면 이 애플리케이션은 큐 관리자와의 새 연결을 설정해야 합니다. 이 애플리케이션은 이 애플리케이션이 새 연결에 대한 처리를 계속하기 위해 필요로 하는 상태 정보 전송에 전적으로 책임이 있습니다. 연결이 끊긴 후 큐 관리자와 다시 연결되는 기존 클라이언트 애플리케이션은 이 방식으로 작성됩니다.

클라이언트는 연결이 끊긴 후 다음 MQI 호출에서 MQRC_CONNECTION_BROKEN 또는 MQRC_Q_MGR_NOT_AVAILABLE과 같은 이유 코드를 수신합니다. 애플리케이션은 큐 핸들과 같은 모든 IBM MQ 상태 정보를 제거하고 새 연결을 설정하기 위해 새 MQCONN 또는 MQCONNX 호출을 발행한 다음 처리해야 하는 IBM MQ 오브젝트를 다시 열어야 합니다.

기본 MQI 작동은 큐 관리자와의 연결이 끊기면 큐 관리자 연결 핸들이 사용할 수 없게 되는 것입니다. 기본값은 장애 복구 후 애플리케이션 다시 연결을 방지하기 위해 MQCONNX에 MQCNO_RECONNECT_DISABLED 옵션을 설정하는 것과 같습니다.

장애 복구 허용

장애 복구의 영향을 받지 않도록 애플리케이션을 작성하십시오. 주의하여 오류를 핸들링하는 것으로 장애 복구 처리에 충분한 경우가 있습니다.

다시 연결 인지

MQCBT_EVENT_HANDLER 이벤트 핸들러를 큐 관리자에 등록하십시오. 이 이벤트 핸들러는 클라이언트가 서버에 다시 연결하기 위한 시도를 시작하면 MQRC_RECONNECTING로 게시되고 성공적인 다시 연결 후에는 MQRC_RECONNECTED로 게시됩니다. 그런 다음 클라이언트 애플리케이션이 처리를 계속할 수 있도록 예상 가능한 상태를 재설정하는 루틴을 시작할 수 있습니다.

자동으로 다시 연결된 클라이언트의 복구

장애 복구는 예상치 못한 이벤트이고 자동으로 다시 연결된 클라이언트가 설계한대로 작동하도록 다시 연결의 결과가 예상 가능해야 합니다.

예상치 못한 실패를 예상 가능하고 신뢰할 수 있는 복구로 전환하는 주요 요소는 트랜잭션의 사용입니다.

이전 절에서는 예 444 페이지의 『2』가 로컬 트랜잭션을 사용하여 MQGET 및 MQPUT을 조정하는 IBM MQ MQI client의 예로 제공되었습니다. 클라이언트는 MQRC_BACKED_OUT 오류에 대한 응답으로 MQCMIT 또는 MQBACK 호출을 발행한 다음 백아웃된 트랜잭션을 다시 제출합니다. 큐 관리자 실패로 인해 트랜잭션은 백아웃되고 클라이언트 애플리케이션의 작동은 트랜잭션 및 메시지가 손실되지 않도록 합니다.

모든 프로그램 상태가 트랜잭션의 일부로 관리되는 것은 아니기 때문에 다시 연결의 결과가 더 이해하기 어려워집니다. 클라이언트 애플리케이션이 큐 관리자 장애 복구에서 존속되도록 설계하기 위해서 다시 연결이 IBM MQ MQI client의 상태를 변경하는 방식을 알아야 합니다.

특수 장애 복구 코드 없이 다른 오류와 같은 논리를 사용하여 다시 연결 오류를 핸들링하는 애플리케이션을 설계하도록 결정할 수 있습니다. 또는 다시 연결에 특수 오류 처리가 필요함을 인식하고 이벤트 핸들러를 IBM MQ에 등록하여 장애 복구를 핸들링하기 위한 루틴을 실행하도록 선택할 수 있습니다. 이 루틴은 다시 연결 처리 그 자체를 핸들링하거나 처리를 재개할 때 복구 처리를 수행해야 하는 기본 프로그램 스레드를 표시하는 플래그를 설정할 수 있습니다.

IBM MQ MQI client 환경은 장애 복구 자체를 인지하고, 해당 IBM MQ 상태를 복원하기 위해 클라이언트에 일부 상태 정보를 저장하고 클라이언트 애플리케이션을 대신해서 추가 MQI 호출을 발행하여 다시 연결 후 가능한 한 많은 컨텍스트를 복원합니다. 예를 들어, 실패 지점에서 열린 오브젝트에 대한 핸들이 복원되고 임시 동적 큐가 동일한 이름으로 열립니다. 그러나 불가피한 변경사항이 있고 설계가 이러한 변경사항을 처리해야 합니다. 이러한 변경사항은 다섯 가지 종류로 분류할 수 있습니다.

1. 일치하는 새 컨텍스트 상태가 애플리케이션 프로그램에 의해 복원될 때까지 새 오류 또는 이전에 진단되지 않은 오류가 MQI 호출에서 리턴됩니다.

   새 오류를 수신하는 예는 컨텍스트를 저장한 후 다시 연결 전에 컨텍스트를 전달하려고 시도할 때의 리턴 코드 MQRC_CONTEXT_NOT_AVAILABLE입니다. 이 컨텍스트는 다시 연결 후에는 복원할 수 없습니다. 보안 컨텍스트는 권한이 없는 클라이언트 프로그램에 전달되지 않기 때문입니다. 전달될 경우 악성 애플리케이션 프로그램이 보안 컨텍스트를 확보할 수 있습니다.

   일반적으로 애플리케이션은 주의하여 설계된 방식으로 공통 오류 및 예상 가능한 오류를 핸들링하고 비공통 오류는 일반 오류 핸들러로 격하시킵니다. 이 오류 핸들러는 IBM MQ에서 연결을 끊고 다시 연결하거나 프로그램을 모두 중지할 수도 있습니다. 연속성을 향상시키기 위해 일부 오류를 다른 방식으로 처리해야 될 수도 있습니다.
2. 비지속 메시지는 손실될 수 있습니다.
3. 트랜잭션은 롤백됩니다.
4. 동기점 외부에서 사용된 MQGET 또는 MQPUT 호출은 가능한 메시지 손실로 인터럽트될 수 있습니다.
5. MQI 호출에서의 연장된 대기 시간으로 인해 발생한 타이밍 오류입니다.

손실된 컨텍스트에 대한 일부 세부사항이 다음 절에 나열됩니다.

- NPMCLASS(HIGH) 옵션을 사용하여 큐에 넣지 않거나 큐 관리자 실패가 종료 시 비지속 메시지 저장 옵션을 인터럽트한 경우 비지속 메시지는 버려집니다.
- 연결이 끊기면 지속 불가능 구독이 손실됩니다. 다시 연결 시 이 구독은 재설정됩니다. 지속 가능 구독의 사용을 고려해 보십시오.
- 가져오기-대기 간격이 다시 계산되고 그 한계가 초과되면 MQRC_NO_MSG_AVAILABLE을 리턴합니다. 마찬가지로 구독 만기가 다시 계산되어 동일한 전체 만기 시간을 제공합니다.

- 큐에서 찾아보기 커서의 위치가 손실되고 이는 일반적으로 첫 번째 메시지 전에 재설정됩니다.
  - MQGMO_BROWSE_MSG_UNDER_CURSOR 또는 MQGMO_MSG_UNDER_CURSOR를 지정하는 MQGET 호출이 이유 코드 MQRC_NO_MSG_AVAILABLE로 실패합니다.
  - 찾아보기에 대해 잠겨진 메시지의 잠금이 해제됩니다.
  - 핸들 범위가 있는 찾아보기 표시 메시지는 표시가 취소되고 다시 찾아볼 수 있습니다.
  - 협업 찾아보기 표시 메시지는 대부분의 경우 표시가 취소됩니다.
- 보안 컨텍스트는 손실됩니다. MQPMO_PASS_ALL_CONTEXT로 메시지 넣기가 MQRC_CONTEXT_NOT_AVAILABLE로 실패함과 같이 저장된 메시지 컨텍스트를 사용하려고 시도합니다.
- 메시지 토큰이 손실됩니다. 메시지 토큰을 사용하는 MQGET은 이유 코드 MQRC_NO_MSG_AVAILABLE을 리턴합니다.

  참고: 메시지의 일부인 *MsgId* 및 *CorrelId*는 장애 복구 중에 메시지와 함께 보존되므로 MsgId 또는 CorrelId를 사용하는 MQGET은 예상대로 작동합니다.
- 커미트되지 않은 트랜잭션에서 동기점 아래 큐에 넣은 메시지는 더 이상 사용 가능하지 않습니다.
- 논리 순서로 또는 메시지 그룹으로 메시지를 처리하면 다시 연결 후 리턴 코드 MQRC_RECONNECT_INCOMPATIBLE이 발생합니다.
- MQI 호출은 클라이언트가 요즘 일반적으로 수신하는 보다 일반적인 MQRC_CONNECTION_BROKEN이 아닌 MQRC_RECONNECT_FAILED를 리턴할 수 있습니다.
- 메시지가 큐 관리자에 전달되었는지 여부를 IBM MQ MQI client 가 알지 못하는 경우 동기점 외부에서 MQPUT 호출 중에 다시 연결하면 MQRC_CALL_INTERRUPTED 가 리턴됩니다. MQCMIT 중 다시 연결도 비슷하게 작동합니다.
- MQRC_CALL_INTERRUPTED가 리턴됨 - 성공적인 다시 연결 후 - IBM MQ MQI client가 큐 관리자로부터 다음 항목의 성공 또는 실패를 나타내는 응답을 수신하지 않은 경우
  - 동기점 외부 MQPUT 호출을 사용한 지속 메시지의 전달
  - 동기점 외부 MQPUT1 호출을 사용한 지속 메시지 또는 기본값 지속성을 가진 메시지의 전달
  - MQCMIT 호출을 사용하는 트랜잭션의 커미트. 응답은 다시 연결에 성공한 후에만 리턴됩니다.
- 채널이 새 인스턴스(다른 채널일 수도 있음)로 재시작되어 채널 엑시트 상태가 보유됩니다.
- 임시 동적 큐가 열린 다시 연결 가능한 클라이언트 처리의 일부로 임시 동적 큐가 복원됩니다. 임시 동적 큐의 어떤 메시지도 복원되지 않지만 큐가 열려 있거나 큐의 이름을 기억하는 애플리케이션은 처리를 계속할 수 있습니다.

  큐를 작성한 애플리케이션 외의 애플리케이션에서 해당 큐가 사용되고 있는 경우 이 큐는 다음에 참조될 때 표시될 정도로 빠르게 복원되지 못할 가능성이 있습니다. 예를 들어, 클라이언트가 응답 대상 큐로 임시 동적 큐를 작성하고 응답 메시지가 채널에 의해 큐에 배치되어야 하는 경우 이 큐는 제 시간에 복구되지 않을 수 있습니다. 이 경우 일반적으로 채널은 응답 대상 메시지를 데드-레터 큐에 넣습니다.

  다시 연결 가능한 클라이언트 애플리케이션이 이름으로(다른 애플리케이션이 이름을 이미 작성했으므로) 임시 동적 큐를 여는 경우 다시 연결할 때 작성에 사용할 모델이 없기 때문에 IBM MQ MQI client에서 임시 동적 큐를 다시 작성할 수 없습니다. MQI에서는 한 개의 애플리케이션만 모델에 의해 임시 동적 큐를 열 수 있습니다. 임시 동적 큐를 사용하려고 하는 다른 애플리케이션은 MQPUT1 또는 서버 바인딩을 사용하거나 실패할 경우 다시 연결을 다시 시도할 수 있어야 합니다.

  비지속 메시지만 임시 동적 큐에 넣을 수 있으며 이러한 메시지는 장애 복구 중에 손실됩니다. 이 손실은 다시 연결 중 MQPUT1을 사용하여 임시 동적 큐에 넣는 메시지에 대해 True입니다. MQPUT1 중에 장애 복구가 발생하면 MQPUT1이 성공하더라도 이 메시지를 넣을 수 없습니다. 이 문제점에 대한 임시 해결책 하나는 영구 동적 큐를 사용하는 것입니다. 서버 바인딩 애플리케이션은 다시 연결 가능하지 않기 때문에 임시 동적 큐를 이름으로 열 수 있습니다.

Multi 데이터 복구 및 고가용성

다중 인스턴스 큐 관리자를 사용한 고가용성 솔루션에는 스토리지 실패 후 데이터를 복구하는 메커니즘이 포함되어야 합니다.

다중 인스턴스 큐 관리자는 큐 관리자 프로세스의 가용성을 높이지만 파일 시스템과 같이 큐 관리자가 메시지 및 기타 정보를 저장하는 데 사용하는 다른 컴포넌트의 가용성은 높이지 않습니다.

데이터의 가용성을 높이는 한 가지 방법은 네트워크 복원성 데이터 스토리지를 사용하는 것입니다. 네트워크 파일 시스템 및 복원성 데이터 스토리지를 사용하여 자체 솔루션을 빌드하거나 통합 솔루션을 구매할 수 있습니다. 복원성과 재해 복구를 결합하려는 경우 수십 또는 수백 킬로미터가 넘는 디스크 복제를 허용하는 비동기 디스크 복제가 사용 가능합니다.

매체를 최대한 이용하기 위해 여러 IBM MQ 디렉토리를 스토리지 매체에 맵핑하는 방식을 구성할 수 있습니다. 다중 인스턴스 큐 관리자의 경우 두 가지 유형의 IBM MQ 디렉토리와 파일 사이에는 중요한 차이가 있습니다.

큐 관리자의 인스턴스 사이에 공유되어야 하는 디렉토리입니다.
: 큐 관리자의 여러 인스턴스 사이에서 공유되어야 하는 정보는 `qmgrs` 및 `logs` 두 디렉토리에 있습니다. 이러한 디렉토리는 공유된 네트워크 파일 시스템에 있어야 합니다. 데이터는 메시지가 작성되고 삭제될 때마다 계속해서 변경되기 때문에 지속적 고가용성과 우수한 성능을 제공하는 스토리지 매체를 사용하도록 권장됩니다.

큐 관리자의 인스턴스 사이에서 공유되지 않아도 되는 디렉토리 및 파일입니다.
: 일부 다른 디렉토리는 큐 관리자의 여러 인스턴스 사이에서 공유되지 않아도 되며 미러링된 파일 시스템 외의 수단으로 빠르게 복원됩니다.

  - IBM MQ 실행 가능 파일 및 도구 디렉토리입니다. 재설치 또는 백업된 파일 아카이브에서 백업 및 복원하여 바꾸십시오.
  - 설치를 위해 전체적으로 수정되는 구성 정보입니다. 구성 정보는 IBM MQ, 예를 들어 Windows의 `mqs.ini` 파일, UNIX and Linux 시스템 또는 사용자 고유의 구성 관리 (예: **`MQSC`** 구성 스크립트) 의 일부로 관리됩니다. 파일 아카이브를 사용하여 백업하고 복원하십시오.
  - 추적, 오류 로그 및 FFDC 파일과 같은 설치 전체 출력입니다. 파일은 `errors` 및 기본 데이터 디렉토리의 `trace` 서브디렉토리에 저장됩니다. UNIX and Linux 시스템의 기본 데이터 디렉토리는 `/var/mqm`입니다. Windows에서 기본 데이터 디렉토리는 IBM MQ 설치 디렉토리입니다.

백업 큐 관리자를 사용하여 선형 로깅을 사용하는 다중 인스턴스 큐 관리자의 정기적인 매체 백업을 수행할 수 있습니다. 백업 큐 관리자는 미러링된 파일 시스템만큼 빠른 복구를 제공하고 마지막 백업 이후 변경사항은 복구하지 않습니다. 백업 큐 관리자 메커니즘은 로컬화된 스토리지 실패 후 큐 관리자 복구보다는 외부 재해 복구 시나리오에서의 사용에 더 적절합니다.

IBM MQ 가용성 솔루션 결합

애플리케이션은 가용성을 향상시키기 위해 다른 IBM MQ 기능을 사용합니다. 다중 인스턴스 큐 관리자는 다른 고가용성 기능을 보완합니다.

IBM MQ 클러스터는 큐 가용성을 증가시킴

클러스터 내 각 관리자에 있는 모든 큐 중 하나까지 클러스터 큐의 다중 정의를 작성하여 큐 가용성을 높일 수 있습니다.

클러스터의 한 멤버가 실패한 후 새 메시지가 클러스터 큐에 송신된다고 가정하십시오. 메시지가 실패한 큐 관리자에게 이동해야 하는 경우가 아니면 이 메시지는 큐의 정의가 있는 클러스터 내 다른 실행 큐 관리자로 송신됩니다.

클러스터가 가용성을 상당히 높이긴 하지만 메시지가 지연되는 결과를 가져올 수 있는 두 개의 관련 실패 시나리오가 있습니다. 다중 인스턴스 큐 관리자로 클러스터를 빌드하면 메시지가 지연될 가능성이 줄어듭니다.

고립된 메시지
: 클러스터의 큐 관리자가 실패하면 클러스터 내 다른 큐 관리자에게 라우트될 수 있는 메시지가 실패한 큐 관리자로 더 이상 라우트되지 않습니다. 이미 송신된 메시지는 실패한 큐 관리자가 재시작될 때까지 고립됩니다.

연관관계
: 연관관계는 둘 사이에 공유되거나 그렇지 않으면 개별 계산 사이에서 공유되는 정보를 설명하는 데 사용되는 용어입니다. 예를 들어, 연관관계는 요청 메시지를 서버에 송신하는 애플리케이션과 응답을 처리할 것으로 예상되는 샘플 애플리케이션 사이에 존재합니다. 또 다른 예는 이전 메시지에 따른 각 메시지의 처리인 메시지의 순서입니다.

메시지를 클러스터된 큐에 송신하는 경우 연관관계를 고려해야 합니다. 동일한 큐 관리자에게 연속 메시지를 송신해야 합니까? 또는 각 메시지는 클러스터의 임의 멤버로 이동할 수 있습니까?

메시지를 클러스터 내 동일한 큐 관리자에게 송신해야 하고 이 송신이 실패할 경우 메시지는 실패한 클러스터 큐 관리자가 다시 실행될 때까지 송신자의 전송 큐에서 대기합니다.

클러스터가 다중 인스턴스 큐 관리자로 구성되는 경우 실패한 큐 관리자가 재시작될 때까지 대기하는 지연은 1분 정도로 제한되거나 잠시 동안 대기가 인수합니다. 대기가 실행되면 고립된 메시지는 처리를 계속하고 새로 활성화된 큐 관리자 인스턴스에 대한 채널이 시작되며 전송 큐에서 대기하고 있었던 메시지가 플로우를 시작합니다.

실패한 큐 관리자에 의해 메시지가 지연되는 것을 극복하도록 클러스터를 구성하는 가능한 방법은 서로 다른 두 개의 큐 관리자를 클러스터 내 각 서버에 배치하고 그 중 하나는 활성으로, 다른 하나는 대기 인스턴스로 배열하는 것입니다. 이는 활성 대기 구성이고 클러스터의 가용성을 높입니다.

관리가 줄고 확장성이 증가하는 이점 외에도 클러스터는 계속해서 추가 가용성 요소를 제공하여 다중 인스턴스 큐 관리자를 보완합니다. 클러스터는 큐 관리자의 활성 및 대기 인스턴스 둘 다에 영향을 미치는 다른 유형의 실패에 대해 보호합니다.

인터럽트되지 않는 서비스
: 클러스터는 인터럽트되지 않는 서비스를 제공합니다. 클러스터에 의해 수신되는 새 메시지는 처리되기 위해 활성 큐 관리자로 송신됩니다. 인터럽트되지 않는 서비스를 제공하기 위해 다중 인스턴스 큐 관리자에만 의존하지 마십시오. 대기 큐 관리자가 실패를 감지하고 그 시동을 완료하고, 해당 채널이 다시 연결되며, 실패한 메시지 배치가 다시 제출되는 데에는 시간이 걸리기 때문입니다.

로컬화된 가동 중단
: 활성, 대기 및 파일 시스템 서버는 허용 가능한 성능을 제공하면서 밀리초 속도로 상호작용하므로 그 거리에 대한 실제적 제한이 있습니다.

: 클러스터된 큐 관리자는 수초 정도의 속도의 상호작용을 요구하며 지리적으로는 전 세계 어디로든 분산될 수 있습니다.

조작 오류
: 가용성을 높이기 위해 두 개의 서로 다른 메커니즘을 사용하면 인적 오류와 같은 조작상의 오류가 가용성을 높이기 위한 노력을 반감시킬 가능성이 줄어듭니다.

큐 공유 그룹은 메시지 처리 가용성을 높임

z/OS z/OS에서만 제공되는 큐 공유 그룹에서는 큐 관리자 그룹이 큐에 대한 서비스를 공유할 수 있습니다. 한 큐 관리자가 실패하면 다른 큐 관리자가 큐에 있는 모든 메시지를 계속해서 처리합니다. 다중 인스턴스 큐 관리자는 z/OS에서 지원되지 않으며 더 넓은 메시징 아키텍처의 일부로서만 큐 공유 그룹을 보완합니다.

IBM MQ 클라이언트는 애플리케이션 가용성을 높임

IBM MQ MQI client 프로그램은 큐 관리자 가용성, 연결 가중치 및 연관관계를 기반으로 하여 큐 관리자 그룹 내 여러 큐 관리자에 연결할 수 있습니다. 애플리케이션이 연결된 큐 관리자 인스턴스가 실패할 경우 애플리케이션을 다시 연결할 수 있는 방법이 있는 한 큐 관리자가 실행되고 있는 시스템과 다른 시스템에서 애플리케이션을 실행하여 솔루션의 전체 가용성을 향상시킬 수 있습니다.

큐 관리자 그룹은 클라이언트를 중지된 큐 관리자로부터 분리하고 IP 스프레이어와 같이 큐 관리자 그룹 전체에 클라이언트 연결을 로드 밸런싱하여 클라이언트 가용성을 높이는 데 사용됩니다. 이 클라이언트 애플리케이션에는 실패한 큐 관리자와의 연관관계(예: 특정 큐에 대한 종속 항목)가 없어야 합니다. 그렇지 않으면 처리를 계속할 수 없습니다.

자동 클라이언트 다시 연결과 다중 인스턴스 큐 관리자는 일부 연관관계 문제점을 해결하여 클라이언트 가용성을 높입니다. 자동 클라이언트 다시 연결은 IBM MQ classes for Java에서 지원되지 않습니다.

`MQCNO` 옵션 `MQCNO_RECONNECT_Q_MGR`을 설정하여 클라이언트가 동일한 큐 관리자에 다시 연결하도록 강제 실행할 수 있습니다.

1. 이전에 연결된 단일 인스턴스 큐 관리자가 실행되지 않을 경우 이 큐 관리자가 다시 실행될 때까지 연결이 다시 시도됩니다.
2. 큐 관리자가 다중 인스턴스 큐 관리자로 구성되면 클라이언트가 활성인 인스턴스에 다시 연결합니다.

동일한 큐 관리자에 자동으로 다시 연결하면 큐 관리자가 클라이언트를 대신해서 보유하고 있었던 대부분의 상태 정보(예: 클라이언트가 열었던 큐 및 구독했던 토픽)가 복원됩니다. 클라이언트가 요청에 대한 응답을 수신하기 위해 동적 응답 대상 큐를 연 경우 이 응답 대상 큐에 대한 연결도 복원됩니다.

V 9.0.4 Linux RDQM 고가용성

복제 데이터 큐 관리자(RDQM)는 Linux 플랫폼에서 사용 가능한 고가용성 솔루션입니다.

RDQM 구성은 각각 큐 관리자의 인스턴스가 있는 고가용성(HA) 그룹에서 구성된 세 서버로 구성됩니다. 하나의 인스턴스는 다른 두 인스턴스에 해당 데이터를 동시에 복제하는, 실행 중인 큐 관리자입니다. 이 큐 관리자를 실행하는 서버가 실패하면 큐 관리자의 다른 인스턴스가 시작되고 현재 데이터를 사용하여 작동합니다. 큐 관리자의 세 인스턴스는 부동 IP 주소를 공유하므로 클라이언트는 단일 IP 주소로만 구성해야 합니다. 네트워크 문제점으로 인해 HA 그룹이 파티션되더라도 한 번에 한 개의 큐 관리자 인스턴스만 실행할 수 있습니다. 큐 관리자를 실행 중인 서버는 '1차'로 알려져 있으며 다른 두 서버는 각각 '2차'로 알려져 있습니다.

스플릿 브레인 상황이 발생하는 가능성을 크게 줄이기 위해 세 개의 노드가 사용됩니다. 2-노드의 고가용성 시스템에서 스플릿 브레인은 두 노드 간 연결이 끊어질 때 발생할 수 있습니다. 연결이 없으면 두 노드가 동시에 큐 관리자를 실행하고 서로 다른 데이터를 축적하게 됩니다. 연결이 복원되면 데이터에 두 개의 다른 버전이 있게 되고('스플릿 브레인'), 어떤 데이터를 유지하고 어떤 데이터를 버릴지를 사용자가 직접 개입하여 결정해야 합니다.

RDQM은 스플릿 브레인 상황을 피하기 위해 쿼럼이 있는 세 개의 노드 시스템을 사용합니다. 하나 이상의 다른 노드와 통신할 수 있는 노드는 쿼럼을 구성합니다. 큐 관리자는 쿼럼이 있는 노드에서만 실행될 수 있습니다. 큐 관리자가 하나 이상의 노드에 연결되지 않은 노드에서 실행될 수 없으므로 동시에 두 노드에서 실행될 수 없습니다.

- 한 노드가 실패하면 큐 관리자는 다른 두 노드 중 하나에서 실행될 수 있습니다. 두 노드가 실패하면 노드가 쿼럼을 가지지 않기 때문에 큐 관리자는 남은 노드에서 실행될 수 없습니다(남은 노드는 다른 두 노드가 실패했는지 또는 여전히 실행 중인지 알 수 없으며 연결이 끊어집니다.).
- 한 노드가 연결이 끊기면 노드가 쿼럼을 보유하지 않으므로 큐 관리자를 이 노드에서 실행할 수 없습니다. 큐 관리자는 쿼럼을 보유하는, 남은 두 노드 중 하나에서 실행될 수 있습니다. 모든 노드가 연결이 끊어진 경우 큐 관리자는 쿼럼이 있는 노드가 없으므로 노드에서 큐 관리자를 실행할 수 없습니다.

참고: IBM MQ Console은 복제 데이터 큐 관리자를 지원하지 않습니다. IBM MQ Explorer를 복제 데이터 큐 관리자에 사용할 수 있지만 이는 RDQM 기능에 특정한 정보를 표시하지 않습니다.

세 노드의 그룹 구성은 Pacemaker에서 처리합니다. 세 노드 간의 복제는 DRBD가 처리합니다. (Pacemaker에 대한 정보는 https://clusterlabs.org/pacemaker/를 참조하고 DRBD에 대한 정보는 https://docs.linbit.com/docs/users-guide-9.0/을 참조하십시오.)

501 페이지의 『큐 관리자 데이터 백업』에서 설명된 프로세스를 사용하여 복제 데이터 큐 관리자를 백업할 수 있습니다. 큐 관리자를 중지하고 백업해도 RDQM 구성에서 수행되는 노드 모니터링에는 영향을 미치지 않습니다.

다음 그림은 HA 그룹의 세 노드 각각에서 실행 중인 RDQM이 있는 일반 배치를 보여줍니다.

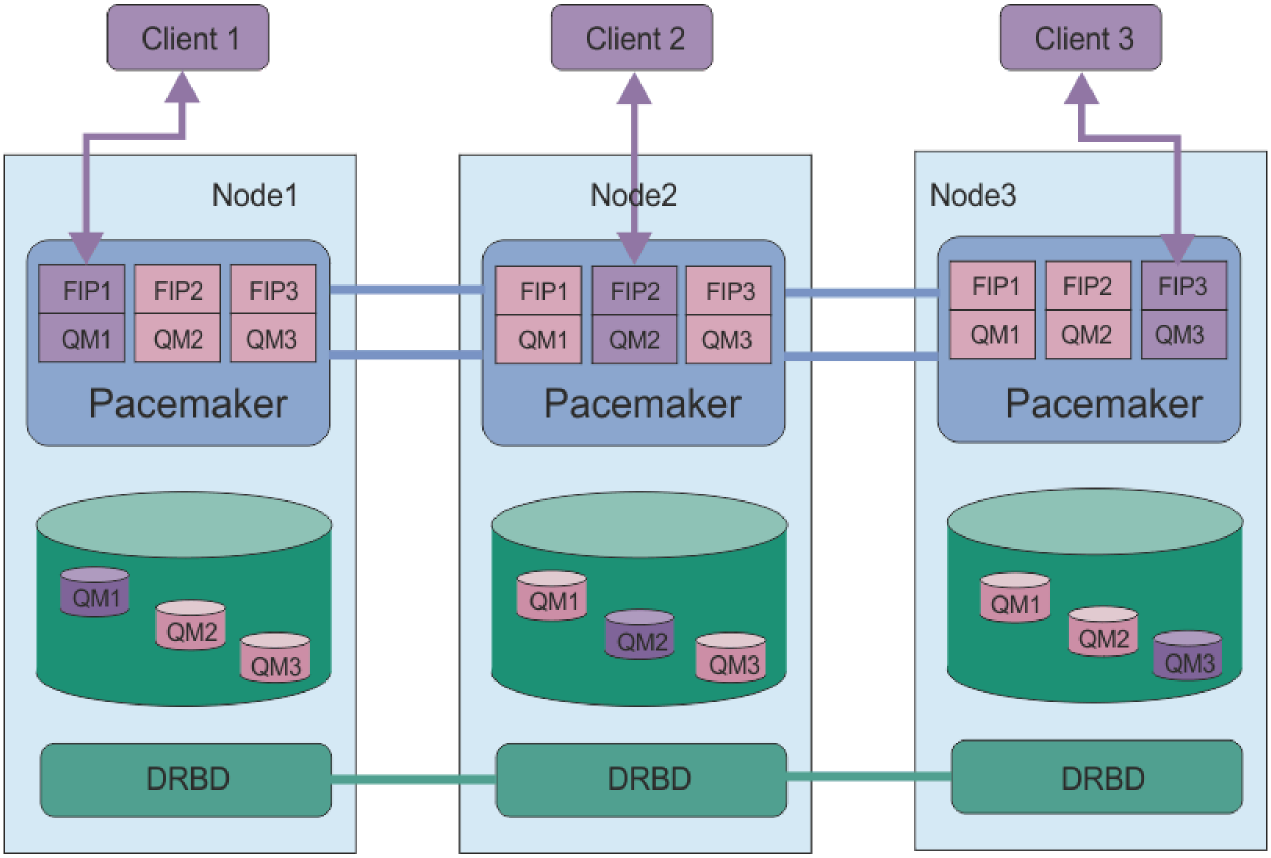


그림 *80.* 세 *RDQM*이 있는 *HA* 그룹의 예

다음 그림에서는 Node3가 실패하고 Pacemaker 링크가 유실되어 큐 관리자 QM3가 Node2에서 대신 실행됩니다.

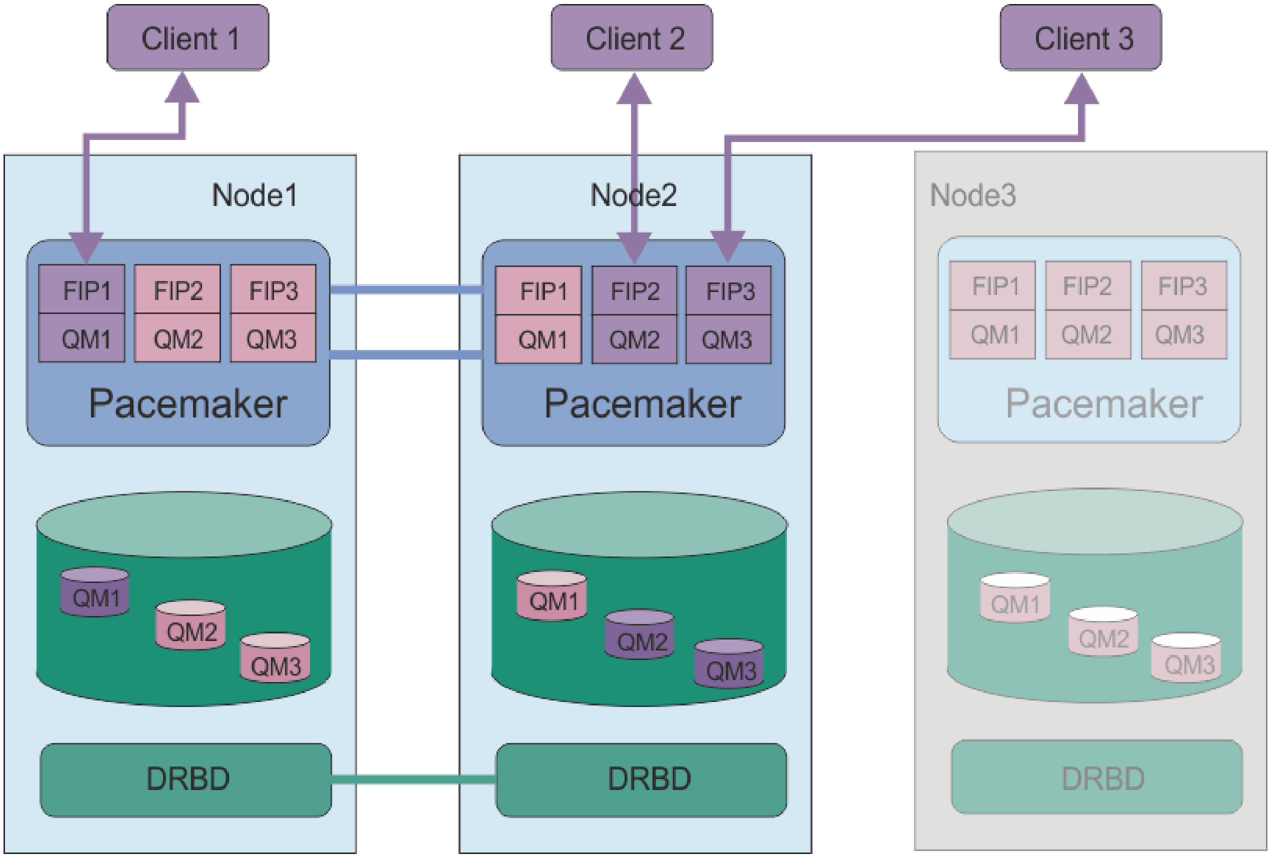


그림 *81. node3*가 실패 후 예

관련 정보
RDQM(복제된 데이터 큐 관리자) 설치
복제된 데이터 큐 관리자 마이그레이션

V 9.0.4 Linux *RDQM HA* 솔루션에 대한 요구사항

RDQM 고가용성(HA) 그룹을 구성하기 전에 여러 가지 요구사항을 충족시켜야 합니다.

시스템 요구사항

RDQM HA 그룹을 구성하기 전에 HA 그룹에 포함될 세 개의 서버 각각에서 몇 가지 구성을 완료해야 합니다.

- 각 노드에는 drbdpool이라는 볼륨 그룹이 필요합니다. 복제 데이터 큐 관리자 각각에 대한 스토리지는 이 볼륨 그룹에서 큐 관리자별로 별도의 논리적 볼륨으로 할당됩니다. 최상의 성능을 위해 이 볼륨 그룹은 하나 이상의 물리적 볼륨으로 구성되어야 합니다. 이때 각 물리적 볼륨은 내부 디스크 드라이브(SSD 선호)에 해당합니다. RDQM HA 솔루션을 설치한 후 실제로 RDQM을 작성하기 전에 drbdpool 을 작성해야 합니다. **vgs** 명령을 사용하여 볼륨 그룹 구성을 확인하십시오. 출력은 다음과 유사합니다.

  ```
  VG        #PV #LV #SN Attr   VSize   VFree
  drbdpool    1   9   0 wz--n- <16.00g <7.00g
  rhel        1   2   0 wz--n- <15.00g      0
  ```

  특히 속성의 여섯 번째 열에 c 문자가 없는지 확인하십시오 (즉, wz--nc). c은(는) 클러스터링이 사용 가능함을 표시하며, 볼륨 그룹을 삭제하고 클러스터링을 사용하지 않고 다시 작성해야 합니다.
- drbdpool 볼륨 그룹을 작성한 후에는 아무것도 하지 마십시오. drbdpool에 작성된 논리적 볼륨 및 이러한 볼륨의 마운트 방법과 위치를 IBM MQ에서 관리합니다.
- 각 노드에는 RDQM 지원을 구성하는 데 사용할 최대 세 개의 인터페이스가 필요합니다.
  - HA 그룹을 모니터하기 위한 Pacemake용 1차 인터페이스
  - HA 그룹을 모니터하기 위한 Pacemake용 대체 인터페이스

- 동기 데이터 복제용 인터페이스(복제 인터페이스). HA 그룹에서 실행되는 모든 복제 데이터 큐 관리자의 예상 워크로드를 감안할 경우 이 인터페이스가 복제 요구사항을 지원할 수 있을 정도의 충분한 대역폭을 제공해야 합니다.

세 인터페이스 모두에 동일한 IP 주소를 사용하거나, 각 인터페이스에 별도의 IP 주소를 사용하거나, 기본 및 대체 인터페이스에는 동일한 IP 주소를 사용하고 복제 인터페이스에는 별도의 IP 주소를 사용하도록 HA 그룹을 구성할 수 있습니다.

최대 결함 허용을 위해 이러한 인터페이스는 독립된 네트워크 인터페이스 카드(NIC)여야 합니다.

- DRBD에는 RFC 1123에 의해 수정된 RFC 952에 의해 정의된 대로 HA 그룹의 각 노드에 유효한 인터넷 호스트 이름(`uname -n`에 의해 리턴되는 값)이 있어야 합니다.
- HA 그룹의 노드 간에 방화벽이 있을 경우 방화벽이 포트 범위에서 노드 간 트래픽을 허용해야 합니다. 샘플 스크립트(`/opt/mqm/samp/rdqm/firewalld/configure.sh`)가 제공되는데, 이 스크립트는 표준 방화벽이 RHEL에서 실행 중인 경우 필요한 포트를 엽니다. `root`로 스크립트를 실행해야 합니다. 다른 방화벽을 사용하는 경우, 서비스 정의 `/usr/lib/firewalld/services/rdqm*` 을(를) 검토하여 열어야 하는 포트를 확인하십시오.
- 시스템이 허용 모드가 아닌 모드에서 SELinux를 사용하는 경우 다음 명령을 실행해야 합니다.

```
semanage permissive -a drbd_t
```

네트워크 요구사항

사용자가 동일한 데이터 센터에서 RDQM HA 그룹에 3개 노드를 배치하는 것이 좋습니다.

사용자가 다른 데이터 센터에 노드를 배치하도록 선택할 경우, 다음 제한사항을 유의하십시오.

- 데이터 센터 사이의 대기 시간이 늘어나면 성능이 급격하게 저하됩니다. IBM이 최대 5밀리초의 대기 시간을 지원하지만 애플리케이션 성능이 1-2밀리초의 대기 시간을 허용하지 못할 수도 있습니다.
- 복제 링크를 통해 전송된 데이터는 IBM MQ AMS를 사용하여 적용되는 것 이외의 추가 암호화가 적용되지 않습니다.

유동 IP 주소를 구성하면 복제 데이터 큐 관리자(RDQM)가 실행 중인 HA 그룹의 노드에 관계없이, 클라이언트가 RDQM에 대해 동일한 IP 주소를 사용할 수 있습니다. 부동 주소는 RDQM의 기본 노드에 있는 이름 지정된 물리적 인터페이스에 바인드됩니다. RDQM이 장애 복구되어 다른 노드가 기본 노드가 된 경우에는 새 기본 노드에 있는 동일한 이름의 인터페이스에 유동 IP가 바인드됩니다. 세 노드의 물리적 인터페이스는 모두 동일한 이름을 가져야 하고 유동 IP 주소와 동일한 서브넷에 속해 있어야 합니다.

클러스터 구성에 필요한 사용자 요구사항

`root` 사용자로 RDQM HA 그룹을 구성할 수 있습니다. `root`로 구성하지 않으려는 경우, 대신 mqm 그룹의 사용자로 구성하십시오. mqm 사용자가 RDQM 클러스터를 구성하려면 다음 요구사항을 충족시켜야 합니다.

- mqm 사용자는 sudo를 사용하여 RDQM HA 그룹을 구성하는 세 개의 서버 각각에서 명령을 실행할 수 있어야 합니다.
- mqm 사용자가 비밀번호 없이 SSH를 사용하여 RDQM HA 그룹을 구성하는 세 개의 서버 각각에서 명령을 실행할 수 있는 경우, 사용자는 한 서버에서만 명령을 실행해야 합니다.
- mqm 사용자에 대해 비밀번호 없는 SSH를 구성할 경우, 사용자가 세 서버 모두에서 동일한 UID를 사용해야 합니다.

mqm 사용자가 루트 권한으로 다음 명령을 실행할 수 있도록 sudo를 구성해야 합니다.

```
/opt/mqm/bin/crtmqm
/opt/mqm/bin/dltmqm
/opt/mqm/bin/rdqmadm
/opt/mqm/bin/rdqmstatus
```

큐 관리자 작업에 필요한 사용자 요구사항

복제 데이터 큐 관리자(RDQM)를 작성하거나 삭제하거나 구성하려면 mqm 그룹과 `haclient` 그룹(`haclient` 그룹은 Pacemaker의 설치 중에 작성됨) 모두에 속해 있는 사용자 ID를 사용해야 합니다.

V 9.0.4 Linux 비밀번호 없는 *SSH* 설정

HA 그룹의 한 노드에서만 구성 명령을 발행하도록 비밀번호 없는 SSH를 설정할 수 있습니다.

이 태스크 정보

비밀번호 없는 SSH를 설정하려면 각 노드에서 mqm ID를 구성한 다음 해당 사용자에 대해 각 노드에서 키를 생성해야 합니다. 그런 다음 키를 다른 노드에 분배하고 연결을 테스트하여 알려진 호스트 목록에 각 노드를 추가하십시오. 마지막으로 mqm ID를 잠그십시오.

참고: 이 지시사항은 사용자가 별도의 기본, 대체 및 복제 인터페이스를 사용하는 HA 그룹을 정의하고 있다고 가정하므로, 기본 및 대체 인터페이스에 대해 비밀번호 없는 SSH를 정의하십시오. 단일 IP 주소를 사용하는 시스템을 구성하려는 경우 해당 단일 인터페이스에 대해 비밀번호 없는 SSH를 정의하십시오.

프로시저

1. 세 노드 각각에서 다음 단계를 수행하여 mqm 사용자를 설정하고 SSH 키를 생성하십시오.
   a) mqm 홈 디렉토리를 `/home/mqm`으로 변경하십시오.

   ```
   usermod -d /home/mqm mqm
   ```

   b) `/home/mqm` 디렉토리를 작성하십시오.

   ```
   mkhomedir_helper mqm
   ```

   c) mqm 비밀번호를 추가하십시오.

   ```
   passwd mqm
   ```

   d) 대화식 쉘을 mqm으로 실행하십시오.

   ```
   su mqm
   ```

   e) mqm 인증 키를 생성하십시오.

   ```
   ssh-keygen -t rsa -f /home/mqm/.ssh/id_rsa -N ''
   ```

2. 세 노드 각각에서 다음 단계를 수행하여 해당 노드의 키를 다른 두 노드에 추가하고 각 노드의 기본 및 대체(사용된 경우) 주소에 대한 연결을 테스트합시오.
   a) 원격 노드에 키를 추가하십시오.

   ```
   ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node1_primary_address
   ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node1_alternate_address
   ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node2_primary_address
   ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node2_alternate_address
   ```

   b) 비밀번호 없는 SSH를 확인하고 원격 노드에 대해 known_hosts를 업데이트하십시오.

   ```
   ssh remote_node1_primary_address uname -n
   ssh remote_node1_alternate_address uname -n
   ssh remote_node2_primary_address uname -n
   ssh remote_node2_alternate_address uname -n
   ```

   연결마다 계속할지 확인하는 프롬프트가 표시됩니다. 매번 확인하여 known_hosts를 업데이트하십시오. 비밀번호 없는 SSH를 사용하여 HA 그룹을 구성하기 전에 이를 완료해야 합니다.

   c) mqm으로서 대화식 쉘을 종료하십시오.

   ```
   exit
   ```

3. 각 노드에서 루트로 다음 단계를 수행하여 mqm 비밀번호를 제거하고 ID를 잠그십시오.
   a) mqm 비밀번호를 제거하십시오.

   ```
   passwd -d mqm
   ```

   b) mqm을 잠그십시오.

   ```
   passwd -l mqm
   ```

4. 각 노드에서 루트로 다음 단계를 완료하여 mqm 사용자에 대한 sudo 액세스를 설정하십시오.
   a) /etc로 디렉토리를 변경한 후 sudoers 파일을 편집하십시오.

   ```
   cd /etc
   vi sudoers
   ```

   b) "## Allows people in group wheel to run all commands" 행을 검색하여 이 행 아래에 다음 텍스트를 추가하십시오.

   ```
   #%mqm    ALL=(ALL)       ALL
   ```

   c) "## Same thing without a password" 행을 검색하여 이 행 아래에 다음 텍스트를 추가하십시오.

   ```
   %mqm            ALL=(ALL)       NOPASSWD: ALL
   ```

V 9.0.4 Linux *Pacemaker* 클러스터*(HA* 그룹*)* 정의

HA 그룹은 Pacemaker 클러스터입니다. Pacemaker 클러스터를 정의하려면 /var/mqm/rdqm.ini 파일을 편집하고 **rdqmadm** 명령을 실행하십시오.

이 태스크 정보

Pacemaker에 대한 정보는 https://clusterlabs.org/pacemaker/의 내용을 참조하십시오. 사용자가 sudo를 사용할 수 있는 경우 mqm 그룹에 Pacemaker 클러스터를 사용자로 작성할 수 있습니다. 사용자가 비밀번호 없이 각 서버에 대해 SSH도 사용할 수 있는 경우, 서버 중 하나에서 rdqm.ini 파일을 편집하고 **rdqmadm**을 실행하기만 하면 Pacemaker 클러스터를 작성할 수 있습니다. 그렇지 않은 경우 노드로 사용될 각 서버에서 파일을 작성하고 root로 명령을 실행해야 합니다.

rdqm.ini 파일은 Pacemaker 클러스터에 있는 모든 노드에 대한 IP 주소를 제공합니다. Pacemaker 클러스터가 1-3개의 IP 주소를 사용하도록 지정할 수 있습니다. 동기 데이터 복제에 사용되는 인터페이스를 '복제 인터페이스'라고 합니다. HA 그룹에서 실행되는 모든 RDQM의 예상 워크로드를 감안할 경우 이 인터페이스가 복제 요구사항을 지원할 수 있을 정도의 충분한 대역폭을 제공해야 합니다. 시스템을 모니터하기 위해 기본 및 보조 인터페이스가 Pacemaker에 사용되지만, Pacemaker는 필요한 경우 이를 위해 복제 인터페이스를 사용할 수 있습니다.

다음 예제 파일은 각 인터페이스에 대해 별도의 IP 주소를 사용하는 예제 Pacemaker 클러스터의 구성을 보여줍니다.

```
Node:
  HA_Primary=192.168.4.1
  HA_Alternate=192.168.5.1
  HA_Replication=192.168.6.1
Node:
  HA_Primary=192.168.4.2
  HA_Alternate=192.168.5.2
  HA_Replication=192.168.6.2
Node:
  HA_Primary=192.168.4.3
  HA_Alternate=192.168.5.3
  HA_Replication=192.168.6.3
```

다음 예제 파일은 각 인터페이스에 대해 동일한 IP 주소를 사용하는 예제 Pacemaker 클러스터의 구성을 보여줍니다. 이 경우에는 복제 인터페이스만 지정하십시오.

```
Node:
  HA_Replication=192.168.4.1
Node:
  HA_Replication=192.168.4.2
Node:
  HA_Replication=192.168.4.3
```

두 개의 IP 주소를 사용하려고 했던 경우에는 `rdqm.ini` 파일에 노드별 `Primary` 및 `Replication` 필드가 있지만, `Alternate` 필드는 없습니다.

```
Node:
  HA_Primary=192.168.4.1
  HA_Replication=192.168.5.1
Node:
  HA_Primary=192.168.4.2
  HA_Replication=192.168.5.2
Node:
  HA_Primary=192.168.4.3
  HA_Replication=192.168.5.3
```

프로시저

- `mqm` 사용자로 Pacemaker 클러스터를 정의하려면 다음을 수행하십시오.
  a) 사용자 mqm이 **sudo**를 사용하여 명령을 실행할 수 있고 선택적으로 비밀번호 없이 SSH를 사용하여 각 서버에 연결할 수 있는지 확인하십시오.
  b) 세 개의 서버 중 하나에서 `/var/mqm/rdqm.ini` 파일을 편집하여 파일에 Pacemaker 클러스터를 정의하십시오.
  c) 다음 명령을 실행하십시오.

  ```
  rdqmadm -c
  ```

  비밀번호 없이 SSH를 사용할 수 없는 경우 `.ini` 파일을 각 서버에 복사한 후 각 서버에서 명령을 실행해야 합니다.
- `root` 사용자로 Pacemaker 클러스터를 정의하려면 다음을 수행하십시오.
  a) 세 개의 서버 중 하나에서 `/var/mqm/rdqm.ini` 파일을 편집하여 파일에 클러스터를 정의하십시오.
  b) Pacemaker 클러스터의 노드로 사용될 다른 두 서버로 파일을 복사하십시오.
  c) 세 개의 서버 각각에서 `root`로 다음 명령을 실행하십시오.

  ```
  rdqmadm -c
  ```

관련 정보

rdqmadm(복제된 데이터 큐 관리자 클러스터 관리)

V 9.0.4 Linux *Pacemaker* 클러스터*(HA* 그룹*)* 삭제

HA 그룹은 Pacemaker 클러스터입니다. **rdqmadm** 명령을 `-u` 옵션과 함께 실행하면 Pacemaker 클러스터 구성을 삭제할 수 있습니다.

이 태스크 정보

복제 데이터 큐 관리자가 노드에 남아 있을 경우에는 Pacemaker 클러스터 구성을 삭제할 수 없습니다.

프로시저

- Pacemaker 클러스터 구성을 삭제하려면 원하는 노드에서 다음 명령을 입력하십시오.

```
rdqmadm -u
```

관련 정보

rdqmadm(복제된 데이터 큐 관리자 클러스터 관리)

V 9.0.4 Linux *HA RDQM* 작성

crtmqm 명령을 사용하여 고가용성 복제 데이터 큐 관리자(RDQM)를 작성할 수 있습니다.

이 태스크 정보

mqm 사용자가 sudo를 사용할 수 있는 경우 mqm 그룹의 사용자로 고가용성 복제 데이터 큐 관리자(RDQM)를 작성할 수 있습니다. 사용자가 비밀번호 없이 각 노드에 대해 SSH도 사용할 수 있는 경우, 한 노드에서 RDQM 작성 명령을 실행하기만 하면 세 노드 모두에서 RDQM을 작성할 수 있습니다. 그렇지 않은 경우, root로 RDQM을 작성해야 하며 세 노드 모두에서 명령을 실행해야 합니다.

프로시저

- mqm 그룹의 사용자로 RDQM을 작성하려면 다음을 수행하십시오.
  a) mqm 사용자가 **sudo**를 사용하여 명령을 실행할 수 있고 비밀번호 없이 SSH를 사용하여 각 서버에 연결할 수 있는지 확인하십시오.
  b) 다음 명령을 입력하십시오.

     ```
     crtmqm -sx [-fs FilesystemSize] qmname
     ```

     여기서 *qmname*은 복제된 데이터 큐 관리자의 이름입니다. 선택적으로 큐 관리자의 파일 시스템 크기 (drbdpool 볼륨 그룹에 작성된 논리적 볼륨의 크기)를 지정할 수 있습니다.

     이 명령은 SSH를 사용하여 클러스터의 다른 노드에 mqm 사용자로 연결하려고 시도합니다. 연결에 성공하면 큐 관리자의 보조 인스턴스가 노드에 작성됩니다. 그렇지 않은 경우 보조 인스턴스를 작성한 다음 **crtmqm -sx** 명령(root 사용자의 경우 설명됨)을 실행해야 합니다.
- root 사용자로 RDQM을 작성하려면 다음을 수행하십시오.
  a) RDQM의 보조 인스턴스를 호스팅할 각 노드에서 다음 명령을 입력하십시오.

     ```
     crtmqm -sxs [-fs FilesystemSize] qmname
     ```

     여기서 *qmname*은 복제된 데이터 큐 관리자의 이름입니다. 선택적으로 큐 관리자의 파일 시스템 크기 (drbdpool 볼륨 그룹에 작성된 논리적 볼륨의 크기)를 지정할 수 있습니다. HA 그룹에 있는 세 노드 모두에서 RDQM 대해 동일한 파일 시스템 크기를 지정해야 합니다.

     이 명령은 RDQM의 보조 인스턴스를 작성합니다.
  b) 나머지 노드에서는 다음 명령을 입력하십시오.

     ```
     crtmqm -sx [-fs FilesystemSize] qmname
     ```

     여기서 *qmname*은 복제된 데이터 큐 관리자의 이름입니다. 선택적으로 큐 관리자의 파일 시스템 크기를 지정할 수 있습니다.

     이 명령은 큐 관리자의 보조 인스턴스가 다른 두 노드에 있는지 판별합니다. 보조 인스턴스가 있는 경우 이 명령은 기본 큐 관리자를 작성하고 시작합니다. 보조 인스턴스가 없는 경우 각 노드에서 **crtmqm -sxs** 명령을 실행해야 합니다.

  DataPath(**-md**) 및 LogPath(**-ld**) 인수 외에도 표준 Linux 큐 관리자를 작성하는 데 유효한 모든 인수는 기본 복제 데이터 큐 관리자에도 유효합니다.

관련 정보

crtmqm

V 9.0.4 Linux *HA RDQM* 삭제

dltmqm 명령을 사용하여 고가용성 복제 데이터 큐 관리자(RDQM)를 삭제할 수 있습니다.

이 태스크 정보

RDQM 삭제 명령은 RDQM의 기본 노드에서 실행해야 합니다. 먼저 RDQM을 종료해야 합니다. mqm 사용자가 필요한 sudo 권한을 보유하고 있는 경우 해당 사용자로 명령을 실행할 수 있습니다. 그렇지 않은 경우 스크립트를 루트로 실행해야 합니다. 기본 큐 관리자와 연관된 자원이 삭제되면 명령은 다른 노드에 연결하기 위해 SSH를 사용하여 보조 큐 관리자를 삭제하려고 시도합니다. 삭제하지 못할 경우 다른 노드에서 `dltmqm`을 수동으로 실행하여 프로세스를 완료해야 합니다. 기본 큐 관리자가 아직 삭제되지 않은 경우 보조 노드에서 명령이 실패합니다.

프로시저

- RDQM을 삭제하려면 다음 명령을 입력하십시오.

  ```
  dltmqm RDQM_name
  ```

관련 정보

dltmqm

V 9.0.4 Linux *RDQM*의 선호 위치 설정

복제 데이터 큐 관리자(RDQM)의 선호 위치로 노드가 사용 가능할 경우 RDQM을 실행해야 하는 노드를 식별할 수 있습니다.

이 태스크 정보

선호 위치란 HA 그룹이 정상 상태(모든 노드와 연결이 사용 가능함)일 때 Pacemaker가 큐 관리자를 실행해야 할 노드의 이름입니다. 선호 위치는 큐 관리자가 작성될 때 기본 노드의 이름으로 초기화됩니다. 선호 위치 설정 명령은 세 노드 중 원하는 노드에서 실행할 수 있습니다. mqm 그룹과 `haclient` 그룹에 모두 속한 사용자여야 합니다.

프로시저

- 로컬 노드 또는 지정된 노드를 이름 지정된 큐 관리자의 선호 위치로 지정하려면 다음 명령을 입력하십시오.

  ```
  rdqmadm -p -m qmname [ -n nodename[,nodename  ]
  ```

  여기서 *qmname*은 선호 위치를 지정할 RDQM의 이름이고, *nodename*은 선호 노드의 이름입니다(선택사항).

  HA 그룹이 정상 상태이고 선호 위치가 현재 기본 노드가 아닐 경우, 큐 관리자가 중지되고 새 선호 위치에서 재시작됩니다. 두 개의 노드 이름으로 구성되는 쉼표로 구분되는 목록을 지정하여 두 번째로 선호하는 선호 위치를 지정할 수 있습니다.
- 큐 관리자가 복원될 때 자동으로 노드로 돌아가지 않도록 선호 위치를 지우려면 다음 명령을 입력하십시오.

  ```
  rdqmadm -p -m qmname -d
  ```

관련 정보

rdqmadm(복제된 데이터 큐 관리자 클러스터 관리)

V 9.0.4 Linux 유동 *IP* 주소 작성 및 삭제

유동 IP 주소를 사용하면 복제 데이터 큐 관리자(RDQM)가 실행 중인 HA 그룹의 노드에 관계없이, 클라이언트가 RDQM에 대해 동일한 IP 주소를 사용할 수 있습니다.

이 태스크 정보

rdqmint 명령을 사용하여 유동 IP 주소를 작성하거나 삭제할 수 있습니다. 부동 주소는 RDQM의 기본 노드에 있는 이름 지정된 물리적 인터페이스에 바인드됩니다. RDQM이 장애 복구되어 다른 노드가 기본 노드가 된 경우

에는 새 기본 노드에 있는 동일한 이름의 인터페이스에 유동 IP가 바인드됩니다. 세 노드의 물리적 인터페이스는 유동 IP 주소와 동일한 서브넷에 속해 있어야 합니다. 다음 다이어그램은 유동 IP 주소 사용을 보여줍니다.

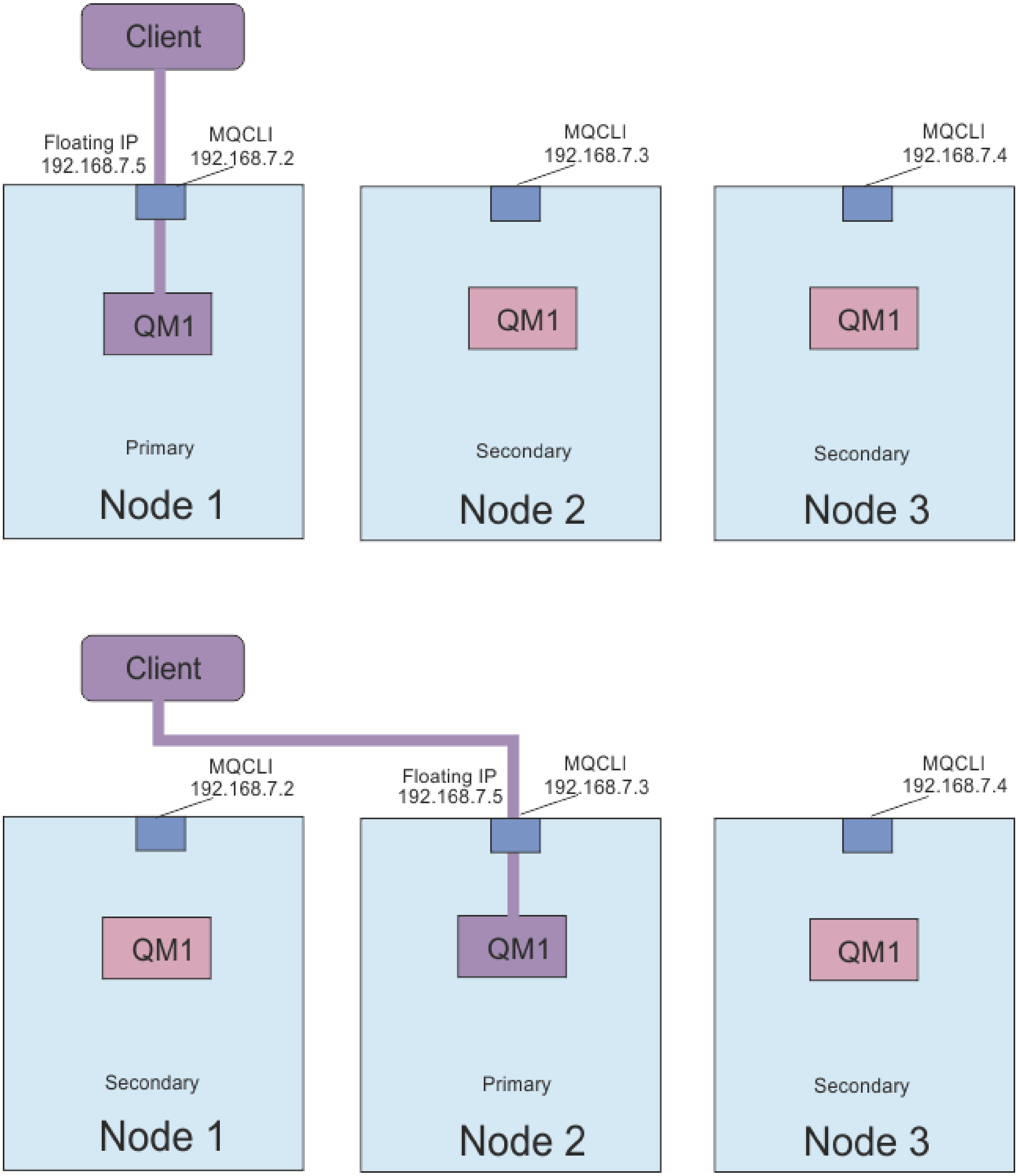


그림 *82. 부동 IP 주소*

rdqmint 명령을 실행하려면 mqm 및 haclient 그룹의 사용자여야 합니다. 유동 IP 주소는 RDQM의 기본 노드 또는 보조 노드에서 작성하거나 삭제할 수 있습니다.

참고: 여러 RDQM에 대해 동일한 유동 IP 주소를 사용할 수 없습니다. 각 RDQM의 유동 IP 주소는 고유해야 합니다.

프로시저

- RDQM의 유동 IP 주소를 작성하려면 다음 명령을 입력하십시오.

  ```
  rdqmint -m qmname -a  -f ipv4address -l interfacename
  ```

  설명:

  qmname
  유동 IP 주소를 작성할 RDQM의 이름입니다.

  ipv4address
  ipv4 형식의 유동 IP 주소입니다.

  유동 IP 주소는 이미 어플라이언스에 정의되지 않은 올바른 IPv4 주소여야 하며 로컬 인터페이스에 대해 정의된 정적 IP 주소와 동일한 서브넷에 속해야 합니다.

  interfacename
  바인드할 기본 노드의 물리적 인터페이스 이름입니다.

  예를 들면, 다음과 같습니다.

  ```
  rdqmint -m QM1 -a -f 192.168.7.5 -l MQCLI
  ```

- 기존 유동 IP 주소를 삭제하려면 다음 명령을 입력하십시오.

  ```
  rdqmint -m qmname -d
  ```

관련 정보

rdqmint(RDQM의 부동 IP 주소 추가 또는 삭제)

V 9.0.4 Linux *HA RDQM* 시작, 중지 및 상태 표시

변형된 형태의 표준 IBM MQ 제어 명령을 사용하여 복제 데이터 큐 관리자(RDQM)의 현재 상태를 시작, 중지 및 확인할 수 있습니다.

이 태스크 정보

mqm 그룹과 `haclient` 그룹 모두에 속해 있는 사용자로 복제 데이터 큐 관리자(RDQM)의 현재 상태를 시작, 중지 및 확인하는 명령을 실행해야 합니다.

큐 관리자를 시작하고 중지하는 명령은 해당 큐 관리자의 기본 노드에서 실행해야 합니다.

프로시저

- RDQM을 시작하려면 RDQM의 기본 노드에서 다음 명령을 입력하십시오.

  ```
  strmqm qmname
  ```

  여기서 *qmname*은 시작할 RDQM의 이름입니다.

  RDQM이 시작되고 Pacemaker가 RDQM 관리를 시작합니다. 다른 `strmqm` 옵션을 지정하려는 경우에는 `strmqm`과 함께 `-ns` 옵션을 지정해야 합니다.

- RDQM을 중지하려면 RDQM의 기본 노드에서 다음 명령을 입력하십시오.

  ```
  endmqm qmname
  ```

  여기서 *qmname*은 중지할 RDQM의 이름입니다.

  Pacemaker가 RDQM 관리를 중단하고 RDQM이 종료됩니다. RDQM을 중지할 때 다른 모든 **endmqm** 매개변수를 사용할 수 있습니다.

- RDQM의 상태를 확인하려면 다음 명령을 입력하십시오.

  ```
  dspmq
  ```

출력되는 상태 정보는 명령을 RDQM의 기본 노드 또는 보조 노드에서 실행할지 여부에 따라 달라집니다. 기본 노드에서 실행할 경우 **dspmq**가 리턴하는 정상 상태 메시지 중 하나가 표시됩니다. 보조 노드에서 명령을 실행할 경우 `running elsewhere` 상태가 표시됩니다. 예를 들어 **dspmq**가 RDQM7 노드에서 실행될 경우 다음과 같은 정보가 리턴될 수 있습니다.

```
QMNAME(RDQM8)                                       STATUS(Running elsewhere)
QMNAME(RDQM9)                                       STATUS(Running elsewhere)
QMNAME(RDQM7)                                       STATUS(Running)
```

기본 노드를 사용할 수 없거나 `haclient` 그룹의 구성원 또는 `root`가 아닌 사용자가 **dspmq**를 실행할 경우 `Unavailable` 상태가 보고됩니다. 예를 들면, 다음과 같습니다.

```
QMNAME(RDQM8)                   STATUS(Unavailable)
QMNAME(RDQM9)                   STATUS(Unavailable)
QMNAME(RDQM7)                   STATUS(Unavailable)
```

노드에 알려진 큐 관리자 목록 및 큐 관리자가 RDQM인지 여부를 표시하려면 **dspmq -o ha**(또는 **dspmq -o HA**) 명령을 입력하십시오. 예를 들면 다음과 같습니다.

```
dspmq -o ha

QMNAME(RDQM8)                                                 HA(Replicated)
QMNAME(RDQM9)                                                 HA(Replicated)
QMNAME(RDQM7)                                                 HA(Replicated)
QMNAME(QM7)                                                   HA()
```

관련 정보

dspmq(큐 관리자 표시)
endmqm(큐 관리자 종료)
strmqm(큐 관리자 시작)

V 9.0.4 Linux *RDQM* 및 *HA* 그룹 상태 보기

HA 그룹 및 개별 복제 데이터 큐 관리자(RDQM)의 상태를 볼 수 있습니다.

이 태스크 정보

rdqmstatus 명령을 사용하면 개별 RDQM 및 HA 그룹의 상태를 전체적으로 볼 수 있습니다.

rdqmstatus 명령을 실행하려면 mqm 및 `haclient` 그룹의 사용자여야 합니다. 이 명령은 세 노드 중 원하는 노드에서 실행할 수 있습니다.

프로시저

- HA 구성에 포함된 RDQM 및 노드의 상태를 보려면 다음을 수행하십시오.

  ```
  rdqmstatus
  ```

  명령을 실행한 노드의 ID 및 HA 구성에 포함된 RDQM의 상태가 다음과 같이 표시됩니다.

  ```
  Node:                                    mqhavm07.exampleco.com

  Queue manager name:                      RDQM8
  Queue manager status:                    Running elsewhere
  HA current location:                     mqhavm08.exampleco.com

  Queue manager name:                      RDQM9
  Queue manager status:                    Running elsewhere
  HA current location:                     mqhavm09.exampleco.com

  Queue manager name:                      RDQM7
  Queue manager status:                    Running
  HA current location:                     This node
  ```

- HA 그룹에 있는 세 노드의 상태를 보려면 다음 명령을 입력하십시오.

```
rdqmstatus -n
```

각 노드의 온라인 또는 오프라인 상태가 보고됩니다. 예를 들면, 다음과 같습니다.

```
Node mqha04(mqhavm04.example.com) is online
Node mqha05(mqhavm05.example.com) is offline
Node mqha06(mqhavm06.example.com) is online
```

- HA 그룹에 있는 모든 노드에서 특정 큐 관리자의 상태를 보려면 다음 명령을 입력하십시오.

```
rdqmstatus -m qmname
```

여기서 *qmname*은 상태를 보려는 RDQM의 이름입니다. 현재 노드의 RDQM 상태가 표시되고 그 뒤에 현재 노드의 관점에서 다른 두 노드의 상태에 대한 요약 정보가 표시됩니다.

다음 표는 RDQM에 대한 **rdqmstatus** 명령으로 리턴할 수 있는 현재 노드에 대한 정보를 요약하여 보여줍니다.

표 *30.* 현재 노드 상태

상태 속성	가능한 값	표시 시기
노드 이름	*nodename*	항상 표시됨
큐 관리자 상태	실행 중 다른 위치에서 실행 중 종료됨 사용 불가능	항상 표시됨
CPU	*n.nn*%	현재 노드가 기본 역할을 사용하는 경우(RDQM이 이 노드에서 실행 중임)에만 표시됩니다.
메모리	*nnn*MB 사용됨, *y.y*GB 할당됨	현재 노드가 기본 역할을 사용하는 경우(RDQM이 이 노드에서 실행 중임)에만 표시됩니다.
큐 관리자 파일 시스템	*nnn*MB 사용됨, *y.y*GB 할당됨 [*z*%]	현재 노드가 기본 역할을 사용하는 경우(RDQM이 이 노드에서 실행 중임)에만 표시됩니다.
HA 역할	1차 2차 알 수 없음	항상 표시됨
HA 상태	모든 노드가 대기 중임 이 노드가 대기 중임 원격 노드가 대기 중임 혼합됨 원격 노드의 상태	모든 노드가 대기 중임 현재 노드가 대기 중임 두 원격 노드가 대기 중임 원격 노드마다 서로 다른 상태임(개별 상태는 다음 표 참조) 두 원격 노드의 상태가 동일함(모든 값을 보려면 다음 표 참조)
HA 제어	사용 사용 불능 알 수 없음	항상 표시됨. RDQM이 Pacemaker의 제어를 받는지 여부를 표시합니다.
HA 선호 위치	없음 이 노드 알 수 없음 *nodename*	항상 표시됨

표 30. 현재 노드 상태 (계속)		
상태 속성	**가능한 값**	**표시 시기**
HA 유동 IP 인터페이스	*Interface_name*	항상 표시됨
HA 유동 IP 주소	*IPV4_address*	항상 표시됨

다음 표는 HA 그룹의 다른 노드에 대한 **rdqmstatus** 명령으로 리턴되는 정보를 요약하여 보여줍니다.

표 31. 기타 노드 상태		
상태 속성	**가능한 값**	**표시 시기**
노드 이름	`nodename`	항상 표시됨
HA 상태	정상 동기화 진행 중 원격 사용 불가능 불일치 일시정지 원격 노드가 대기 상태임 알 수 없음	노드가 서로 동기화됨 원격 노드와 동기화 중 원격 노드와 통신할 수 없음 원격 노드와 동기화되지 않았으며 동기화 중이지 않음 복제가 일시정지됨 원격 노드가 대기 상태임
HA 동기화 진행 중	*n.n*%	동기화가 진행 중이고 명령이 `root`로 실행될 때 표시됨
HA 예상 동기화 시간	*yyyy-mm-dd hh:mm:ss.nnn*	동기화가 진행 중일 때 표시됨
HA 비동기 데이터	*n*KB	원격 노드가 사용 불가능하거나 일치하지 않을 때 표시됨

예

예: 기본 노드의 정상 상태

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                    Running
CPU:                                     0.00
Memory:                                  123MB
Queue manager file system:               606MB used, 1.0GB allocated [60%]
HA role:                                 Primary
HA status:                               Normal
HA control:                              Enabled
HA current location:                     This node
HA preferred location:                   This node
HA floating IP interface:                Eth4
HA floating IP address:                  192.0.2.4


Node:                                    mqhavm08.exampleco.com
HA status:                               Normal


Node:                                    mqhavm09.exampleco.com
HA status:                               Normal
```

예: 보조 노드의 정상 상태

```
Node:                                    mqhavm08.exampleco.com
Queue manager status:                    Running elsewhere
HA role:                                 Secondary
HA status:                               Normal
HA control:                              Enabled
HA current location:                     mqhavm07.exampleco.com
HA preferred location:                   mqhavm07.exampleco.com
HA floating IP interface:                Eth4
HA floating IP address:                  192.0.2.4
```

```
Node:                                   mqhavm07.exampleco.com
HA status:                              Normal

Node:                                   mqhavm09.exampleco.com
HA status:                              Normal
```

예: 동기화가 진행 중일 때 기본 노드의 상태

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.53
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Synchronization in progress
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA floating IP interface:               Eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm08.exampleco.com
HA status:                              Synchronization in progress
HA synchronization progress:            11.0%
HA estimated time to completion:        2017-09-06 14:55:05

Node:                                   mqhavm09.exampleco.com
HA status:                              Synchronization in progress
HA synchronization progress:            11.0%
HA estimated time to completion:        2017-09-06 14:55:06
```

여러 상태를 표시하는 기본 노드 예

```
Node:                                   mqhavm07.exampleco.com
Queue manager status:                   Running
CPU:                                    0.02
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
HA role:                                Primary
HA status:                              Mixed
HA control:                             Enabled
HA current location:                    This node
HA preferred location:                  This node
HA floating IP interface:               Eth4
HA floating IP address:                 192.0.2.4

Node:                                   mqhavm08.exampleco.com
HA status:                              Normal

Node:                                   mqhavm09.exampleco.com
HA status:                              Inconsistent
```

관련 정보

Linux rdqmstatus

V 9.0.4 Linux 실패한 노드 바꾸기

HA 그룹의 노드 중 하나가 실패할 경우 해당 노드를 바꿀 수 있습니다.

이 태스크 정보

노드 바꾸기를 위해 수행할 단계는 상황에 따라 다릅니다.

- 실패한 노드를 동일한 구성의 노드로 바꾸는 경우 HA 그룹을 중단하지 않고 노드를 바꿀 수 있습니다.
- 새 노드가 다른 구성을 사용할 경우에는 HA 그룹을 삭제한 후 다시 빌드해야 합니다.

프로시저

- 대체 노드가 실패한 노드와 유사하게 구성된 경우(동일한 호스트 이름, 동일한 IP 주소 등), 새 노드에서 다음 단계를 완료하십시오.

a) 다른 노드의 파일과 일치하는 rdqm.ini 파일을 작성한 후 rdqmadm -c 명령을 실행하십시오 (454 페이지의 『Pacemaker 클러스터(HA 그룹) 정의』 참조).

b) crtmqm -sxs *qmanager* 명령을 실행하여 복제 데이터 큐 관리자를 각각 다시 작성하십시오(456 페이지의 『HA RDQM 작성』 참조).

- 대체 노드가 실패한 노드와 다른 구성을 사용할 경우 다음 단계를 수행하십시오.

a) **dltmqm** 명령을 사용하여 HA 그룹에 있는 다른 노드에서 복제 데이터 큐 관리자를 삭제하십시오(457 페이지의 『HA RDQM 삭제』).

b) **rdqmadm -u** 명령을 사용하여 Pacemaker 클러스터를 구성 해제하십시오 (455 페이지의 『Pacemaker 클러스터(HA 그룹) 삭제』 참조).

c) **rdqmadm -c** 명령을 사용하여 새 노드에 대한 정보를 포함하여 Pacemaker 클러스터를 다시 구성하십시오(454 페이지의 『Pacemaker 클러스터(HA 그룹) 정의』).

d) crtmqm -sxs *qmanager* 명령을 실행하여 복제 데이터 큐 관리자를 각각 다시 작성하십시오(456 페이지의 『HA RDQM 작성』 참조).

V 9.0.4 스플릿 브레인 상황의 해결

HA 그룹의 특정 실패 시퀀스가 스플릿 브레인 상황을 야기하여 보고될 수 있습니다.

예를 들어 세 개의 노드 모두가 연결이 손실되었습니다. 기본 노드에 앞서 두 보조 노드가 먼저 연결을 재획득하면 새 쿼럼을 형성하고 둘 중 한 노드가 큐 관리자를 실행합니다. 원래 기본 노드가 연결을 되찾으면 스플릿 브레인 상황이 보고될 수 있습니다.

이 상황에서 원래 기본 노드에 rdqmstatus -m *QMname*을 실행하면 HA 상태가 Inconsistent로 표시됩니다.

```
Node:                                   node1
Queue manager status:                   Running elsewhere
HA role:                                Secondary
HA status:                              Inconsistent
HA control:                             Enabled
HA current location:                    hanode2
HA preferred location:                  This node
HA floating IP interface:               None
HA floating IP address:                 None

Node:                                   node2
HA status:                              Inconsistent
HA out of sync data:                    8KB

Node:                                   node3
HA status:                              Inconsistent
HA out of sync data:                    8KB
```

이 인스턴스에서 원래 보조 노드(새 쿼럼을 형성한)에서 데이터를 보유해야 합니다. 다음 단계를 완료하십시오.

1. 원래 기본 노드에서 루트로 다음 명령을 실행하십시오.

```
drbdadm connect --discard-my-data QMname
```

2. 각 보조 노드에서 루트로 다음 명령을 실행하십시오.

```
drbdadm connect QMname:first-node-name
```

drbdadm은 DRBD에서 제공되는 명령입니다. 이는 /usr/sbin/drbdadm에 drbd-utils 패키지의 부분으로 설치됩니다.

V 9.0.5 Linux RDQM 재해 복구

복제 데이터 큐 관리자(RDQM)는 Linux 플랫폼의 서브세트에서 사용 가능하며 재해 복구 솔루션을 제공할 수 있습니다.

자세한 내용은 소프트웨어 제품 호환성 보고서를 참조하십시오.

한 서버에서 실행되는 재해 복구 큐 관리자의 기본 인스턴스를 작성하고 복구 노드로 작동하는 다른 서버에서 큐 관리자의 보조 인스턴스를 작성할 수 있습니다. 데이터는 큐 관리자 인스턴스 간에 복제됩니다. 기본 큐 관리자가 손실된 경우 수동으로 보조 인스턴스를 기본 인스턴스로 만들고 큐 관리자를 시작한 다음 동일한 위치에서 작업을 재개할 수 있습니다. 큐 관리자가 보조 역할에 있는 동안에는 시작할 수 없습니다. 두 노드 간의 데이터 복제는 DRBD가 처리합니다.

기본 및 보조 큐 관리자 사이의 동기 및 비동기 데이터 복제를 선택할 수 있습니다. 비동기 옵션을 선택하면 IBM MQ PUT 또는 GET과 같은 옵션이 완료되고 이벤트가 보조 큐 관리자로 복제되게 전에 애플리케이션에 리턴됩니다. 비동기 복제는 복구 상황 후에 일부 메시지 데이터가 유실될 수 있음을 의미합니다. 그러나 보조 큐 관리자는 일관된 상태에 있으며 메시지 스트림의 약간 더 빠른 파트에서 시작되는 경우에도 즉시 실행을 시작할 수 있습니다.

기존 큐 관리자에 재해 복구를 추가할 수 없으며 큐 관리자는 RDQM 재배 복구 및 RDQM 고가용성 둘 다와 함께 구성될 수 없습니다.

다수의 여러 서버에서 실행되는 여러 쌍의 RDQM 큐 관리자를 가질 수 있습니다. 예를 들어 동일한 노드에서 실행되는 6개의 기본 DR 큐 관리자를 보유하고 해당 보조 큐 관리자는 6개의 다른 데이터 센터에 있는 6개의 개별 노드에서 구성할 수 있습니다. 마찬가지로 여러 노드에서 실행되는 기본 재해 복구 큐 관리자를 보유하고 해당하는 모든 보조 재해 복구 큐 관리자를 동일한 노드에서 실행할 수 있습니다. 일부 예제 구성이 다음 다이어그램에 설명되어 있습니다.

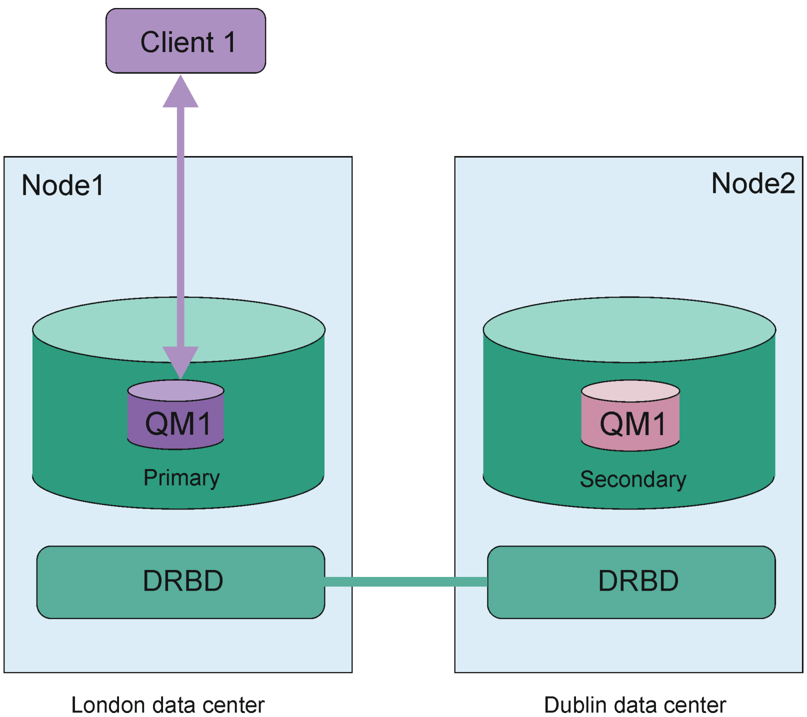


그림 *83.* 단일 *RDQM* 쌍

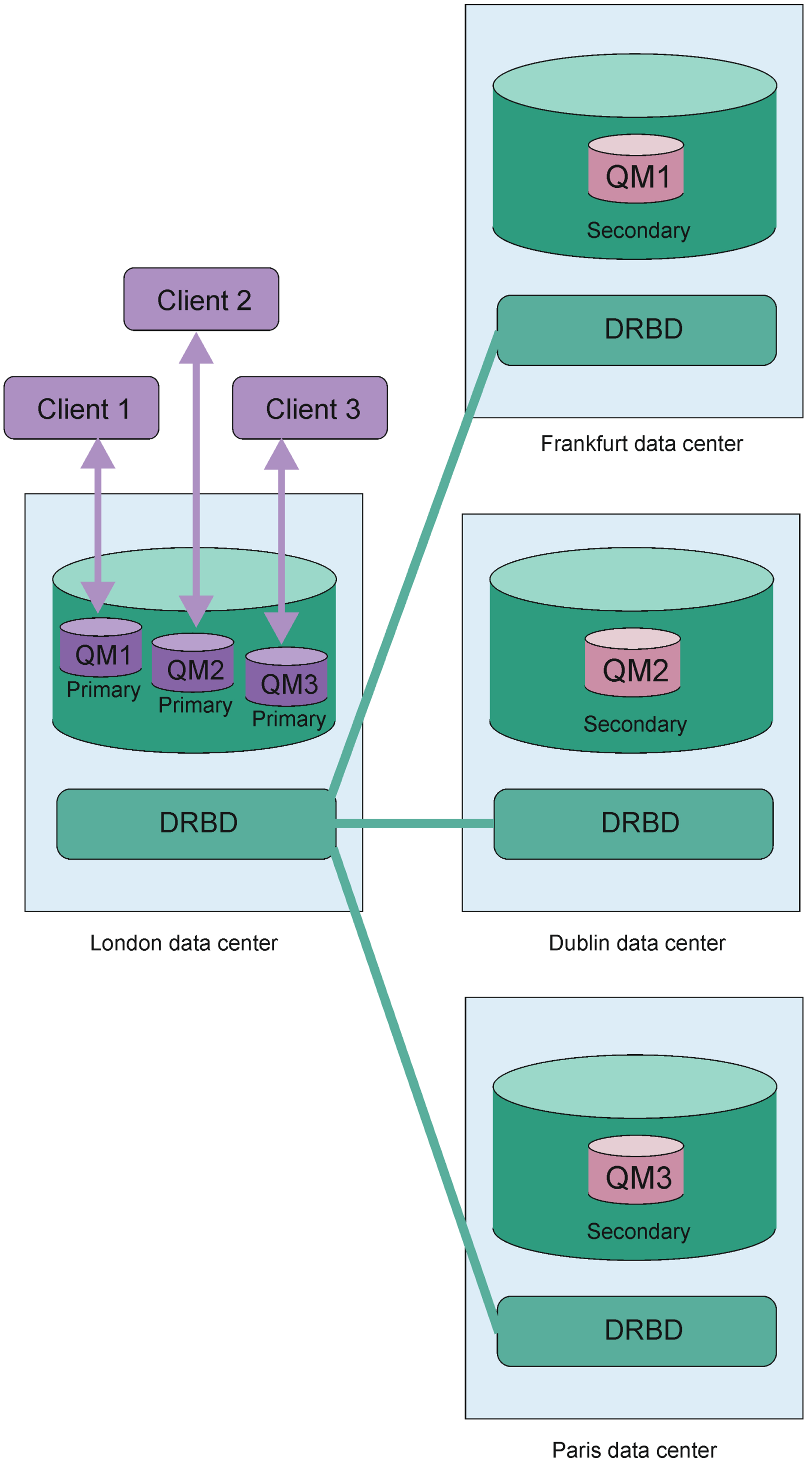


그림 *84.* 동일한 노드의 기본 큐 관리자

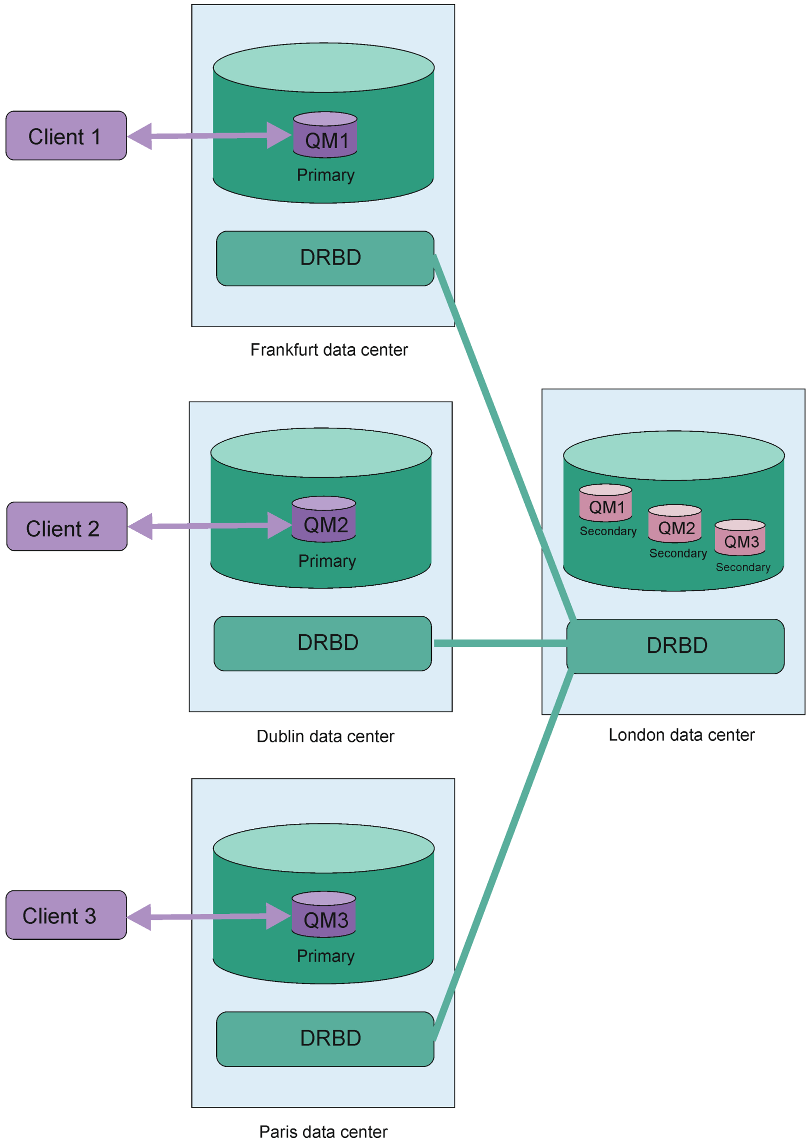


그림 *85.* 동일한 노드의 보조 큐 관리자

복제, 동기화 및 스냅샷

재해 복구 구성의 두 노드가 연결된 동안 재해 복구 큐 관리자를 위한 지속적 데이터의 업데이트는 큐 관리자의 기본 인스턴스에서 보조 인스턴스로 전송됩니다. 이것을 **복제**라고 합니다.

두 노드 사이의 네트워크 연결이 유실되면 큐 관리자의 기본 인스턴스를 위한 지속적 데이터에 대한 변경이 추적됩니다. 네트워크 연결이 복원될 때 서로 다른 프로세스가 보조 인스턴스를 가능한 빨리 최대 속도로 가져오는 데 사용됩니다. 이것을 **동기화**라고 합니다.

동기화가 진행 중인 동안 보조 인스턴스의 데이터는 불일치 상태에 있습니다. 보조 큐 관리자 데이터 상태의 **스냅샷**이 작성됩니다. 동기화 동안 기본 노드 또는 네트워크 연결 실패가 발생하면 보조 인스턴스가 이 스냅샷으로 되돌아가므로 큐 관리자가 시작될 수 있습니다. 그러나 원래 네트워크 장애 이후에 발생한 모든 업데이트는 유실됩니다.

V 9.0.5 Linux *RDQM DR* 솔루션에 대한 요구사항

RDQM 재해 복구(DR) 큐 관리자를 쌍을 구성하려면 여러 요구사항을 만족시켜야 합니다.

시스템 요구사항

RDQM DR을 구성하기 전에 RDQM DR 큐 관리자를 호스팅할 서버 각각에서 몇 가지 구성을 완료해야 합니다.

- 각 노드에는 drbdpool이라는 볼륨 그룹이 필요합니다. 각 재해 복구 복제 데이터 큐 관리자(DR RDQM)를 위한 스토리지는 이 볼륨 그룹의 큐 관리자마다 별도의 두 논리적 볼륨으로 할당됩니다. (각 큐 관리자는 스냅샷으로 되돌리기 조작을 지원하기 위해 두 개의 논리적 볼륨이 필요하며 따라서 사용자가 작성할 때 지정한 스토리지보다 두 배가 넘는 스토리지가 각 DR RDQM에 할당됩니다.) 최상의 성능을 위해 이 볼륨 그룹은 하나 이상의 물리적 볼륨으로 구성되어야 합니다. 이때 각 물리적 볼륨은 내부 디스크 드라이브(SSD 선호)에 해당합니다.
- 각 노드에는 데이터 복제에 사용되는 인터페이스가 필요합니다. 모든 복제 데이터 큐 관리자의 예상 워크로드를 감안할 경우 이 인터페이스가 복제 요구사항을 지원할 수 있을 정도의 충분한 대역폭을 제공해야 합니다.

  최대 결함 허용을 위해 이 인터페이스는 독립된 네트워크 인터페이스 카드(NIC)여야 합니다.
- DRBD에는 RFC 1123에 의해 수정된 RFC 952에 의해 정의된 대로 RDQM에 사용되는 각 노드에 유효한 인터넷 호스트 이름 (uname -n에 의해 리턴되는 값) 이 있어야 합니다.
- DR RDQM에 사용된 노드 사이에 방화벽이 있으면 방화벽이 복제에 사용된 포트에서 노드 사이에 트래픽을 허용해야 합니다.
- 시스템이 허용 모드가 아닌 모드에서 SELinux를 사용하는 경우 다음 명령을 실행해야 합니다.

```
semanage permissive -a drbd_t
```

네트워크 요구사항

다른 데이터 센터에서 재해 복구에 사용될 노드를 배치하는 것이 좋습니다.

다음 제한사항을 알고 있어야 합니다.

- 데이터 센터 사이의 대기 시간이 늘어나면 성능이 급격하게 저하됩니다. IBM은 동기 복제에 최대 5밀리초의 대기 시간을, 비동기 복제에 50밀리초의 대기 시간을 지원합니다.
- 복제 링크를 통해 전송된 데이터는 IBM MQ AMS를 사용하여 적용되는 것 이외의 추가 암호화가 적용되지 않습니다.
- 재해 복구를 위해 RDQM 큐 관리자를 구성하면 두 개의 RDQM 노드 사이에 데이터 복제를 위한 요구사항으로 인해 오버헤드가 발생합니다. 동기 복제는 비동기 복제보다 더 큰 오버헤드가 발생합니다. 동기 복제가 사용되는 경우 데이터가 두 노드에 작성될 때까지 디스크 입출력 조작이 차단됩니다. 비동기 복제를 사용하는 경우 데이터를 기본 노드에만 작성해야 처리를 계속할 수 있습니다.

큐 관리자 작업에 필요한 사용자 요구사항

복제 데이터 큐 관리자(RDQM)를 작성, 삭제 또는 구성하기 위해서는 사용자가 루트 사용자이거나 다음 명령에 대한 sudo 권한이 부여된 mqm 그룹에 속한 사용자 ID를 가지고 있어야 합니다.

- **crtmqm**
- **dltmqm**
- **rdqmdr**

mqm 그룹에 속한 사용자는 다음 명령을 사용하여 DR RDQM의 상태를 볼 수 있습니다.

- **dspmq**
- **rdqmstatus**

V 9.0.5 Linux 재해 복구 *RDQM* 작성

crtmqm 명령을 사용하여 재해 복구 구성에서 기본 또는 보조로 작동할 복제 데이터 큐 관리자(RDQM)를 작성합니다.

이 태스크 정보

사용자가 sudo를 사용할 수 있는 경우 mqm 그룹의 사용자로 복제 데이터 큐 관리자(RDQM)를 작성할 수 있습니다. 그렇지 않으면 루트로 RDQM을 작성해야 합니다.

한 노드에서 기본 RDQM DR 큐 관리자를 작성해야 합니다. 그런 다음 다른 노드에서 동일한 큐 관리자의 보조 인스턴스를 작성해야 합니다. 기본 및 보조 인스턴스는 이름이 동일해야 하고 동일한 스토리지 용량이 할당되어야 합니다.

프로시저

- 기본 DR RDQM을 작성하려면 다음을 수행하십시오.
  a) 다음 명령을 입력하십시오.

```
crtmqm -rr p [-rt (a | s)] -rl Local_IP -ri Recovery_IP -rn Recovery_Name -rp Port
[other_crtmqm_options] [-fs size] QMname
```

설명:

-rr p
사용자가 큐 관리자의 기본 인스턴스를 작성하고 있음을 지정합니다.

-rt a | s
-rt s는 DR 구성에서 동기 복제를 사용함을 지정하고 **-rt a** 는 DR 구성에서 비동기 복제를 사용함을 지정합니다. 비동기 복제가 기본값입니다.

-rl *Local_IP*
이 큐 관리자의 DR 복제에 사용되는 로컬 IP 주소를 지정합니다.

-ri *Recovery_IP*
큐 관리자의 보조 인스턴스를 호스팅하는 서버에서 복제에 사용되는 인터페이스의 IP 주소를 지정합니다.

-rn *Recovery_Name*
큐 관리자의 보조 인스턴스를 호스팅하는 시스템의 이름을 지정합니다. 이 이름은 사용자가 해당 서버에서 `uname -n`을 실행하면 리턴되는 값입니다. 해당 서버에서 보조 큐 관리자를 명시적으로 작성해야 합니다.

-rp *Port*
DR 복제에 사용할 포트를 지정합니다.

other_crtmqm_options
이러한 일반 **crtmqm** 옵션 중 하나 이상을 선택적으로 지정할 수 있습니다.

- -z
- -q
- -c *Text*
- -d *DefaultTransmissionQueue*
- -h *MaxHandles*
- -g *ApplicationGroup*
- -oa *user|group*
- -t *TrigInt*
- -u *DeadQ*
- -x *MaxUMsgs*
- -lp *LogPri*
- -ls *LogSec*
- -lc | -l
- -lla | -lln
- -lf *LogFileSize*
- -p *Port*

-fs ***size***
: 선택적으로 큐 관리자를 위해 작성할 파일 시스템의 크기 즉, drbdpool 볼륨 그룹에 작성된 논리적 볼륨의 크기를 지정합니다. 스냅샷으로 되돌리기 조작을 지원하기 위해 해당 크기의 다른 논리적 볼륨이 작성되므로 DR ROM에 대한 총 스토리지 크기는 여기에 지정된 것의 두 배가 넘게 됩니다.

QMname
: 복제 데이터 큐 관리자의 이름을 지정합니다. 이름은 대소문자가 구분됩니다.

명령을 완료한 후 큐 관리자의 보조 인스턴스를 작성하기 위해 보조 노드에서 입력해야 하는 명령을 출력합니다. 기본 노드에서 **rdqmdr** 명령을 사용하여 보조 큐 관리자를 작성하기 위해 보조 노드에서 실행해야 하는 **crtmqm** 명령을 검색할 수도 있습니다 (471 페이지의 『DR RDQM의 기본 및 보조 특성 관리』 참조).

- 보조 DR RDQM을 작성하려면 다음을 수행하십시오.
  a) RDQM의 보조 인스턴스를 호스팅할 노드에서 다음 명령을 입력하십시오.

```
crtmqm  -rr s [-rt (a | s)] -rl Local_IP -ri Primary_IP -rn Primary_Name  -rp Port
[other_crtmqm_options] [-fs size] QMname
```

여기서:

-rr s
: 사용자가 큐 관리자의 보조 인스턴스를 작성하고 있음을 지정합니다.

-rt a | s
: **-rt s** 는 DR 구성이 동기 복제를 사용하도록 지정합니다. **-rt a** 은 DR 구성이 비동기 복제를 사용하도록 지정합니다.

-rl ***Local_IP***
: 이 큐 관리자의 DR 복제에 사용되는 로컬 IP 주소를 지정합니다.

-ri ***Primary_IP***
: 큐 관리자의 기본 인스턴스를 호스팅하는 서버에서 복제에 사용되는 인터페이스의 IP 주소를 지정합니다.

-rn ***Primary_Name***
: 큐 관리자의 기본 인스턴스를 호스팅하는 시스템의 이름을 지정합니다. 이 이름은 사용자가 해당 서버에서 uname -n을 실행하면 리턴되는 값입니다.

-rp ***Port***
: DR 복제에 사용할 포트를 지정합니다.

other_crtmqm_options
: 이러한 일반 **crtmqm** 옵션 중 하나 이상을 선택적으로 지정할 수 있습니다.

  - -z

-fs ***size***
: 큐 관리자를 위해 작성할 파일 시스템의 크기 즉, drbdpool 볼륨 그룹에 작성된 논리적 볼륨의 크기를 지정합니다. 기본 큐 관리자를 작성할 때 기본이 아닌 크기를 지정한 경우 여기에 동일한 값을 지정해야 합니다.

QMname
: 복제 데이터 큐 관리자의 이름을 지정합니다. 이름은 큐 관리자의 기본 인스턴스에 지정된 이름과 동일해야 합니다. 이름은 대소문자를 구분합니다.

다음에 수행할 작업

큐 관리자의 기본 및 보조 인스턴스를 작성한 후, 두 노드 모두에서 상태를 확인하여 올바른지 검사해야 합니다. 두 노드에서 **rdqmstatus** 명령을 사용하십시오. 노드는 472 페이지의 『DR RDQM 상태 보기』에 설명된 대로 정상 상태를 표시해야 합니다. 이 상태를 표시하지 않는 경우 보조 인스턴스를 작성하고 재작성하십시오. 이 때, 올바른 인수를 사용해야 합니다.

관련 정보

crtmqm

V 9.0.5 Linux *DR RDQM* 삭제

dltmqm 명령을 사용하여 재해 복구 복제 데이터 큐 관리자(RDQM)를 삭제할 수 있습니다.

이 태스크 정보

RDQM 삭제 명령은 RDQM의 기본 노드 및 보조 노드 둘 다에서 실행해야 합니다. 먼저 RDQM을 종료해야 합니다. mqm 사용자가 필요한 sudo 권한을 보유하고 있는 경우 해당 사용자로 명령을 실행할 수 있습니다. 그렇지 않은 경우 스크립트를 루트로 실행해야 합니다.

프로시저

- DR RDQM을 삭제하려면 다음 명령을 입력하십시오.

```
dltmqm RDQM_name
```

관련 정보

dltmqm

V 9.0.5 Linux *DR RDQM*의 기본 및 보조 특성 관리

보조 재해 복구 복제 데이터 큐 관리자(DR RDQM)를 기본 DR RDQM으로 변경할 수 있습니다. 또한 기본 인스턴스를 보조 인스턴스로 변경할 수 있습니다.

이 태스크 정보

RDQM의 보조 인스턴스를 기본 인스턴스로 바꾸려면 **rdqmdr** 명령을 사용합니다. 어떤 이유로 기본 인스턴스를 유실한 경우 이 조치를 수행해야 할 수 있습니다. 그런 다음 큐 관리자를 시작하고 복구 노드에서 실행할 수 있습니다.

또한 **rdqmdr** 명령을 사용하여 RDQM의 기본 인스턴스를 보조 인스턴스로 변경할 수도 있습니다. 예를 들어 시스템을 재구성하는 경우 이 조치를 수행해야 합니다.

복구 노드에서 큐 관리자의 보조 인스턴스를 작성하기 위해 필요한 정확한 명령을 검색하기 위해 기본 큐 관리자에서 **rdqmdr**을 사용할 수도 있습니다.

사용자가 sudo를 사용할 수 있는 경우 mqm 그룹의 사용자로 **rdqmdr** 명령을 사용할 수 있습니다. 그렇지 않으면 루트로 로그인되어 있어야 합니다.

프로시저

- DR RDQM의 보조 인스턴스를 기본 인스턴스로 변경하려면 다음 명령을 입력하십시오.

  ```
  rdqmdr -m QMname -p
  ```

  큐 관리자의 1차 인스턴스가 아직 실행 중이고 DR 복제 링크가 아직 작동 중인 경우 이 명령은 실패합니다.
- 큐 관리자의 기본 인스턴스를 보조 인스턴스로 변경하려면 다음 명령을 입력하십시오.

  ```
  rdqmdr -m QMname -s
  ```

- 큐 관리자의 보조 인스턴스를 구성하기 위해 필요한 **crtmqm** 명령을 표시하려면 기본 노드에서 다음 명령을 입력하십시오.

  ```
  rdqmdr -d -m QMname
  ```

  RD RDQM의 보조 인스턴스를 작성하려면 리턴된 **crtmqm** 명령을 보조 노드에서 입력하십시오.

V 9.0.5 Linux *DR RDQM* 시작, 중지 및 상태 표시

표준 IBM MQ 제어 명령의 변형을 사용하여 재해 복구 복제 데이터 큐 관리자(DR RDQM)를 시작 및 중지하고 현재 상태를 확인할 수 있습니다.

이 태스크 정보

mqm 그룹에 속해 있는 사용자로 복제 데이터 큐 관리자(RDQM)를 시작 및 중지하고 현재 상태를 보는 명령을 실행해야 합니다.

큐 관리자를 시작 및 중지하는 명령은 해당 큐 관리자의 기본 노드(큐 관리자가 현재 실행 중인 노드)에서 실행해야 합니다.

프로시저

- DR RDQM을 시작하려면 RDQM의 기본 노드에서 다음 명령을 입력하십시오.

  ```
  strmqm qmname
  ```

  여기서 *qmname*은 시작할 RDQM의 이름입니다.
- RDQM을 중지하려면 RDQM의 기본 노드에서 다음 명령을 입력하십시오.

  ```
  endmqm qmname
  ```

  여기서 *qmname*은 중지할 RDQM의 이름입니다.
- RDQM의 상태를 확인하려면 다음 명령을 입력하십시오.

  ```
  dspmq -m QMname
  ```

  출력되는 상태 정보는 명령을 RDQM의 기본 노드 또는 보조 노드에서 실행할지 여부에 따라 달라집니다. 기본 노드에서 실행할 경우 **dspmq**가 리턴하는 정상 상태 메시지 중 하나가 표시됩니다. 보조 노드에서 명령을 실행하면 `Ended immediately` 상태가 표시됩니다. 예를 들어 **dspmq**가 RDQM7 노드에서 실행될 경우 다음과 같은 정보가 리턴될 수 있습니다.

  ```
  QMNAME(DRQM8)                                STATUS(Ended immediately)
  QMNAME(DRQM7)                                STATUS(Running)
  ```

  RDQM을 재해 복구를 위해 구성할지 및 현재 기본 인스턴스인지 또는 보조 인스턴스인지를 설정하기 위해 dspmq에 인수를 함께 사용할 수 있습니다.

  ```
  dspmq -m QMname -o (dr | DR)
  ```

  다음 응답 중 하나가 표시됩니다.

  DRROLE()
  큐 관리자가 재해 복구를 위해 구성되지 않았음을 표시합니다.

  DRROLE(Primary)
  큐 관리자가 DR 1차로 구성되었음을 표시합니다.

  DRROLE(Secondary)
  큐 관리자가 DR 2차로 구성되었음을 표시합니다.

관련 정보

dspmq

endmqm

strmqm

V 9.0.4 Linux *DR RDQM* 상태 보기

노드에 있는 모든 재해 복구 복제 데이터 큐 관리자(DR RDQM)의 상태 또는 지정된 DR RDQM에 대한 세부 정보를 볼 수 있습니다.

이 태스크 정보

모든 DR RDQM 또는 개별 RDQM의 상태를 보려면 **rdqmstatus** 명령을 사용하십시오.

rdqmstatus 명령을 실행하려면 mqm 그룹의 사용자이어야 합니다. DR RDQM 노드 쌍 중 하나에서 명령을 실행할 수 있습니다.

프로시저

- 노드에 있는 모든 DR RDQM의 상태를 보려면 해당 노드에서 다음 명령을 실행하십시오.

  ```
  rdqmstatus
  ```

  노드에 있는 DR RDQM의 상태가 표시됩니다. 예를 들면, 다음과 같습니다.

  ```
  Queue manager name:                     DRQM8
  Queue manager status:                   Ended immediately
  DR role:                                Secondary

  Queue manager name:                     DRQM7
  Queue manager status:                   Running
  DR role:                                Primary
  ```

- 특정 RDQM의 상태를 확인하려면 다음 명령을 입력하십시오.

  ```
  rdqmstatus -m  qmname
  ```

  다음 표는 리턴되는 정보를 요약합니다.

표 *32.* 상태 속성

상태 속성	가능한 값	표시 시기
큐 관리자 상태	상태(dspmq로 표시됨)	항상 표시됨
CPU	*n.nn*%	현재 노드의 RDQM에 기본 역할이 있는 경우에만 표시
메모리	*nnn*MB	현재 노드의 RDQM에 기본 역할이 있는 경우에만 표시
큐 관리자 파일 시스템	*nnn*MB 사용됨,*n.n*GB 할당됨 [*n*%]	현재 노드의 RDQM에 기본 역할이 있는 경우에만 표시
DR 역할	1차 보조 알 수 없음	항상 표시됨
DR 상태	정상	정상 작동
	동기화 진행 중	동기화가 진행 중입니다.
	파티션됨	DR 복제 네트워크를 사용할 수 없는 동안 큐 관리자가 두 노드에서 시작되었습니다.
	원격 시스템 사용 불가능	다른 노드에 대한 연결이 끊어졌습니다.
	불일치	동기화가 진행 중이었으나 인터럽트되었습니다.
	스냅샷으로 되돌리기	큐 관리자가 불일치 상태가 되었을 때 작성해 놓은 스냅샷으로 되돌리도록 선택했습니다.
	원격 시스템이 구성되지 않음	RDQM의 기본 인스턴스가 구성되었지만 보조 인스턴스가 구성되지 않았습니다.

표 32. 상태 속성 (계속)		
상태 속성	**가능한 값**	**표시 시기**
	협상 실패	노드 중 하나에 동기 복제가 설정되고 다른 노드에 비동기 복제가 설정되었습니다.
DR 유형	동기 또는 비동기	항상 표시됨
DR 포트	*port_number*(이 큐 관리자에 대한 데이터를 복제하는 데 사용되는 TCP/IP 포트)	항상 표시됨
DR 로컬 IP 주소	이 큐 관리자가 DR을 위해 복제할 로컬 IP 주소	항상 표시됨
DR 원격 IP 주소	이 큐 관리자가 DR을 위해 복제할 원격 IP 주소	항상 표시됨
DR 비동기 데이터	*n*KB	원격 노드가 사용 불가능하거나 일치하지 않을 때 표시됨
DR 동기화 진행상태	*n*%	동기화가 진행 중일 때 표시됨
DR 완료 예상 시간	*YYYY-MM-DD HH:MM:SS*	동기화가 진행 중일 때 표시됨
스냅샷 되돌리기 진행	*n*%	DR 상태가 `Reverting to snapshot`일 때 표시됩니다. 진행 상태를 거꾸로 세므로 0%가 완료를 나타냅니다.

예

예: 기본 노드의 정상 상태

```
Queue manager status:                   Running
CPU:                                    0.00
Memory:                                 123MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Normal
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

예: 보조 노드의 정상 상태

```
Queue manager status:                   Ended immediately
DR role:                                Secondary
DR status:                              Normal
DR port:                                3000
DR local IP address:                    192.168.20.2
DR remote IP address:                   192.168.20.1
```

예: 동기화가 진행 중일 때 기본 노드의 상태

```
Queue manager status:                   Running
CPU:                                    0.53
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Synchronization in progress
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

```
DR synchronization progress:            11.0%
DR estimated time to completion:        2017-09-06 14:55:05
```

예: 파티션된 기본 노드

```
Queue manager status:                   Running
CPU:                                    0.02
Memory:                                 124MB
Queue manager file system:              51MB used, 1.0GB allocated [5%]
DR role:                                Primary
DR status:                              Partitioned
DR type:                                Synchronous
DR port:                                3000
DR local IP address:                    192.168.20.1
DR remote IP address:                   192.168.20.2
```

관련 정보

Linux rdqmstatus

V 9.0.5 Linux 재해 복구 환경에서 조작

재해 복구 구성에서 보조 큐 관리자로 전환하고자 하는 여러 상황이 있을 수 있습니다.

재해 복구

기본 사이트에서 기본 큐 관리자가 완전히 손실된 후 복구 사이트에서 보조 큐 관리자를 시작합니다. 애플리케이션은 복구 사이트의 큐 관리자에 다시 연결하고 보조 큐 관리자가 애플리케이션 메시지를 처리합니다. 이전 구성으로 되돌리기 위해 따라야 하는 단계는 장애의 원인에 따라 다릅니다. 예를 들어 기본 노드의 완전한 손실과 일시적 손실이 있습니다.

기본 사이트의 일시적 손실에 따른 단계에 대해서는 475 페이지의 『복구 노드로 전환』의 내용을 참조하십시오. 영구적 실패에 따른 단계는 476 페이지의 『재해 복구 구성에서 실패한 노드 대체』의 내용을 참조하십시오.

재해 복구 테스트 지원

일시적으로 보조 인스턴스로 전환하고 애플리케이션이 성공적으로 연결될 수 있는지 확인하여 재해 복구 구성을 테스트할 수 있습니다. 기본 노드의 일시적 실패 후에 전환할 때와 동일한 프로시저를 따릅니다. 475 페이지의 『복구 노드로 전환』의 내용을 참조하십시오.

스냅샷으로 되돌리기

동기화가 진행 중인 동안 기본 노드에서 실패를 겪는 경우 동기화가 시작되기 바로 전 보조 큐 관리자 데이터의 스냅샷으로 되돌릴 수 있습니다. 그런 다음 보조를 일치 상태로 복원하고 기본으로 실행할 수 있습니다. 스냅샷으로 되돌리려면 475 페이지의 『복구 노드로 전환』에 설명한 대로 보조를 기본으로 만드십시오. 큐 관리자를 시작하기 전에 **rdqmstatus** 명령을 사용하여 스냅샷으로 되돌리기가 완료되었는지 확인하십시오.

V 9.0.5 Linux 복구 노드로 전환

기본 사이트에서 재해가 발생하는 경우, 복구 사이트로 전환하기 위한 단계를 수행합니다.

이 태스크 정보

기본 사이트에서 기본 큐 관리자가 손실된 후 복구 사이트에서 보조 큐 관리자를 기본 큐 관리자로 만들고 이를 시작합니다. 애플리케이션은 복구 사이트의 큐 관리자에 다시 연결하고 큐 관리자가 애플리케이션 메시지를 처리합니다. 또한 이 프로시저를 사용하여 복구 노드를 테스트할 수 있습니다.

루트로 로그인하거나 mqm 그룹에 속하고 필수 sudo 구성이 있는 사용자로 로그인해야 합니다.

프로시저

1. 이 프로시저를 사용하여 보조 큐 관리자를 테스트하는 경우(즉, 기본 인스턴스가 여전히 실행 중), 기본 인스턴스를 중지하고 보조 인스턴스로 재지정하야 합니다.

   ```
   endmqm qmname
   rdqmdr -m qmname -s
   ```

2. 복구 노드에서 다음 명령을 입력하여 보조 큐 관리자를 기본 큐 관리자로 만드십시오.

```
rdqmdr -m qmname -p
```

3. 다음 명령을 입력하여 큐 관리자를 시작하십시오.

```
strmqm qmname
```

4. 애플리케이션이 복구 큐 관리자의 큐 관리자에 다시 연결되어 있는지 확인하십시오. 기본 및 보조 큐 관리자를 지정하여 대체 연결 이름 목록으로 채널을 정의한 경우라면 애플리케이션이 새 기본 큐 관리자에 자동으로 연결됩니다.

관련 정보

strmqm

rdqmdr

V 9.0.5 Linux 재해 복구 구성에서 실패한 노드 대체

재해 복구 구성에서 노드 중 하나가 손실된 경우 다음 프로시저를 사용하여 해당 노드를 교체하고 재해 복구 구성을 복원할 수 있습니다.

이 태스크 정보

재해가 발생하고 기본 사이트의 노드를 복구할 수 없는 경우 큐 관리자를 복구 노드에서 실행하는 동안 실패한 노드를 바꾼 다음 원래 재해 복구 구성을 복원할 수 있습니다. 대체 노드는 실패한 노드의 ID를 나타내야 합니다. 이름과 IP 주소가 동일해야 합니다.

루트로 로그인하거나 mqm 그룹에 속하고 필수 sudo 구성이 있는 사용자로 로그인해야 합니다.

프로시저

기본 사이트에서 큐 관리자가 손실된 후 다음 단계를 수행하십시오.

1. 복구 노드에서 다음 명령을 실행하여 보조 큐 관리자가 기본 역할을 갖도록 만드십시오.

```
rdqmdr -m QMname -p
```

여기서 *QMname*은 큐 관리자의 이름입니다.

2. 재해 복구를 재구성하기 위해 대체 기본 노드에서 실행해야 하는 명령을 검색하십시오.

```
rdqmdr -m QMname -d
```

이 명령의 출력을 복사하십시오.

3. 다음 명령을 실행하여 큐 관리자를 시작하십시오.

```
strmqm QMname
```

4. 애플리케이션이 복구 노드의 큐 관리자에 다시 연결되는지 확인하십시오. 기본 및 보조 큐 관리자를 지정하여 대체 연결 이름 목록으로 채널을 정의한 경우라면 애플리케이션이 새 기본 큐 관리자에 자동으로 연결됩니다.
5. 기본 사이트의 실패한 노드를 교체하고 원래 노드에서 재해 복구에 사용한 동일한 이름 및 IP 주소를 가지도록 이를 구성하십시오. 그런 다음 2단계에서 복사한 **crtmqm** 명령을 실행하여 재해 복구를 구성하십시오. 이제 큐 관리자의 보조 인스턴스가 있으며 기본 인스턴스가 해당 데이터를 보조 인스턴스와 동기화합니다.
6. 현재 기본 인스턴스를 종료하십시오.
7. 동기화가 완료되면 복구 노드에서 실행되고 있는 기본 인스턴스를 보조 인스턴스로 다시 한 번 작성하십시오.

```
rdqmdr -m QMname -s
```

8. 대체 기본 노드에서 큐 관리자의 보조 인스턴스를 기본 인스턴스로 작성하십시오.

```
rdqmdr -m QMname -p
```

9. 대체 기본 노드에서 큐 관리자를 시작하십시오.

   ```
   strmqm QMname
   ```

   이제 기본 사이트에서 장애 전에 있었던 구성을 복원했습니다.

관련 정보

strmqm

rdqmdr

endmqm

로깅: 메시지가 유실되지 않았는지 확인

IBM MQ는 큐 관리자에 의해 제어되는 데이터에 대한 모든 중요한 변경사항을 복구 로그에 기록합니다.

여기에는 오브젝트 작성 및 삭제, 지속 메시지 업데이트, 트랜잭션 상태, 오브젝트 속성에 대한 변경사항 및 채널 활동이 포함됩니다. 이 로그에는 다음을 수행하여 메시지 큐에 대한 모든 업데이트를 복구하는 데 필요한 정보가 들어 있습니다.

- 큐 관리자 변경사항의 레코드 보관
- 재시작 프로세스에 사용할 큐 업데이트의 레코드 보관
- 하드웨어 및 소프트웨어 실패 후 데이터를 복원할 수 있게 함

그러나 IBM MQ도 로그 파일을 포함한 해당 파일을 호스팅하는 디스크 시스템에 의존합니다. 디스크 시스템 자체가 신뢰할 수 없는 경우 로그 정보를 포함한 정보가 계속 유실될 수 있습니다.

로그의 표시

로그는 기본 및 보조 파일과 제어 파일로 구성됩니다. 로그 파일의 수 및 크기와 파일 시스템에 저장되는 위치를 정의합니다.

IBM MQ 로그는 다음 두 개의 컴포넌트로 구성됩니다.

1. 하나 이상의 로그 데이터 파일
2. 로그 제어 파일

로그 데이터 파일은 로그 익스텐트라고도 합니다.

기록되는 데이터를 포함하는 여러 로그 익스텐트가 있습니다. 개수와 크기(100 페이지의 『IBM MQ의 로그 기본값』에 설명된 대로)를 정의하거나 시스템 기본값인 3개의 기본 파일과 2개의 보조 익스텐트를 사용할 수 있습니다.

3개의 기본 및 2개의 보조 익스텐트의 기본값은 16MB입니다.

큐 관리자를 작성할 때 사전 할당된 로그 익스텐트 개수는 할당된 기본 로그 익스텐트의 수입니다. 숫자를 지정하지 않을 경우 기본값이 사용됩니다.

IBM MQ는 두 가지 유형의 로깅을 사용합니다.

- 순환
- 선형

  선형 로깅에 사용되는 로그 익스텐트 수는 매체 이미지 기록 빈도에 따라 매우 많아질 수 있습니다.

자세한 정보는 478 페이지의 『로깅 유형』의 내용을 참조하십시오.

IBM MQ for Windows에서 로그 경로를 변경하지 않은 경우 로그 익스텐트는 다음 디렉토리에 작성됩니다.

```
C:\ProgramData\IBM\MQ\log\QMgrName
```

IBM MQ for UNIX and Linux 시스템에서 로그 경로를 변경하지 않은 경우 로그 익스텐트는 다음 디렉토리에 작성됩니다.

```
/var/mqm/log/QMgrName
```

IBM MQ는 이러한 기본 로그 익스텐트로 시작되지만 기본 로그 공간이 충분하지 않은 경우 보조 로그 익스텐트를 할당합니다. 이를 동적으로 수행하며 로그 공간에 대한 수요가 감소되면 보조 로그 파일을 제거합니다. 기본적으로 최대 두 개의 보조 로그 익스텐트가 할당될 수 있습니다. 78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』에 설명된 대로 이 기본 할당을 변경할 수 있습니다.

로그 익스텐트 앞에는 S 문자 또는 R 문자가 붙습니다. 활성, 비활성 및 불필요한 익스텐트 앞에는 S가 붙지만 재사용 익스텐트 앞에는 R이 붙습니다.

큐 관리자를 백업하거나 복원하는 경우 모든 활성, 비활성 및 불필요한 익스텐트를 로그 제어 파일과 함께 백업하고 복원하십시오.

참고: 재사용 익스텐트는 백업 및 복원하지 않아도 됩니다.

로그 제어 파일

로그 제어 파일에는 로그 익스텐트의 상태를 설명하는 데 필요한 정보가 포함됩니다(예: 로그 파일의 크기와 위치 및 그 다음 사용 가능한 파일의 이름).

중요사항: 로그 제어 파일은 내부 큐 관리자 전용입니다.

큐 관리자는 로그 제어 파일에 복구 로그의 상태와 연관된 제어 데이터를 보관하고 로그 제어 파일의 컨텐츠를 수정해서는 안됩니다.

로그 제어 파일은 로그 경로에 있으며 `amqhlctl.lfh`라고 합니다. 큐 관리자를 백업하거나 복원하는 경우 로그 제어 파일이 로그 익스텐트와 함께 백업 및 복원되는지 확인하십시오.

로깅 유형

IBM MQ에서는 순환 로깅과 선형 로깅이라는 두 가지 방식으로 큐 관리자 활동의 레코드를 유지합니다.

순환 로깅

재시작 복구만 필요한 경우 로그를 사용하여 시스템이 중지될 때 진행 중이었던 트랜잭션을 롤백하는 순환 로깅을 사용하십시오.

순환 로깅은 모든 재시작 데이터를 로그 파일의 링에 보관합니다. 로깅은 링에서 첫 번째 파일을 채운 후 다음 단계로 이동하며 모든 파일이 가득 찰 때까지 계속됩니다. 그런 다음 링에서 첫 번째 파일로 돌아와서 다시 시작합니다. 이 순환은 제품을 사용 중인 한 계속되며 로그 파일 부족이 발생하지 않는 장점이 있습니다.

IBM MQ는 큐 관리자 데이터 복구를 보장하기 위해 데이터 손실 없이 큐 관리자를 재시작하는 데 필요한 로그 입력 항목을 더 이상 필요하지 않을 때까지 보관합니다. 재사용을 위해 로그 파일을 릴리스하기 위한 메커니즘은 480 페이지의 『체크포인트를 사용한 전체 복구 보장』에 설명되어 있습니다.

선형 로깅

재시작 복구 및 매체 복구를 둘 다 원하는 경우 선형 로깅을 사용하십시오(로그의 컨텐츠를 재실행하여 손실되거나 손상된 데이터를 다시 작성). 선형 로깅에서는 일련의 연속 로그 파일에 로그 데이터를 보관합니다.

선택적으로 로그 파일에 대해 다음을 수행할 수 있습니다.

- 재사용(재시작 복구 또는 매체 복원을 위해 더 이상 필요하지 않은 경우에만)
- 장기 저장 및 분석을 위해 수동으로 아카이브

매체 이미지의 빈도는 선형 로그 파일을 재사용할 수 있는 시기를 판별하므로 선형 로그 파일에 사용 가능해야 하는 디스크 공간의 크기를 결정하는 데 있어 중요한 요인입니다.

시간 또는 로그 사용량을 기반으로 자동으로 주기적 매체 이미지를 기록하도록 큐 관리자를 구성하거나 수동으로 매체 이미지를 스케줄할 수 있습니다.

구현할 정책 및 디스크 공간 사용량에 대한 의미는 관리자가 결정합니다. 재시작 복구에 필요한 로그 파일은 항상 사용 가능해야 하지만 매체 복원을 위해서만 필요한 로그 파일은 장기 스토리지(예: 테이프)에 아카이브할 수 있습니다.

관리자가 자동 로그 관리 및 자동 매체 이미지를 사용으로 설정하면 선형 로깅은 매우 큰 순환 로그의 경우와 비슷한 방식으로 작동하지만 매체 복원에 의해 사용으로 설정된 매체 장애에 대한 중복성이 개선됩니다.

V 9.0.4 IBM MQ 9.0.4에서는 migmqlog 명령을 사용하여 큐 관리자의 기존 로그 유형을 선형에서 순환 또는 순환에서 선형으로 변경할 수 있습니다.

로거 변경

V 9.0.2 Multi

IBM MQ 9.0.2부터는 자동 로그 관리(아카이브 포함)를 사용하는 경우 로거가 활성 상태가 아닌 선형 로그 익스텐트를 추적합니다.

주의: 아카이브 없이 자동 로그 관리를 사용하는 경우에는 이 프로세스에 대해 백업 큐 관리자 사용이 지원되지 않습니다.

ULW 복구를 위해 로그 익스텐트가 더 이상 필요하지 않을 때 필요한 경우 해당 로그 익스텐트가 아카이브되면 로거는 편리한 지점에서 해당 로그 익스텐트를 삭제하거나 재사용합니다.

재사용된 로그 익스텐트는 로그 순서에서 다음 로그 익스텐트로 이름이 바뀝니다. 작성되거나 삭제되거나 재사용된 익스텐트 수를 표시하는 AMQ7490 메시지가 주기적으로 기록됩니다.

로거는 재사용할 준비가 된 상태를 유지할 익스텐트 수와 해당 익스텐트를 삭제할 시기를 선택합니다.

활성 로그

선형 및 순환 로깅 모두에서 활성으로 불리는 파일이 여러 개 있습니다. 순환 로깅인지 아니면 선형 로깅인지에 관계없이 활성 로그는 재시작 복구에서 참조하는 로그 공간의 최대 크기입니다.

활성 로그 파일의 수는 일반적으로 구성 파일에 정의된 1차 로그 파일의 수보다 적습니다. (이 활성 로그 파일 수 정의에 대한 정보는 483 페이지의 『로그의 크기 계산』의 내용을 참조하십시오.)

활성 로그 공간에는 매체 복원을 위해 필요한 공간이 포함되지 않으며 선형 로깅에 사용되는 로그 파일 수는 메시지 플로우 및 매체 이미지 빈도에 따라 매우 많을 수 있습니다.

비활성 로그

로그 파일이 재시작 복구를 위해 더 이상 필요하지 않으면 해당 로그 파일은 `inactive` 상태가 됩니다. 재시작 복구 또는 매체 복원을 위해 필요하지 않은 로그 파일은 불필요한 로그 파일로 간주할 수 있습니다.

자동 로그 관리를 사용하는 경우 큐 관리자는 이 불필요한 로그 파일의 처리를 제어합니다. 수동 로그 관리를 선택한 경우 불필요한 로그 파일이 더 이상 조작과 관련이 없을 때 불필요한 로그 파일을 관리(예: 삭제 및 아카이브)하는 것은 관리자가 담당하게 됩니다.

로그 파일의 속성 지정에 대한 추가 정보는 488 페이지의 『로그 관리』의 내용을 참조하십시오.

보조 로그 파일

선형 로깅에 대해 보조 로그 파일이 정의되더라도 이러한 파일은 정상 조작에서는 사용되지 않습니다. 재시작을 위해 계속해서 필요할 수 있으므로 파일을 활성 풀로부터 비울 수 없는 상황이 발생하면(아마도 장기 트랜잭션으로 인해) 보조 파일이 형식화되고 활성 로그 파일 풀에 추가됩니다.

사용 가능한 보조 파일 수를 다 사용한 경우 로그 활동이 필요한 가장 마지막 조작에 대한 요청이 거부되고 MQRC_RESOURCE_PROBLEM 리턴 코드가 애플리케이션에 리턴되며 장기 실행 트랜잭션은 모두 비동기 롤백을 위해 고려됩니다.

주의: 두 가지 유형의 로깅 모두 하드웨어 실패가 없다는 가정 하에 예상치 못한 전원 손실에 대처할 수 있습니다.

체크포인트를 사용한 전체 복구 보장

순환 로깅 및 선형 로깅 큐 관리자는 모두 재시작 복구를 지원합니다. 큐 관리자의 이전 인스턴스가 갑자기 종료되는 수준(예: 정전)에 관계없이 재시작 시 큐 관리자는 지속 상태를 종료 시 올바른 트랜잭션 상태로 복원합니다.

재시작 복구는 유지 중인 디스크 무결성에 의존합니다. 마찬가지로 운영 체제는 운영 체제가 갑자기 종료되는 수준에 관계없이 디스크 무결성을 보장해야 합니다.

디스크 무결성이 유지되는 않는 매우 비정상적인 이벤트에서는 선형 로깅(및 매체 복원)이 몇몇 추가적인 중복성 및 복구 가능성 옵션을 제공합니다. 점점 더 일반화되는 기술(예: RAID)을 사용하면서 디스크 무결성 문제가 발생하는 경우가 점점 드물어지고 많은 엔터프라이즈에서는 순환 로깅을 구성하고 재시작 복구만 사용합니다.

IBM MQ는 전형적인 미리 쓰기 로깅 자원 관리자로 설계되어 있습니다. 메시지 큐에 대한 지속적인 업데이트는 두 단계로 발생합니다.

1. 업데이트를 나타내는 로그 레코드가 복구 로그에 신뢰성 있게 기록됩니다.
2. 시스템에 대해 가장 효율적인 방식으로 큐 파일 또는 버퍼가 업데이트되지만 일관되지 않아도 됩니다.

따라서 로그 파일은 기본 큐 버퍼 및 파일 상태보다 최신 상태가 됩니다.

이 상황이 계속 업데이트될 수 있는 경우에는 충돌 복구 후 큐 상태의 일관성을 유지하기 위해 매우 큰 볼륨의 로그 재실행이 필요합니다.

IBM MQ는 충돌 복구 후 필요한 로그 재실행의 볼륨을 제한하기 위해 `checkpoints`를 사용합니다. 로그 파일이 활성이라고 하는지 여부를 제어하는 키 이벤트이거나 `checkpoint`입니다.

IBM MQ 체크포인트는 다음과 같은 지점입니다.

- 복구 로그와 오브젝트 파일 사이의 일관성 지점
- 큐 관리자가 종료될 때 올바른 논리 상태로 큐를 복원하도록 보장하기 위해 후속 로그 레코드의 정방향 재실행이 보장되는 로그에서의 위치를 식별하는 지점

체크포인트 수행 중에 IBM MQ는 충돌 복구 후 큐를 다시 일관된 상태로 가져오기 위해 재실행해야 하는 로그 레코드의 볼륨을 제한하기 위해 필요에 따라 큐 파일에 대한 이전 업데이트를 비웁니다.

최근 완료 체크포인트는 충돌 복구 중에 재실행을 수행해야 하는 로그에서의 지점을 표시합니다. 따라서 체크포인트 빈도는 체크포인트 기록의 오버헤드와 해당 체크포인트가 나타내는 잠재적 복구 시간 개선을 상호 보완합니다.

로그에서 최근 완료 체크포인트의 시작 위치는 로그 파일이 활성인지 아니면 비활성인지를 판별하는 주요 요소 중 하나입니다. 다른 주요 요소는 로그에서 현재 활성 트랜잭션에 의해 작성된 첫 번째 지속적 업데이트와 관련된 첫 번째 로그 레코드의 위치입니다.

새 체크포인트가 두 번째 이상의 로그 파일에서 기록되고 현재 트랜잭션이 첫 번째 로그 파일에 있는 로그 레코드를 참조하지 않는 경우에는 첫 번째 로그 파일이 비활성 상태가 됩니다. 순환 로깅의 경우 이제 첫 번째 로그 파일을 재사용할 준비가 되었습니다. 선형 로깅의 경우에는 매체 복원을 위해 여전히 첫 번째 로그 파일이 일반적으로 필요합니다.

순환 로깅 또는 자동 로그 관리를 구성하는 경우 큐 관리자는 비활성 로그 파일을 관리합니다. 수동 로그 관리를 사용하여 선형 로깅을 구성하는 경우 이는 조작의 요구사항에 따라 비활성 파일을 관리하는 관리 태스크가 됩니다.

IBM MQ는 체크포인트를 자동으로 생성합니다. 다음의 경우 수행됩니다.

- 큐 관리자가 시작될 때
- 시스템 종료 시
- 로깅 공간이 부족할 때
- Multi 이전 체크포인트가 수행된 이후 50,000개의 조작이 로깅된 후
- z/OS 이전 체크포인트가 수행된 이후 *number_of_operations*개의 조작이 로깅된 후. 여기서 *number_of_operations*는 **LOGLOAD** 특성에 설정된 조작의 수입니다.

IBM MQ가 재시작되면 로그에서 최근 체크포인트 레코드를 찾습니다. 이 정보는 모든 체크포인트의 끝에서 업데이트되는 체크포인트 파일에 있습니다. 체크포인트 이후 발생한 모든 조작은 앞으로 재실행됩니다. 이를 재실행 단계라고 합니다.

재실행 단계는 큐를 시스템 실패 또는 종료 전 논리 상태로 되돌립니다. 재실행 단계 중에 시스템 실패 또는 종료가 발생하기 전에 인플라이트였던 트랜잭션의 목록이 작성됩니다.

Multi 메시지 AMQ7229 및 AMQ7230은 재실행 단계의 진행 상태를 표시하기 위해 발행됩니다.

백아웃하거나 커미트할 조작을 알기 위해서 IBM MQ는 인플라이트 트랜잭션과 연관된 각 활성 로그 레코드에 액세스합니다. 이를 복구 단계라고 합니다.

Multi 메시지 AMQ7231, AMQ7232 및 AMQ7234는 복구 단계의 진행 상태를 표시하기 위해 발행됩니다.

복구 단계 중 필요한 모든 로그 레코드에 액세스하면 각 활성 트랜잭션이 순서대로 해석되고 트랜잭션과 연관된 각 조작은 백아웃되거나 커미트됩니다. 이를 해석 단계라고 합니다.

Multi 메시지 AMQ7233은 해석 단계의 진행 상태를 표시하기 위해 발행됩니다.

z/OS z/OS의 경우 재시작 처리는 다양한 단계로 구성됩니다.

1. 작업 단위를 백아웃하고 인다우트(in-doubt) 작업 단위에 대한 잠금을 확보하기 위해 필요한 가장 오래된 로그 레코드와 페이지 세트에 필요한 매체 복원을 기반으로 복구 로그 범위가 설정됩니다.
2. 로그 범위가 판별되고 나면 정방향 로그 읽기가 수행되어 페이지 세트를 최신 상태로 가져오고 인다우트(in-doubt) 또는 인플라이트 작업 단위와 관련된 메시지도 모두 잠급니다.
3. 정방향 로그 읽기가 완료된 경우 로그를 역방향으로 읽어서 장애 시 인플라이트 또는 인백아웃(in-backout) 상태였던 작업 단위를 모두 백아웃합니다.

z/OS 표시되는 메시지의 예는 다음과 같습니다.

```
CSQR001I +MQOX RESTART INITIATED
CSQR003I +MQOX RESTART - PRIOR CHECKPOINT RBA=00000001E48C0A5E
CSQR004I +MQOX RESTART - UR COUNTS -  806
IN COMMIT=0, INDOUBT=0, INFLIGHT=0, IN BACKOUT=0
CSQR030I +MQOX Forward recovery log range  815
from RBA=00000001E45FF7AD to RBA=00000001E48C1882
CSQR005I +MQOX RESTART - FORWARD RECOVERY COMPLETE -  816
IN COMMIT=0, INDOUBT=0
CSQR032I +MQOX Backward recovery log range  817
from RBA=00000001E48C1882 to RBA=00000001E48C1882
CSQR006I +MQOX RESTART - BACKWARD RECOVERY COMPLETE -  818
INFLIGHT=0, IN BACKOUT=0
CSQR002I +MQOX RESTART COMPLETED
```

참고: 읽을 로그의 양이 많은 경우 진행 상황을 표시하기 위해 CSQR031I(정방향 복구) 및 CSQR033I(역방향 복구) 메시지가 주기적으로 발행됩니다.

482 페이지의 그림 86에서 최근 체크포인트(체크포인트 2) 이전 모든 레코드는 IBM MQ에서 더 이상 필요하지 않습니다. 큐는 체크포인트 정보 및 이후 로그 입력 항목으로부터 복구할 수 있습니다. 순환 로깅에서는 체크포인트 이전 비워진 파일을 재사용할 수 있습니다. 선형 로깅에서는 비워진 로그 파일이 정상 조작을 위해 더 이상 액세스되지 않으며 비활성이 됩니다. 이 예제에서, 큐 헤드 포인터는 마지막 체크포인트인 체크포인트 2를 가리키도록 이동되며, 체크포인트 2는 새로운 큐 헤드인 헤드 2가 됩니다. 이제 로그 파일 1을 재사용할 수 있습니다.

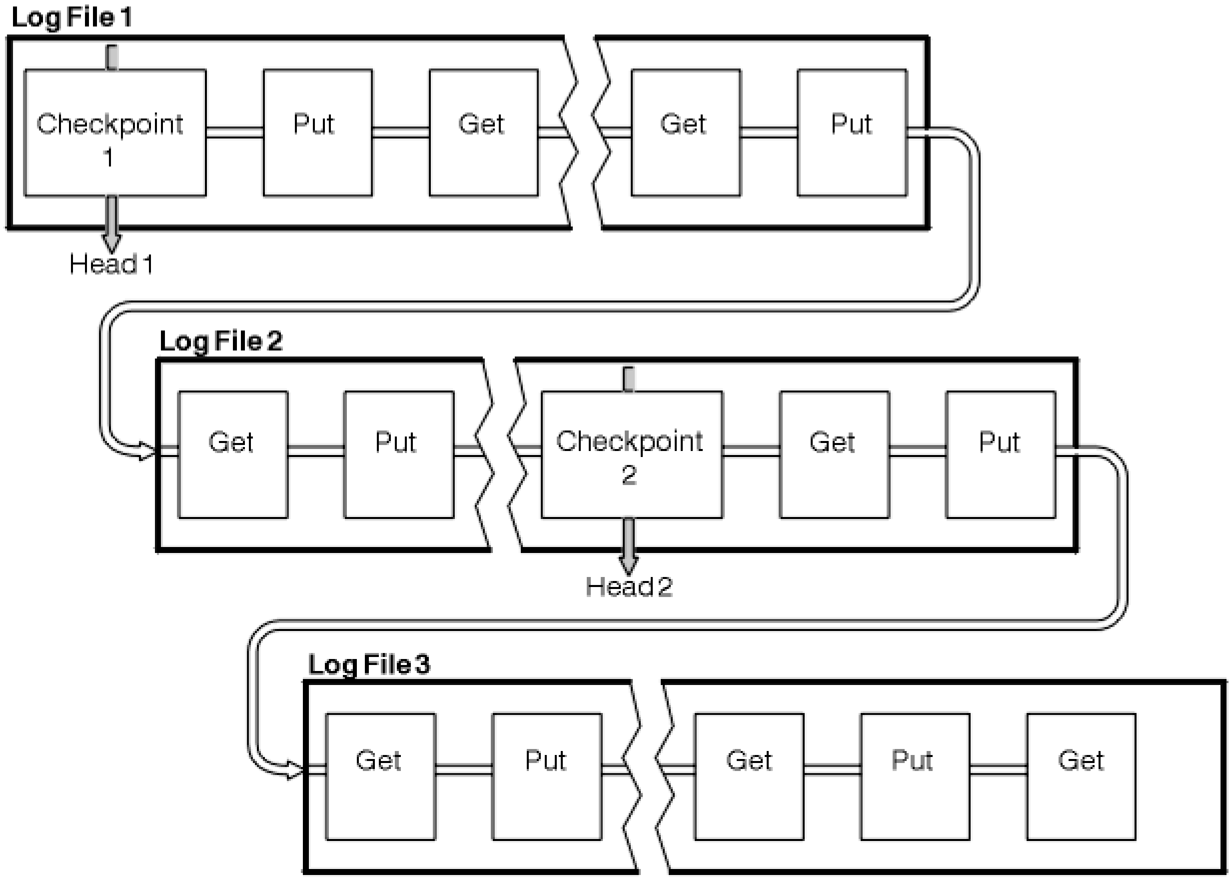


그림 *86.* 체크포인팅

장기 실행 트랜잭션 체크포인트

장기 실행 트랜잭션이 로그 파일의 재사용에 영향을 미치는 방식입니다.

483 페이지의 그림 87에서는 장기 실행 트랜잭션이 로그 파일의 재사용에 영향을 미치는 방식을 보여줍니다. 이 예에서 장기 실행 트랜잭션은 표시된 첫 번째 체크포인트 뒤에 LR1으로 표시되는 로그에 대한 입력 항목을 작성했습니다. 이 트랜잭션은 세 번째 체크포인트를 지나서까지(포인트 LR2) 완료되지 않습니다. LR1 이후의 모든 로그 정보는 해당 트랜잭션의 복구를 위해 보존되며 필요한 경우 그러한 복구가 완료될 때까지 보존됩니다.

장기 실행 트랜잭션이 완료된 후 LR 2에서 로그의 헤드는 최근에 로깅된 체크포인트인 체크포인트 3으로 이동됩니다. 체크포인트 3, 헤드 2 전의 로그 레코드를 포함하는 파일은 더 이상 필요하지 않습니다. 순환 로깅을 사용하는 경우 공간을 재사용할 수 있습니다.

장기 실행 트랜잭션이 완료되기 전에 1차 로그 파일이 완전히 채워지면 2차 로그 파일을 사용하여 로그가 가득 채워지지 않게 합니다.

전적으로 큐 관리자의 제어 하에 있는 활동(예: 체크포인팅)은 기본 로그 내에서 활동을 시도하고 보관하도록 스케줄되어 있습니다.

하지만 보조 로그 공간이 큐 관리자의 제어를 벗어난 작동(예: 트랜잭션 중 하나의 지속 기간)을 지원해야 하는 경우 큐 관리자는 정의된 보조 로그 공간을 사용하여 해당 활동이 완료될 수 있게 합니다.

해당 활동이 총 로그 공간의 80%가 사용될 때까지 완료되지 않는 경우 큐 관리자는 애플리케이션에 영향을 미친다는 사실에 관계없이 로그 공간 재확보 조치를 시작합니다.

로그 헤드가 이동하고 순환 로깅을 사용하는 경우 1차 로그 파일이 재사용에 적합해질 수 있고 현재 파일을 채운 후에 로거는 사용 가능한 첫 번째 1차 파일을 재사용합니다. 선형 로깅을 사용하는 경우 로그 헤드가 여전히 아래로 이동하면 활성 풀 및 첫 번째 파일이 비활성이 됩니다. 이후 로깅 활동을 준비하기 위해 새 1차 파일이 형식화되어 풀의 맨 아래에 추가됩니다.

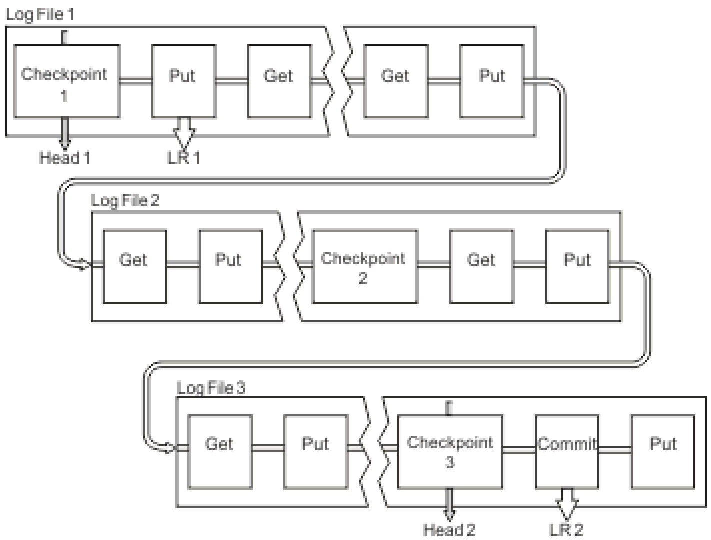


그림 *87.* 장기 실행 트랜잭션 체크포인트

로그의 크기 계산

큐 관리자가 필요로 하는 로그의 크기를 계산합니다.

큐 관리자가 순환 또는 선형 로깅을 사용하는지 여부를 결정한 후에 큐 관리자가 필요로 하는 활성 로그의 크기를 계산해야 합니다. 활성 로그의 크기는 다음 로그 구성 매개변수에 의해 판별됩니다.

LogFilePages
: 각 1차 및 보조 로그 파일의 크기(4K 페이지 단위)

LogPrimaryFiles
: 사전 할당된 1차 로그 파일의 수

LogSecondaryFiles
: 1차 로그 파일이 다 차면 사용하기 위해 작성될 수 있는 2차 로그 파일의 수

참고사항:

1. 2차 로그에 변경한 내용이 즉시 영향을 미치지 않더라도 큐 관리자가 시작할 때마다 1차 및 2차 로그 파일의 수를 변경할 수 있습니다.
2. 로그 파일의 크기를 변경할 수 없으며 큐 관리자를 작성하기 **전에** 로그 파일 크기를 판별해야 합니다.
3. 1차 로그 파일의 수 및 로그 파일 크기는 큐 관리자가 작성될 때 사전 할당되는 로그 공간의 양을 판별합니다.
4. 1차 및 보조 로그의 파일의 총 수는 UNIX and Linux 시스템에서는 511을 초과할 수 없고 Windows에서는 255를 초과할 수 없으며 이는 장기 실행 트랜잭션이 있을 때에는 재시작 복구를 위해 큐 관리자가 사용할 수 있는 최대 로그 공간 양을 제한합니다. 큐 관리자가 매체 복구에 필요로 할 수 있는 로그 공간의 양은 이러한 제한을 공유하지 않습니다.
5. 순환 로깅을 사용 중일 때 큐 관리자는 1차 및 2차 로그 공간을 재사용합니다. 로그 파일이 다 차고 순서 상으로 다음인 1차 로그 파일이 사용 불가능하면 큐 관리자는 보조 로그 파일을 최대로 할당합니다.

   할당해야 할 로그 수에 대한 자세한 정보는 484 페이지의 『내 활성 로그는 얼마나 커야 합니까?』의 내용을 참조하십시오. 기본 로그 익스텐트를 순차적으로 사용하고 해당 순서는 변경되지 않습니다.

   예를 들어, 세 개의 1차 로그 `0`, `1` 및 `2`가 있는 경우 사용 순서는 `0`,`1`,`2` 다음에 `1`,`2`,`0`, `2`,`0`,`1` 다시 `0`,`1`,`2` 식입니다. 할당한 모든 2차 로그는 필요에 따라 배치됩니다.
6. 체크포인트 중에 1차 로그 파일을 재사용할 수 있게 됩니다. 로그 공간의 양이 줄고 있으므로 큐 관리자는 체크포인트를 사용하기 전에 1차 및 2차 로그 공간을 모두 고려합니다.

   V 9.0.2 IBM MQ 9.0.2부터 큐 관리자는 로그 사용을 1차 범위 내에서 유지하는 방식으로 체크포인트를 스케줄하도록 시도합니다.

자세한 정보는 100 페이지의 『IBM MQ의 로그 기본값』의 내용을 참조하십시오.

내 활성 로그는 얼마나 커야 합니까?

큐 관리자에 필요한 활성 로그의 크기를 추정합니다.

활성 로그 크기는 다음만큼 제한됩니다.

```
logsize = (primaryfiles + secondaryfiles) * logfilepages * 4096
```

큐 관리자가 초당 최대량의 데이터를 디스크에 쓰는 경우 장기 실행 트랜잭션에 맞게 로그가 충분히 커야 합니다.

장기 실행 트랜잭션이 N초 동안 실행되고 있고 큐 관리자가 디스크에 쓴 초당 최대 데이터 양이 로그에서 초당 B 바이트인 경우 최소한 다음과 같아야 합니다.

```
logsize >= 2 * (N+1) * B
```

최대 워크로드에서 실행 중이거나 매체 이미지를 기록 중인 경우 큐 관리자는 초당 최대량의 데이터를 디스크에 씁니다.

트랜잭션이 장기간 실행되고 첫 번째 로그 레코드를 포함하는 로그 익스텐트가 활성 로그에 포함되지 않는 경우, 큐 관리자는 가장 오래된 로그 레코드가 포함된 트랜잭션부터 시작하여 한 번에 하나의 트랜잭션을 롤백합니다.

큐 관리자는 기본 및 보조 파일의 최대 수가 사용되기 전에 이전 로그 익스텐트를 비활성 상태로 만들어야 하며 큐 관리자는 다른 로그 익스텐트를 할당해야 합니다.

큐 관리자가 롤백을 허용하기 전에 장기 실행 트랜잭션을 실행할 기간을 결정하십시오. 장기 실행 트랜잭션이 느린 네트워크 트래픽을 대기하거나, 트랜잭션 설계가 잘못된 경우 사용자 입력을 대기할 수 있습니다.

runmqsc 명령을 실행하여 장기 실행 트랜잭션이 실행된 기간을 조사할 수 있습니다.

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

dspmqtrn -a 명령을 실행하면 모든 상태의 XA 및 비XA 명령이 표시됩니다.

이 명령을 실행하면 모든 현재 트랜잭션에 대해 기록된 첫번째 로그 레코드의 날짜 및 시간이 표시됩니다.

주의: 로그 크기를 계산하기 위해 트랜잭션 또는 애플리케이션이 시작된 이후의 시간이 아니라 해당하는 첫 번째 로그 레코드가 기록된 이후의 시간입니다. 장기 실행 트랜잭션의 기간을 가장 가까운 시간으로 반올림합니다. 이는 큐 관리자의 최적화때문입니다.

예를 들어 실제로 메시지를 가져오기 전에 장기간 대기하는 MQGET 호출을 실행하여 애플리케이션이 시작된 경우, 애플리케이션이 시작된 후 첫 번째 로그 레코드를 쓸 수 있습니다.

처음 실행한

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

명령에서 출력된 최대 날짜 및 시간을 검토하여 장기 실행 트랜잭션 시간을 추정할 수 있습니다.

장기 실행 트랜잭션의 기간을 미달 추정하지 않도록 최대 워크로드에서 장기 실행 트랜잭션이 실행되는 동안 **runmqsc** 명령을 반복해서 실행하십시오.

IBM MQ 8.0에서 운영 체제 도구를 사용하십시오. 예를 들어, UNIX 플랫폼에서 **iostat**를 사용하십시오.

IBM MQ 9.0부터는 다음 명령을 실행하여 큐 관리자가 로그에 기록하는 초당 바이트를 확인할 수 있습니다.

```
amqsrua -m qmgr -c DISK -t Log
```

기록된 논리적 바이트는 큐 관리자가 로그에 기록하는 초당 바이트 수를 표시합니다. 예를 들면, 다음과 같습니다.

```
$ amqsrua -m mark -c DISK -t Log
Publication received PutDate:20160920 PutTime:15383157 Interval:4 minutes,39.579 seconds
Log - bytes in use 37748736
Log - bytes max 50331648
Log file system - bytes in use 316243968
Log file system - bytes max 5368709120
Log - physical bytes written 4334030848 15501948/sec
```

```
Log - logical bytes written 3567624710 12760669/sec
Log - write latency 411 uSec
```

이 예에서 로그에 기록된 초당 논리적 바이트 수는 12760669/sec 또는 초당 약 12MiB입니다.

사용

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

을 사용하면 장기 실행 트랜잭션이 다음과 같이 표시됩니다.

```
CONN(57E14F6820700069)
EXTCONN(414D51436D61726B2020202020202020)
TYPE(CONN)
APPLTAG(msginteg_r)                    UOWLOGDA(2016-09-20)
UOWLOGTI(16.44.14)
```

현재 날짜 및 시간이 2016-09-20 16.44.19이기 때문에 이 트랜잭션은 5초동안 실행되었습니다. 하지만 큐 관리자가 트랜잭션을 롤백하기 전에 10초 동안 트랜잭션이 실행되는 것을 허용해야 합니다. 따라서 로그 크기는 다음과 같아야 합니다.

```
2 * (10 + 1) * 12 = 264 MiB
```

.

로그 파일 수는 예상되는 가장 큰 로그 크기(선행 텍스트에서 계산됨)를 포함할 수 있어야 합니다. 이는 다음과 같습니다.

로그 파일의 최소 수 = (필요한 로그 크기) / (**LogFilePages** * 로그 파일 페이지 크기(4096))

기본 **LogFilePages**(4096) 및 선행 텍스트에서 계산된 264MiB의 로그 크기 추정치를 사용하는 경우 로그 파일의 최소 수는 다음과 같아야 합니다.

```
264MiB / (4096 x 4096) = 16.5
```

즉, 17개의 로그 파일입니다.

예상 워크로드가 기본 파일 내에서 실행되도록 로그 크기를 정할 경우:

- 보조 파일은 추가 로그 공간이 필요한 경우 일부 대책을 제공합니다.
- 순환 로깅이 항상 사전 할당된 기본 파일을 사용하며 이는 보조 파일을 할당하고 할당 취소하는 것보다 약간 빠릅니다.
- 큐 관리자는 기본 파일에 남아있는 공백만 사용하여 다음 체크포인트를 가져오는 시점을 계산합니다.

따라서 선행 예에서 워크로드가 기본 로그 파일 내에서 실행되도록 다음과 같은 값을 설정하십시오.

- **LogFilePages** = 4096
- **LogPrimaryFiles** = 17
- **LogSecondaryFiles** = 5

다음에 유의하십시오.

- 이 예제에서 5 보조 파일은 활성 로그 공간의 20퍼센트 이상입니다.

  V 9.0.2 IBM MQ 9.0.2부터 로거는 기본 파일에만 워크로드를 보관합니다. 따라서 로거는 기본 파일의 일부가 가득 찰 때 체크포인트를 스케줄합니다.

  V 9.0.2 예상치 못하게 장기 실행되는 트랜잭션이 있는 경우 대책으로 보조 파일을 가지고 있습니다.

  총 로그 공간의 80퍼센트 이상을 사용하는 경우 큐 관리자가 로그 공간 사용을 줄이는 조치를 수행합니다.
- 선형 또는 순환 로깅을 사용하는지 여부와 상관없이 동일한 계산을 수행합니다.

  선형 로깅 및 순환 로깅 모두에서 활성 로그의 개념이 동일하기 때문에 선형 또는 순환 활성 로그 크기를 계산하는지 여부와 상관없이 차이가 없습니다.

- 매체 복원을 위해서만 필요한 로그 익스텐트는 활성 로그에 없으므로 기본 및 보조 파일의 수 계산에 포함되지 않습니다.
- V 9.0.2 IBM MQ 9.0.2 에서 DISPLAY QMSTATUS LOG 의 *LOGUTIL* 필드를 사용하여 필요한 활성 로그의 크기를 대략적으로 계산할 수 있습니다.

  이 필드는 가장 오래 실행되는 트랜잭션의 지속 기간 또는 큐 관리자의 최대 처리량을 판별하기 위해 지속적으로 샘플링을 수행하지 않고도 필요한 로그 크기의 합리적인 추정치를 작성할 수 있도록 설계되어 있습니다.

LogFilePages의 크기

일반적으로 LogFilePages의 크기를 충분히 크게 하여 최대 기본 파일 수에 도달하지 않고 활성 로그의 크기를 쉽게 늘릴 수 있게 하십시오. 소수의 대형 로그 파일이 다수의 소형 로그 파일보다 선호되며 그 이유는 소수의 대형 로그 파일이 필요할 때 로그 크기를 늘릴 수 있는 유연성이 더 높기 때문입니다.

선형 로깅의 경우 너무 큰 로그 파일은 성능의 변동을 야기할 수 있습니다. 아주 큰 로그 파일의 경우 새 로그 파일을 작성 및 형식화하거나 이전 것을 아카이브할 때 더 큰 단계가 필요합니다. 이는 자동 로그 관리로 인해 새 로그 파일이 잘 작성되지 않으므로 수동 및 아카이브 로그 관리 문제 이상의 문제가 됩니다.

너무 작은 로그를 작성한 경우 발생하는 상황

최소 크기의 로그를 추정할 때 고려해야 할 점입니다.

너무 작은 로그를 작성한 경우

- 장기 실행 트랜잭션이 백아웃됩니다.
- 이전 체크포인트가 종료되기 전에 다음 체크포인트가 시작하기를 원합니다.

중요사항: 로그 크기를 얼마나 부정확하게 추정하는지는 상관없이 데이터 무결성이 유지됩니다.

체크포인트에 대한 설명은 480 페이지의 『체크포인트를 사용한 전체 복구 보장』의 내용을 참조하십시오. 활성 로그 익스텐트에 남아있는 로그 공간이 줄어들 경우 큐 관리자는 더 많은 체크포인트를 자주 스케줄링합니다.

체크포인트는 즉각적이지 않기 때문에 약간 시간이 걸립니다. 체크포인트에 더 많은 데이터를 기록해야 할수록 체크포인트에 시간이 더 걸립니다. 로그가 작을 경우 체크포인트가 겹칠 수 있습니다. 즉 이전 체크포인트가 종료되기 전에 다음 체크포인트가 요청됨을 의미합니다. 이 상황이 발생하면 오류 메시지가 기록됩니다.

장기 실행 트랜잭션이 백아웃되거나 체크포인트가 겹칠 경우 큐 관리자는 워크로드 처리를 계속합니다. 단기 트랜잭션은 계속 정상으로 실행됩니다.

그러나 큐 관리자는 최적으로 실행되지 않고 성능이 저하될 수 있습니다. 충분한 로그 공간을 사용하여 큐 관리자를 다시 시작해야 합니다.

너무 큰 로그를 작성한 경우 발생하는 상황

최대 크기의 로그를 추정할 때 고려해야 할 점입니다.

너무 큰 로그를 작성한 경우

- 가능성은 없지만 긴급 재시작에 걸리는 시간이 증가할 수 있습니다.
- 불필요한 디스크 공간을 사용하고 있습니다.
- 장기 실행 트랜잭션이 허용됩니다.

중요사항: 로그 크기를 얼마나 부정확하게 추정하는지는 상관없이 데이터 무결성이 유지됩니다.

V 9.0.2 로그의 최대 크기를 추정하는 데 도움이 되도록 로그 이용 통계를 사용할 수 있습니다. 추가적인 정보는 491 페이지의 『IMGLOGLN 및 IMGINTVL 설정 방법 결정』 및 ALTER QMGR을 참조하십시오.

재시작시 큐 관리자가 로그를 읽는 방법에 대한 설명은 480 페이지의 『체크포인트를 사용한 전체 복구 보장』의 내용을 참조하십시오. 큐 관리자는 마지막 체크포인트의 로그를 재생한 다음, 큐 관리자가 종료될 때 활성 상태인 모든 트랜잭션을 해석합니다.

트랜잭션을 해석하기 위해 큐 관리자가 해당 트랜잭션과 연관된 모든 로그 레코드를 읽습니다. 이 로그 레코드가 마지막 체크포인트보다 이전일 수 있습니다.

큐 관리자에 대용량 로그를 할당하면 재시작시 로그의 모든 로그 레코드를 읽을 수 있는 권한을 큐 관리자에게 제공하게 됩니다. 그러나 큐 관리자는 대개 이런 작업을 수행할 필요가 없습니다. 잠재적으로 이런 상황이 발생할 가능성은 거의 없지만 해당 프로세스에 시간이 오래 걸릴 수 있습니다.

큐 관리자가 종료되기 전에 체크포인트가 예기치 않게 중지되면 대용량 로그가 포함된 큐 관리자의 재시작 시간이 크게 증가합니다. 로그 크기를 제한하면 긴급 재시작 시간이 제한됩니다.

이 문제를 방지하려면 다음을 확인하십시오.

- 워크로드가 지나치게 크지 않은 로그에 적당하게 맞는지 확인하십시오.
- 장기 실행 트랜잭션을 방지하십시오.

V 9.0.2 내 로그 파일 시스템은 얼마나 커야 합니까?

큐 관리자에 필요한 로그 파일 시스템의 크기를 추정합니다.

큐 관리자가 로그를 기록할 수 있는 공간을 충분히 가질 수 있는 크기로 로그 파일 시스템의 크기를 설정하는 것이 중요합니다. 큐 관리자가 로그 파일 시스템을 완전히 채우면 FFDC가 기록되고 트랜잭션이 롤백되고 큐 관리자가 갑자기 종료됩니다.

로그를 위해 예약하는 디스크 공간의 크기는 활성 로그 크기 이상의 크기여야 합니다. 얼마나 더 커야 하는지는 다음에 의해 결정됩니다.

- 선택한 로그 유형(선형 또는 순환)
- 활성 로그의 크기(기본 파일, 보조 파일, 로그 파일 페이지)
- 선택한 로그 관리(수동, 자동 또는 아카이브)
- 오브젝트 손상 시 비상사태 계획

순환 로그를 선택하는 경우 로그 파일 시스템은 다음과 같아야 합니다.

```
LogFilesystemSize >= (PrimaryFiles + SecondaryFiles + 1) * LogFileSize
```

이를 통해 큐 관리자는 모든 기본 및 보조 파일에 기록할 수 있습니다. 예외적인 상황에서 큐 관리자는 보조 수를 벗어난 추가 익스텐트를 기록합니다. 선행 알고리즘에서는 이를 고려합니다.

선형 로그를 선택하는 경우 로그 파일 시스템은 활성 로그보다 훨씬 커야 합니다.

수동 로그 관리를 선택하는 경우 큐 관리자는 필요할 때 새 로그 익스텐트에 계속 기록하며 이 새 로그 익스텐트가 더 이상 필요하지 않을 때 이를 삭제하고 아카이브하는 것은 사용자의 책임입니다.

로그 파일 시스템이 얼마나 더 커야 하는지는 불필요하거나 비활성인 익스텐트를 삭제하기 위한 전략에 의해 주로 결정됩니다.

익스텐트가 비활성(재시작 복구를 위해 필요하지 않음) 상태가 되는 즉시 익스텐트를 아카이브하고 삭제하도록 결정하거나 불필요한 익스텐트(매체 복원 또는 재시작 복구를 위해 필요하지 않음)만 아카이브하고 삭제하도록 결정합니다.

불필요한 익스텐트만 아카이브하고 삭제하는 경우 손산된 오브젝트가 있으면 **MEDIALOG**가 앞으로 이동하지 않으므로 더 이상 불필요해지는 익스텐트가 없습니다. 오브젝트를 복구하는 등의 방식으로 문제점을 해결할 때까지 익스텐트 아카이브 및 삭제를 중지합니다.

워크로드를 중지하는 경우가 아니면 문제점을 해결하기 위해 걸리는 시간은 로그 파일 시스템의 크기에 따라 다릅니다. 따라서 선형 로깅을 사용할 때 넉넉한 로그 파일 시스템을 가지고 있는 것이 좋습니다.

선형 로그 및 자동 또는 아카이브 로그 관리를 선택하는 경우 큐 관리자는 로그 익스텐트를 재사용합니다.

재사용할 수 있는 로그 익스텐트에는 R문자가 접두부로 붙습니다. 매체 이미지가 기록되면, 불필요한 익스텐트가 아카이브될 때 큐 관리자가 해당 익스텐트를 재사용할 수 있습니다.

따라서 재사용 익스텐트는 매체 이미지 간 로그에 기록된 데이터 길이보다 짧습니다.

```
ReuseExtents <= LogDataLengthBetweenMediaImages
```

매체 이미지를 자동으로 기록하고 **IMGLOGLN**을 설정할 때 **IMGLOGLN**은 고정 최대값이 아니라 대상이므로 `LogDataLengthBetweenMediaImages`의 크기는 **IMGLOGLN**의 두 배일 수 있습니다.

매체 이미지를 수동으로 기록하거나 간격을 사용하여 자동으로 기록할 때 LogDataLengthBetweenMediaImages는 이미지 기록 간격 및 워크로드에 따라 결정됩니다.

활성 익스텐트 및 재사용 익스텐트 외에도 비활성 익스텐트(매체 복원을 위해서만 필요함) 및 불필요한 익스텐트(재시작 복구 또는 매체 복원을 위해 필요하지 않음)가 있습니다.

자동 또는 아카이브 로그 관리를 사용하는 경우 큐 관리자는 매체 복원을 위해 필요한 익스텐트를 재사용하지 않습니다. 따라서 비활성 익스텐트의 수는 매체 이미지를 기록하는 빈도와 매체 이미지를 수동으로 기록하는지 아니면 자동으로 기록하는지에 따라 결정됩니다.

IMGINTVL 및 **IMGLOGLN**은 매체 이미지 간 고정 최소값 또는 최대값이 아니라 대상입니다. 하지만 필요한 로그 파일 시스템의 최대 크기를 추정할 때는 자동 매체 이미지가 **IMGINTVL** 또는 **IMGLOGLN** 각각보다 두 배 이상 기록될 가능성이 없습니다.

자동 또는 아카이브 로그 관리를 사용하여 로그 파일 시스템의 크기를 조정하는 경우에는 큐 또는 기타 오브젝트가 손상될 때 발생하는 사항도 고려해야 합니다. 이 경우 큐 관리자는 손상된 오브젝트의 매체 이미지를 기록할 수 없으므로 **MEDIALOG**가 앞으로 이동하지 않습니다.

매체 복원을 위해 필요한 가장 오래된 익스텐트는 계속 필요하지만 재사용될 수 없으므로 워크로드가 계속되는 경우 비활성 로그는 제한 없이 증가합니다. 워크로드가 계속되는 경우에는 큐 관리자가 트랜잭션 롤백을 시작하고 갑자기 종료되기 전에 로그 파일 시스템이 완전히 채워질 때까지 문제점을 해결해야 합니다.

따라서 자동 및 아카이브 로그 관리의 경우:

```
LogFilesystemSize > (PrimaryFiles + SecondaryFiles +
(((TimeBetweenMediaImages *2) + TimeNeededToResolveDamagedObject) * ExtentsUsedPerHour))
* LogFilePages
```

참고: 선행 알고리즘에서는 아카이브 로그 관리를 위해 각각의 익스텐트가 매체 복원에 더 이상 필요하지 않는 즉시 각 익스텐트에 대해 **SET LOG ARCHIVED**가 호출된다고 가정합니다.

로그 관리

V 9.0.2 IBM MQ 9.0.2에서 제품은 자동 로그 관리 및 선형 로그의 자동 매체 복구를 지원합니다. 순환 로그는 거의 자체적으로 관리되지만 때때로 공간 문제점을 해결하기 위한 개입이 필요합니다.

순환 로깅의 경우 큐 관리자는 로그 파일에서 비워진 공간을 재확보합니다. 이 활동은 사용자에게 명백하지 않으며 할당된 공간은 빠르게 재사용되므로 사용한 디스크 공간의 크기가 감소되는 것을 일반적으로 알지 못합니다.

V 9.0.2 IBM MQ 9.0.2부터는 순환 로깅 사용 시 보조 파일을 삭제할 수 있습니다. 자세한 정보는 RESET QMGR **TYPE(REDUCELOG)** 을 참조하십시오.

선형 로깅의 경우에는 체크포인트가 장기간 사용되지 않은 경우 또는 장기 실행 트랜잭션이 오래 전에 로그 레코드를 기록한 경우 로그가 채워집니다. 큐 관리자는 첫 번째 문제점을 피하기에 충분히 자주 체크포인트를 사용하려고 시도합니다.

Multi 로그가 채워지면 AMQ7463 메시지가 발행됩니다. 또한 장기 실행 트랜잭션으로 인해 공간이 해제되지 않아 로그가 채워지면 AMQ7465 메시지가 발행됩니다.

로그 레코드 중 마지막 완료 체크포인트 이후에 기록된 로그 레코드와 활성 트랜잭션에 의해 기록된 로그 레코드만 큐 관리자 재시작에 필요합니다.

시간 경과에 따라 기록된 가장 오래된 로그 레코드는 큐 관리자 재시작을 위해 불필요하게 됩니다.

장기 실행 트랜잭션이 감지되면 활동이 비동기식으로 해당 트랜잭션을 롤백하도록 스케줄됩니다. 예상치 못한 이유로 해당 비동기 롤백이 실패하는 경우 일부 MQI 호출에서는 해당 상황에서 MQRC_RESOURCE_PROBLEM을 리턴합니다.

모든 인플라이트 트랜잭션을 커미트하거나 롤백하기 위해 공간이 예약되므로 MQCMIT 또는 MQBACK은 실패해서는 안 됩니다.

큐 관리자는 장기간 실행된 트랜잭션을 롤백합니다. 트랜잭션이 있는 애플리케이션이 이 방식으로 롤백되고 같은 트랜잭션에서 동기점을 지정하는 후속 MQPUT 또는 MQGET 조작을 수행할 수 없습니다.

하지만 수동으로 종료된 트랜잭션은 새 로그를 시작합니다. 새 로그 공간은 즉시 할당되는 반면 해제된 로그 공간을 비우는 데는 일정 시간이 걸립니다.

이 상태에서 동기점 아래에서 메시지를 넣거나 가져오려고 시도하면 MQRC_BACKED_OUT이 리턴됩니다. 그러면 애플리케이션이 MQRC_BACKED_OUT을 리턴하는 **MQCMIT** 또는 **MQBACK**을 발행하고 새 트랜잭션을 시작할 수 있습니다. 너무 많은 로그 공간을 이용하는 트랜잭션이 롤백된 경우 로그 공간이 해제되고 큐 관리자는 계속해서 정상적으로 작동합니다.

디스크가 가득 차면 발생하는 상황

큐 관리자 로깅 컴포넌트는 가득 찬 디스크 및 가득 찬 로그 파일을 처리할 수 있습니다. 로그가 포함된 디스크가 가득 차면 큐 관리자가 AMQ6709 메시지를 발행하고 오류 레코드가 기록됩니다.

로그 파일은 로그 레코드가 기록될 때 확장되지 않고 고정 크기로 작성됩니다. 이는 IBM MQ가 새 파일을 작성할 때에만 디스크 공간을 부족하게 할 수 있고 레코드를 로그에 기록할 때에는 공간을 부족하게 할 수 없음을 의미합니다. IBM MQ는 항상 기존 로그 파일에서 사용 가능한 공간의 양을 알고 그에 따라 파일 내 공간을 관리합니다.

V 9.0.2 IBM MQ 9.0.2부터, 선형 로깅을 사용할 때 다음과 같은 옵션을 사용할 수 있습니다.

- 로그 익스텐트의 자동 관리

  새 로그 속성에 대한 자세한 정보는 DISPLAY QMSTATUS를 참조하십시오.

  다음과 같은 명령 또는 해당 PCF 동등 항목도 참조하십시오.

  - RESET QMGR
  - 분산 플랫폼에 대한 SET LOG
- 매체 이미지 사용을 제어하는 옵션

  ALTER QMGR 명령 및 ALTER QUEUES에서 다음 항목에 대한 자세한 정보를 참조하십시오.

  - IMGINTVL
  - IMGLOGLN
  - IMGRCOVO
  - IMGRCOVQ
  - IMGSCHED

순환 로깅은 자원 문제점을 리턴합니다.

여전히 공간이 부족하면 큐 관리자 구성 파일의 로그 구성이 올바른지 확인하십시오. 로그가 사용 가능한 공간보다 커지지 않도록 1차 또는 보조 로그 파일의 수를 줄일 수 있습니다.

기존 큐 관리자의 로그 파일 크기는 대체할 수 없습니다. 큐 관리자는 모든 로그 익스텐트의 크기가 동일하도록 요구합니다.

로그 파일 관리

로그 파일에 충분한 공간을 할당합니다. 선형 로깅에서는 더 이상 필요하지 않은 오래된 로그 파일을 삭제할 수 있습니다.

순환 로깅에만 해당되는 정보

선형 로깅을 사용하는 경우 시스템 구성 시 로그 파일을 보유하기에 충분한 공간이 있는지 확인하십시오(100 페이지의 『IBM MQ의 로그 기본값』 및 107 페이지의 『큐 관리자 로그』 참조). 필요 시 작성될 보조 파일을 위한 공간을 포함하여 로그가 사용하는 디스크 공간의 양은 구성한 크기를 넘게 증가하지 않습니다.

선형 로깅에만 해당되는 정보

선형 로그를 사용하는 경우 데이터가 로깅될 때마다 로그 파일이 계속해서 추가되고 사용되는 디스크 공간의 양은 시간의 경과에 따라 증가합니다. 로깅되는 데이터의 비율이 높은 경우 새 로그 파일이 빠르게 디스크 공간을 사용합니다.

시간의 경과에 따라 선형 로그의 오래된 로그 파일이 큐 관리자를 재시작하거나 손상된 오브젝트의 매체 복구를 수행하는 데 더 이상 필요하지 않게 됩니다. 다음 방법에 따라 필요한 로그 파일이 결정됩니다.

로거 이벤트 메시지
중요한 이벤트가 발생할 때(예: 매체 이미지 기록) 로거 이벤트 메시지가 생성됩니다. 로거 이벤트 메시지의 컨텐츠는 큐 관리자 재시작 및 매체 복구를 위해 여전히 필요한 로그 파일을 지정합니다. 로거 이벤트 메시지에 대한 자세한 정보는 로거 이벤트를 참조하십시오.

큐 관리자 상태
MQSC 명령인 DISPLAY QMSTATUS 또는 PCF 명령인 Inquire Queue Manager Status를 실행하면 필요한 로그 파일의 세부사항을 포함한 큐 관리자 정보를 리턴합니다. MQSC 명령에 대한 자세한 정보는 스크립트 (MQSC) 명령을 참조하고 PCF 명령에 대한 정보는 관리 태스크 자동화를 참조하십시오.

큐 관리자 메시지
주기적으로 큐 관리자는 필요한 로그 파일을 표시하도록 한 쌍의 메시지를 발행합니다.

- AMQ7467I 메시지는 큐 관리자를 다시 시작하는 데 필요한 가장 오래된 로그 파일의 이름을 제공합니다. 이 로그 파일 및 모든 새 로그 파일은 큐 관리자 재시작 중 사용 가능해야 합니다.
- AMQ7468I 메시지는 매체 복구에 필요한 가장 오래된 로그 파일의 이름을 제공합니다.

"오래된" 및 "새로운" 로그 파일을 판별하려면 파일 시스템에서 적용된 수정 시간 보다는 로그 파일 번호를 사용하십시오.

두 유형의 로깅에 모두 해당되는 정보

큐 관리자 재시작에 필요한 로그 파일 및 활성 로그 파일만 온라인이어야 합니다. 비활성 로그 파일은 재해 복구용 테이프와 같은 아카이브 매체에 복사하고 로그 디렉토리에서는 제거할 수 있습니다. 매체 복구에 필요하지 않은 비활성 로그 파일은 불필요한 로그 파일로 간주할 수 있습니다. 더 이상 조작에 이용되지 않는 불필요한 로그 파일은 삭제할 수 있습니다.

필요한 로그 파일을 찾을 수 없는 경우 운영자 메시지 AMQ6767E가 발행됩니다. 로그 파일 및 모든 후속 로그 파일을 큐 관리자가 사용할 수 있게 하고 조작을 다시 시도하십시오.

자동으로 로그 익스텐트 정리 - 선형 로깅 전용

V 9.0.2 Multi

IBM MQ 9.0.2부터는 복구를 위해 더 이상 필요하지 않은 선형 로그 익스텐트의 자동 관리를 사용하는 옵션이 있습니다.

qm.ini 파일의 로그 스탠자에 있는 **LogManagement** 속성을 사용하거나 IBM MQ 탐색기를 사용하여 자동 관리를 설정합니다. 자세한 정보는 107 페이지의 『큐 관리자 로그』의 내용을 참조하십시오.

DISPLAY QMSTATUS의 LOG 매개변수에서 로그의 조작에 대한 자세한 내용을 참조하고 다음의 명령에서 로그 사용에 대해 참조하십시오.

- RESET QMGR
- SET LOG

자동으로 매체 이미지 기록 - 선형 로깅 전용

V 9.0.2

IBM MQ 9.0.2부터는 큐 관리자가 자동으로 매체 이미지를 기록하는지 여부를 제어하는 전체 스위치가 있으며 기본값은 스위치가 설정되지 않는 것입니다.

다음 큐 관리자 속성을 사용하여 자동 매체 이미지 발생 여부 및 프로세스의 빈도를 제어할 수 있습니다.

IMGSCHED
큐 관리자가 자동으로 매체 이미지를 기록하는지 여부

IMGINTVL
매체 이미지 기록 빈도(분)

IMGLOGLN
오브젝트의 이전 매체 이미지 이후 기록된 로그의 크기(MB)

하루 중에 워크로드가 매우 많은 위험한 시간이 있을 때 자동 매체 이미지 기록이 시스템 처리량에 영향을 주지 않게 하려는 경우에는 **IMGSCHED**(*MANUAL*)를 설정하여 일시적으로 자동 매체 이미지 기록을 끄길 원할 수 있습니다.

워크로드 중에 언제든지 **IMGSCHED**를 전환할 수 있습니다.

주의: 매체 이미지를 기록하지 않는 경우에는 **MEDIALOG**가 앞으로 이동하지 않으므로 익스텐트를 아카이브하거나 디스크 공간이 충분한지 확인해야 합니다.

기타 사용자 정의 오브젝트에 대한 자동 및 수동 매체 이미지도 제어할 수 있습니다.

- 인증 정보
- 채널
- 클라이언트 연결
- 리스너
- 이름 목록
- 프로세스
- 알리어스 큐
- 로컬 큐
- 서비스
- 주제

내부 시스템 오브젝트(예: 오브젝트 카탈로그 및 큐 관리자 오브젝트)의 경우 큐 관리자는 적절하게 매체 이미지를 자동으로 기록합니다.

속성에 대한 자세한 정보는 ALTER QMGR을 참조하십시오.

로컬 및 영구적 동적 큐에 대해 자동 및 수동 매체 이미지를 사용 또는 사용 안함으로 설정할 수 있습니다. 이를 위해서는 **IMGRCOVQ** 큐 속성을 사용하십시오.

IMGRCOVQ 속성에 대한 자세한 정보는 ALTER QUEUES를 참조하십시오.

참고사항:

1. 선형 로깅을 사용하는 경우에만 매체 이미지가 지원됩니다. 자동 매체 이미지를 사용으로 설정했지만 순환 로그 기록을 사용 중인 경우에는 오류 메시지가 발행되고 큐 관리자의 자동 매체 이미지 속성이 사용 안함으로 설정됩니다.
2. 자동 매체 이미지를 사용으로 설정했지만 빈도(로그의 크기(MB) 또는 시간(분))를 지정하지 않은 경우에는 오류 메시지가 발행되고 자동 매체 이미지가 기록되지 않습니다.
3. 원하는 경우 **IMGSCHED**(*AUTO*)를 설정한 경우 rcdmqimg 를 사용하여 매체 이미지를 수동으로 기록할 수 있습니다.

   그러면 엔터프라이즈에 적합한 시기(예: 시스템이 대기 상태일 때)에 매체 이미지를 기록할 수 있습니다. 이 수동 매체 이미지를 기록하면 다음 자동 매체 이미지가 기록되기 전의 간격 및 로그 길이가 재설정되므로 자동 매체 이미지 기록에서는 이 수동 매체 이미지를 고려합니다.
4. IBM MQ 9.0.2에서는 큐 관리자가 매체 이미지에서 비지속 메시지가 아니라 지속 메시지만 기록합니다. 그러면 IBM MQ 9.0.2 이상으로 마이그레이션할 때 매체 이미지의 크기를 줄일 수 있습니다.

IMGLOGLN 및 IMGINTVL 설정 방법 결정

V 9.0.2

큐 관리자가 매체 이미지를 기록하는 데 시간의 일부만 사용하도록 충분히 크지만 다음과 같이 충분히 작게 **IMGLOGLN** 및 **IMGINTVL**을 설정하십시오.

- 손상된 오브젝트가 합리적인 시간 내에 복구될 수 있고
- 공간 부족 없이 로그가 디스크에 맞을 정도로 작게

IMGLOGLN을 설정하는 경우에는 **IMGLOGLN**을 큐에 있는 데이터 크기의 몇 배 및 워크로드 데이터 속도의 몇 배로 설정하는 것이 좋습니다. **IMGLOGLN**을 더 크게 설정할수록 큐 관리자가 매체 이미지를 기록하는 데 걸리는 시간이 줄어듭니다.

마찬가지로 **IMGINTVL**을 설정하는 경우에는 **IMGINTVL**을 큐 관리자가 매체 이미지를 기록하는 데 걸리는 시간의 몇 배로 설정하는 것이 좋습니다. 하나의 매체 이미지를 수동으로 기록하여 매체 이미지를 기록하는 데 걸리는 시간을 파악할 수 있습니다.

마지막 매체 이미지 이후의 모든 익스텐트를 재실행해야 하기 때문에 **IMGLOGLN** 및 **IMGINTVL**을 너무 크게 설정하면 손상된 오브젝트를 복구하는 데 시간이 매우 오래 걸립니다.

손상된 오브젝트를 복구하는 데 걸리는 최대 시간이 용인 가능한 수준이 되도록 **IMGLOGLN** 및 **IMGINTVL**을 충분히 작게 설정하십시오.

매체 이미지는 매우 드물게 기록되므로 **IMGLOGLN** 및 **IMGINTVL**을 매우 크게 설정하면 로그가 매우 크게 증가합니다.

주의: 로그 파일 시스템이 완전히 채워지면 워크로드가 백아웃되므로 이 크기의 로그가 로그 파일 시스템에 적합한 크기인지 확인하십시오.

IMGINTVL과 **IMGLOGLN**을 모두 설정할 수 있습니다. 이는 과도한 워크로드 수행 중에는 정기적으로 자동 매체 이미지를 기록(**IMGLOGLN**에 의해 제어됨)하지만 워크로드가 매우 적을 때는 가끔 기록(**IMGINTVL**에 의해 제어됨)하게 할 때 유용합니다.

IMGINTVL 및 **IMGLOGLN**은 자동 매체 이미지가 기록되는 간격 및 로그 데이터 길이에 대한 대상입니다.

이 속성은 고정 최대값 또는 최소값으로 표시되어서는 안 됩니다. 실제로 큐 관리자는 매우 적절한 시기라고 인식하는 경우 자동 매체 이미지를 더 빨리 스케줄하도록 결정합니다.

- 큐가 비어 있으므로 매체 이미지를 기록하는 것이 성능 면에서 가장 효율적입니다.
- 한동안 매체 이미지가 기록되지 않았습니다.

가끔 자동 매체 이미지 간 간격이 **IMGINTVL** 및 **IMGLOGLN** 중 하나 또는 둘 다보다 길 수 있습니다.

큐에 있는 데이터의 양이 **IMGLOGLN**에 근접하는 경우 매체 이미지 간 간격이 **IMGLOGLN**보다 클 수 있습니다. 매체 이미지를 기록하는 데 걸리는 시간이 **IMGINTVL**과 거의 비슷한 경우 매체 이미지 간 간격은 **IMGINTVL**보다 클 수 있습니다.

큐 관리자가 매체 이미지를 기록하는 데 상당한 시간을 사용하므로 이는 바람직하지 않습니다.

자동 매체 이미지 기록 사용 시 큐 관리자는 각 오브젝트 및 큐에 대해 개별적으로 매체 이미지를 기록하므로 큐 관리자는 각 오브젝트에 대해 별도로 이미지 간 간격 및 로그 길이를 추적합니다.

모든 오브젝트에 대한 매체 이미지를 동시에 기록하는 대신 시간 경과에 따라 점진적으로 시차를 두고 매체 이미지를 기록하게 됩니다. 이 시차는 매체 이미지 기록이 성능에 미치는 영향을 분산시키므로 자동 매체 이미지 기록 사용이 수동 기록보다 나은 또 다른 장점입니다.

수동으로 매체 이미지 기록 - 선형 로깅 전용

V 9.0.1

큐의 매체 이미지를 기록하는 데는 해당 큐의 모든 지속 메시지를 로그에 기록하는 것이 포함됩니다. 많은 볼륨의 메시지 데이터가 포함된 큐의 경우에는 여기에 많은 양의 데이터를 로그에 기록하는 것이 포함되며 이 프로세스는 발생하는 동안 시스템의 성능에 영향을 미칠 수 있습니다.

다른 오브젝트의 매체 이미지에는 사용자 데이터가 포함되어 있지 않으므로 다른 오브젝트의 매체 이미지를 기록하는 것은 상대적으로 빠르게 수행될 수 있습니다.

프로세스가 최대 워크로드를 방해하지 않도록 큐의 매체 이미지 기록 시기를 주의해서 결정해야 합니다.

매체 복원을 위해 필요한 가장 오래된 로그 익스텐트를 업데이트하기 위해 모든 오브젝트의 매체 이미지를 정기적으로 기록해야 합니다.

큐가 비어 있으면 로그에 기록된 메시지 데이터가 없으므로 큐가 비어 있을 때가 큐의 매체 이미지를 기록할 적절한 시기입니다. 반대로 적절하지 않은 시기는 큐가 매우 크거나 매우 많은 메시지가 큐에 있을 때입니다.

큐의 매체 이미지를 기록하는 좋은 시점은 시스템이 대기 상태일 때입니다. 반면 나쁜 시점은 최대 워크로드 시점입니다. 예를 들어, 워크로드가 자정에 항상 적은 경우 매일 자정에 매체 이미지를 기록하도록 결정할 수 있습니다.

큐의 레코드를 삽입하면 성능 영향이 미칠 수 있으므로 영향을 최소화합니다. 매체 복원을 위해 필요한 로그 익스텐트 수가 증가하므로 매체 이미지를 마지막으로 기록한 후 경과한 시간이 길수록 매체 이미지를 기록하는 작업이 중요해집니다.

참고: 매체 복구를 수행할 때 필요한 모든 로그 파일은 로그 파일 디렉토리에서 동시에 사용 가능해야 합니다. 필요한 모든 로그 파일을 보유하기 위해 디스크 공간이 부족해지는 것을 피하려면 복구가 필요할 수도 있는 오브젝트의 정기적 매체 이미지를 제거해야 합니다.

예를 들어, 큐 관리자에서 모든 오브젝트의 매체 이미지를 기록하려면 다음 예에 표시된 대로 **rcdmqimg** 명령을 실행하십시오.

Windows **Windows**

```
rcdmqimg -m QMNAME -t all *
```

Linux UNIX **UNIX and Linux**

```
rcdmqimg -m QMNAME -t all "*"
```

rcdmqimg를 실행하면 매체 로그 순서 번호(LSN)가 앞으로 이동합니다. 로그 순서 번호에 대한 자세한 내용은 498 페이지의 『dmpmqlog 명령을 사용하여 로그의 컨텐츠 덤프』의 내용을 참조하십시오. **rcdmqimg**는 자동으로 실행되지 않으므로 수동으로 실행되거나 작성한 자동 태스크로부터 실행되어야 합니다. 이 명령에 대한 자세한 정보는 rcdmqimg 및 dmpmqlog를 참조하십시오.

참고: 메시지 AMQ7467 및 AMQ7468은 rcdmqimg 명령 실행 시 발행될 수도 있습니다.

부분 매체 이미지

V 9.0.2

각각의 메시지가 상대적으로 짧은 시간 동안 큐에 머물도록 가까운 미래에 이용될 것으로 예상되는 데이터에 대해서만 IBM MQ 메시지를 사용하는 것이 좋습니다.

반대로 데이터베이스와 같이 데이터를 장기간 저장하기 위해 IBM MQ 메시지를 사용하는 것은 바람직하지 않습니다.

또한 큐의 크기를 상대적으로 작게 유지하는 것이 좋으며 메시지가 큐에 장기간 있었던 크기가 큰 큐를 유지하는 것은 바람직하지 않습니다.

이 지침을 따라주면 큐 관리자가 매체 이미지의 자동 기록 성능을 최적화할 수 있습니다.

비어 있는 큐의 매체 이미지를 기록하는 것은 성능 관점에서 매우 효율적인 반면 많은 양의 데이터가 있는 큐의 매체 이미지를 기록하면 해당 데이터가 모두 매체 이미지의 로그에 기록되기 때문에 매우 비효율적입니다.

최근 메시지가 배치된 크기가 작은 큐의 경우 큐 관리자가 추가적인 최적화를 수행할 수 있습니다.

큐에 현재 있는 모든 메시지가 최근에 큐에 배치된 경우 큐 관리자는 모든 메시지가 배치되기 직전(복구 지점)에 매체 이미지를 대신 기록할 수 있으므로 비어 있는 큐의 이미지를 기록할 수 있습니다. 이 프로세스는 성능 측면에서 비용이 매우 낮습니다.

복구 지점에 큐에 있었던 모든 메시지가 차후에 사라진 경우 해당 메시지는 더 이상 큐에 없으므로 해당 메시지는 매체 이미지에 기록하지 않아도 됩니다.

이를 부분 매체 이미지라고 합니다. 그런 다음 큐를 복구해야 하는 예상 밖의 이벤트가 발생하는 경우 마지막 매체 이미지 이후 이 큐와 관련된 모든 로그 레코드가 재실행되므로 최근에 배치된 메시지를 모두 복원합니다.

복구 지점에 큐에 몇몇 메시지가 있었고 이 메시지가 현재 큐에 있어 부분 매체 이미지에 기록되어야 하는 경우에도 더 작은 이 부분 매체 이미지를 기록하는 것이 모든 메시지의 전체 매체 이미지를 기록하는 것보다 여전히 더 효율적입니다.

짧은 시간 동안 메시지가 큐에 머무르도록 하면 매체 이미지 자동 기록의 성능을 향상시킬 수 있습니다.

불필요한 로그 파일 판별 - 선형 로깅 전용
순환 로깅의 경우 로그 디렉토리에서 데이터를 삭제하지 마십시오. 선형 로그 파일을 관리할 때 삭제하거나 아카이브할 수 있는 파일을 확인하는 것이 중요합니다. 이 정보는 이 결정을 내리는 데 도움을 줍니다.

"오래된" 로그 파일을 판별하는 데 파일 시스템의 수정 시간을 사용하지 마십시오. 로그 파일 번호만 사용하십시오. 큐 관리자의 로그 파일 사용은 필요 전 로그 파일 사전 할당 및 형식화와 같은 복잡한 규칙을 따릅니다. 수정 시간을 사용하여 상대적 연령을 판별하려고 시도할 경우 잘못 유도할 수 있는 수정 시간이 포함된 로그 파일이 표시될 수 있습니다.

필요한 가장 오래된 로그 파일을 판별하기 위해 세 가지 위치를 사용할 수 있습니다.

- DISPLAY QMSTATUS 명령
- 로거 이벤트 메시지 및
- 오류 로그 메시지

DISPLAY QMSTATUS 명령의 경우 필요한 가장 오래된 로그 익스텐트를 판별하려면 다음을 수행하십시오.

- 큐 관리자를 재시작한 후 `DISPLAY QMSTATUS RECLOG` 명령을 실행하십시오.
- 매체 복원을 수행한 후 `DISPLAY QMSTATUS MEDIALOG` 명령을 실행하십시오.
- V 9.0.2 아카이브 알림의 이름을 판별하고 `DISPLAY QMSTATUS ARCHLOG`명령을 실행하십시오.

V 9.0.2 **RESET QMGR TYPE(REDUCELOG)** 명령을 실행하여 선형 로깅 사용 시 보조 로그 익스텐트의 수를 줄일 수 있습니다.

일반적으로 파일 번호가 낮을 수록 오래된 로그를 의미합니다. 10년 동안 하루에 3000개의 로그 파일 순서 중 매우 높은 로그 파일 턴오버가 있는 경우가 아니면 9 999 999에서 줄을 바꾸는 숫자를 제공하지 않아도 됩니다. 이 경우 RECLOG 값 미만의 번호를 가지는 로그 파일을 아카이브할 수 있고 RECLOG 및 MEDIALOG 값 미만의 번호를 가지는 로그 파일을 삭제할 수 있습니다.

주의: 로그 파일은 줄 바꾸기를 수행하므로 9 999 999 다음의 숫자는 영(0)입니다.

로그 파일 위치
로그 파일의 위치를 선택할 때 디스크 공간 부족으로 인해 IBM MQ가 새 로그를 형식화하는 데 실패할 경우 조작이 심각하게 영향을 받음을 기억하십시오.

순환 로깅을 사용하는 경우 적어도 구성된 1차 로그 파일에 대해서만이라도 드라이브에 충분한 공간이 있는지 확인하십시오. 로그의 크기가 커져야 할 경우 필요한 보조 로그 파일 1개에 대한 공간만이라도 남겨두십시오.

선형 로그를 사용하는 경우 상당히 많은 공간을 허용하십시오. 로그가 이용하는 공간은 데이터가 로그됨에 따라 지속적으로 늘어납니다.

큐 관리자 데이터와 별도의 디스크 드라이브에 로그 파일을 배치해야 합니다.

이 디바이스의 데이터 무결성이 무엇보다 중요하므로 내장 중복성을 감안해야 합니다.

로그 파일을 미러링된 배열의 다중 디스크 드라이브에 배치할 수도 있습니다. 이는 로그를 포함하는 드라이브의 실패로부터 보호합니다. 미러링 없이 IBM MQ 시스템의 마지막 백업으로 돌아가도록 강제 실행될 수 있습니다.

복구에 로그 사용

로그의 정보를 사용하여 오류 복구에 도움을 받을 수 있습니다.

데이터가 손상될 수 있는 여러 가지 방법이 있습니다. IBM MQ을(를) 사용하면 다음을 복구할 수 있습니다.

- 손상된 데이터 오브젝트
- 시스템에서 전원 손실
- 통신 실패

이 절에서는 이러한 문제점으로부터 복구하는 데 로그가 사용되는 방식을 알아봅니다.

전원 유실 또는 통신 장애로부터 복구

IBM MQ는 통신 실패 또는 전원 손실 둘 다로부터 복구할 수 있습니다. 또한 파일의 의도치 않은 삭제와 같은 다른 유형의 문제점으로부터도 복구할 수 있습니다.

통신에 실패하는 경우 지속 메시지는 수신하는 애플리케이션에 의해 제거될 때까지 큐에 남아 있습니다. 메시지가 전송되고 있는 경우 성공적으로 전송될 때까지 이 메시지는 전송 큐에 남아 있습니다. 통신 실패로부터 복구하기 위해 일반적으로 실패한 링크를 사용하여 채널을 재시작할 수 있습니다.

전원이 손실되면 큐 관리자가 재시작될 때 IBM MQ는 큐를 실패 시 이 큐의 커미트된 상태로 복원합니다. 이는 지속 메시지가 손실되지 않도록 합니다. 비지속 메시지가 버려지고 IBM MQ가 갑자기 중지되면 이 비지속 메시지는 없어집니다.

손상된 오브젝트 복구

IBM MQ 오브젝트가 의도치 않은 손상 등으로 인해 사용할 수 없게 되는 상황이 있습니다. 그러면 전체 시스템 또는 그 일부분을 복구해야 합니다. 필요한 조치는 손상이 감지되는 시간, 선택한 로그 메소드가 매체 복구를 지원하는지 여부 및 손상된 오브젝트에 따라 다릅니다.

매체 복구

V 9.0.2 IBM MQ 9.0.2부터 선형 로깅 큐 관리자의 경우 복구 가능한 오브젝트에 대해서만 매체 이미지가 기록될 수 있습니다. 예를 들어, **IMGRCOVO** 및 **IMGRCOVQ** 옵션을 고려해야 합니다.

V 9.0.2 마찬가지로 선형 로깅 큐 관리자의 해당 매체 이미지에서 복구 가능한 매체로 정의된 오브젝트의 서브세트만 복구할 수 있습니다. 복구 가능한 매체로 정의되지 않은 오브젝트가 손상되는 경우 해당 오브젝트에 대한 옵션은 순환 로깅 큐 관리자에 대한 옵션과 동일합니다.

매체 복구는 선형 로그에 기록된 정보로부터 오브젝트를 다시 작성합니다. 예를 들어, 오브젝트 파일이 의도치 않게 삭제되거나 다른 이유로 사용할 수 없게 되는 경우 매체 복구가 이 파일을 다시 작성할 수 있습니다. 오브젝트의 매체 복구에 필요한 로그의 정보는 *media image*라고 합니다.

매체 이미지는 오브젝트 자체를 다시 작성하는 데 사용할 수 있는 오브젝트 이미지가 포함된 일련의 로그 레코드입니다.

오브젝트를 다시 작성하는 데 필요한 첫 번째 로그 레코드는 매체 복구 레코드라고 합니다. 이 로그 레코드는 오브젝트에 대한 최신 매체 이미지의 시작입니다. 각 오브젝트의 매체 복구 레코드는 체크포인트 중에 기록되는 정보 조각 중 하나입니다.

오브젝트가 그 매체 이미지로부터 다시 작성될 때 마지막 이미지를 제거한 후 오브젝트에 대해 수행된 업데이트를 설명하는 로그 레코드를 재실행하는 데에도 필요합니다.

예를 들어, 지속 메시지를 큐에 넣기 전에 제거된 큐 오브젝트의 이미지가 있는 로컬 큐를 고려해 보십시오. 오브젝트의 최근 이미지를 다시 작성하기 위해 이미지 자체를 재실행하는 것 외에도 큐에 메시지를 넣은 것을 기록하는 로그 입력 항목을 재실행할 필요가 있습니다.

오브젝트가 작성될 때 기록된 로그 레코드는 오브젝트를 완전히 다시 작성하기 위한 충분한 정보를 포함합니다. 이러한 레코드는 오브젝트의 첫 번째 매체 이미지를 구성합니다. 그런 다음 종료될 때마다 큐 관리자는 다음과 같이 매체 이미지를 자동으로 기록합니다.

- 로컬이 아닌 모든 프로세스 오브젝트와 큐의 이미지
- 비어 있는 로컬 큐의 이미지

또한 rcdmqimg에 설명된 **rcdmqimg** 명령을 사용하여 수동으로 매체 이미지를 기록할 수 있습니다. 이 명령은 IBM MQ 오브젝트의 매체 이미지를 기록합니다.

V 9.0.2 **IMGSCHED***(AUTO)* 가 설정된 경우 큐 관리자는 매체 이미지를 자동으로 기록합니다. 자세한 정보는 **IMGINTVL** 및 **INGLOGLN**에 대한 정보는 ALTER QMGR 을 참조하십시오.

매체 이미지 작성이 완료되면 매체 이미지를 보유하는 로그와 이 시간 후에 작성된 모든 로그만 손상된 오브젝트를 다시 작성한 데 필요합니다. 매체 이미지를 작성하는 것의 이점은 사용 가능한 여유 스토리지의 양 및 로그 파일이 작성되는 속도와 같은 요소에 따라 다릅니다.

매체 이미지로부터 복구

큐 관리자는 큐 관리자를 시작하는 동안 매체 이미지에서 일부 오브젝트를 자동으로 복구합니다. 또한 큐 관리자가 마지막으로 종료될 때 완료되지 않은 트랜잭션과 관련되어 있고 재시작 처리 중에 손상된 경우 큐를 자동으로 복구합니다.

IBM MQ 오브젝트를 다시 작성하기 위해 로그에서 레코드를 재실행하는 **rcrmqobj** 명령을 사용하여 다른 오브젝트를 수동으로 복구해야 합니다. 이미지가 저장된 시간과 다시 작성 명령이 발행된 시간 사이에 적용 가능한 모든 로그 이벤트와 함께 로그에서 찾은 최근 이미지로부터 이 오브젝트가 다시 작성됩니다. IBM MQ 오브젝트가 손상되게 되면 수행할 수 있는 유효한 조치는 해당 오브젝트를 삭제하거나 이 메소드를 통해 다시 작성하는 것뿐입니다. 이 방식으로는 비지속 메시지를 복구할 수 없습니다.

rcrmqobj 명령의 추가 세부사항은 rcrmqobj를 참조하십시오.

오브젝트의 매체 복구를 시도할 때 매체 복구 레코드 및 모든 후속 로그 파일을 포함하는 로그 파일이 로그 파일 디렉토리에서 사용 가능해야 합니다. 필요한 파일을 찾을 수 없는 경우 운영자 메시지 AMQ6767이 발행되고 매체 복구 조작이 실패합니다. 다시 작성하려는 오브젝트의 정기적 매체 이미지를 제거하지 않을 경우 오브젝트를 다시 작성하는 데 필요한 모든 로그 파일을 보유하기에 디스크 공간이 충분하지 않을 수 있습니다.

존재하는 오브젝트 파일

V 9.0.1

큐 관리자는 **runmqsc**에 정의된 오브젝트의 속성을 디스크의 파일에 저장합니다. 이 오브젝트 파일은 큐 관리자의 데이터 디렉토리 아래에 있는 하위 디렉토리에 있습니다.

예를 들어, UNIX 및 Linux 플랫폼에서는 채널이 `/var/mqm/qmgrs/`*`qmgr`*`/channel`에 저장됩니다.

이 오브젝트 파일의 데이터는 오브젝트의 매체 이미지입니다. 이 오브젝트 파일이 삭제 또는 손상된 경우 해당 파일에 저장된 오브젝트가 손상됩니다. 선형 로깅 큐 관리자를 사용하면 rcrmqobj 명령을 사용하여 손상된 오브젝트를 로그에서 복구할 수 있습니다.

대부분의 오브젝트 파일에는 오브젝트의 속성이 포함되므로 채널 파일에는 채널의 속성이 포함됩니다. 예외사항은 다음과 같습니다.

- 카탈로그

  오브젝트 카탈로그는 모든 유형의 모든 오브젝트를 카탈로그화하고 `qmanager/QMQMOBJCAT`에 저장되어 있습니다.
- 동기화파일

  동기화 파일에는 모든 채널과 연관된 내부 상태 데이터가 포함되어 있습니다.
- 큐

  큐 파일에는 해당 큐의 메시지 뿐만 아니라 해당 큐의 속성도 모두 포함됩니다.

runmqsc 또는 IBM MQ 탐색기에 표시되는 카탈로그 또는 동기화 파일 오브젝트는 없습니다.

카탈로그 및 큐 관리자를 기록할 수는 있지만 복구할 수는 없습니다. 이 오브젝트가 손상되고 큐 관리자가 사전에 종료될 경우 재시작시 해당 오브젝트가 자동으로 복구됩니다.

기록 또는 복구할 오브젝트에 구독이 나열되지 않습니다. 지속 가능 구독이 시스템 큐에 저장되기 때문입니다. 지속 가능 구독을 기록하거나 복구할 경우 대신 SYSTEM.DURABLE.SUBSCRIBER.QUEUE를 기록하거나 복구하십시오.

시동 중 손상된 오브젝트 복구

시동 중 큐 관리자가 손상된 오브젝트를 발견하는 경우에 큐 관리자가 취하는 조치는 오브젝트의 유형 및 큐 관리자가 매체 복구를 지원하도록 구성되는지 여부에 따라 다릅니다.

큐 관리자 오브젝트가 손상되는 경우 오브젝트를 복구할 수 없는 한 큐 관리자는 시작할 수 없습니다. 큐 관리자가 선형 로그로 구성되어 매체 복구를 지원하는 경우 IBM MQ는 자동으로 해당 매체 이미지로부터 큐 관리자를 다시 작성하려고 시도합니다. 선택한 로그 메소드가 매체 복구를 지원하지 않는 경우 큐 관리자의 백업을 복원하거나 큐 관리자를 삭제할 수 있습니다.

큐 관리자가 중지될 때 활성인 트랜잭션이 있었다면 이러한 트랜잭션 내에 넣거나 가져온 지속적이고 커미트되지 않은 메시지가 포함된 로컬 큐도 큐 관리자를 성공적으로 시작하기 위해 필요합니다. 로컬 큐 중 손상된 것으로 확인되는 로컬 큐가 있고 큐 관리자가 매체 복구를 지원하는 경우 이 큐 관리자는 해당 매체 이미지로부터 손상된 로컬 큐를 다시 작성하려고 시도합니다. 큐 중 복구할 수 없는 큐가 있을 경우 IBM MQ를 시작할 수 없습니다.

매체 복구를 지원하지 않는 큐 관리자에서 시동 처리 중 커미트되지 않은 메시지가 포함된 손상된 로컬 큐가 발견되면 큐가 손상된 오브젝트로 표시되고 이 큐에 있는 커미트되지 않은 메시지는 무시됩니다. 이 상황은 그러한 큐 관리자에서 손상된 오브젝트의 매체 복구가 불가능하기 때문이고 남은 조치는 해당 오브젝트를 삭제하는 것뿐입니다. 손상을 보고하기 위해 메시지 AMQ7472가 발행됩니다.

다른 때의 손상된 오브젝트 복구

오브젝트의 매체 복구는 시동 중에만 자동입니다. 다른 때에는 오브젝트 손상이 감지되면 메시지 AMQ7472이 발행되고 이 오브젝트를 사용하는 대부분의 조작은 실패합니다. 큐 관리자가 시작된 후 언제든지 큐 관리자 오브젝트가 손상되면 큐 관리자가 선제 종료를 수행합니다. 오브젝트가 손상된 경우 이를 삭제할 수 있습니다. 또는 큐 관리자가 선형 로그를 사용하는 경우 `rcrmqobj` 명령을 사용하여 해당 매체 이미지로부터 이 오브젝트 복구를 시도하십시오(추가 세부사항은 rcrmqobj 참조).

V 9.0.2 큐(또는 기타 오브젝트)가 손상된 경우에는 **MEDIALOG**가 앞으로 이동하지 않습니다. 왜냐하면 **MEDIALOG**는 매체 복원을 위해 필요한 가장 오래된 익스텐트이기 때문입니다. 워크로드가 계속되고 있는 경우 **CURRLOG**는 계속 앞으로 이동하므로 새 익스텐트가 작성됩니다. 구성 (**LogManagement** 설정 포함) 에 따라 로그 파일 시스템을 채우기 시작할 수 있습니다. 로그 파일 시스템이 완전히 채워지면 트랜잭션이 롤백되고 큐 관리자가 갑자기 종료됩니다. 따라서 큐가 손상되면 큐 관리자가 종료되기 전에 조치를 취할 수 있는 시간이 제한됩니다. 가질 수 있는 시간은 워크로드로 인해 큐 관리자가 새 익스텐트를 작성하게 되는 비율과 로그 파일 시스템에 있는 여유 공간의 크기에 따라 다릅니다.

V 9.0.2 수동 로그 관리를 사용 중인 경우에는 재시작 복구를 위해 필요하지 않은 익스텐트를 아카이브한 후 해당 익스텐트가 매체 복원을 위해 계속 필요한 경우에도 해당 익스텐트를 로그 파일 시스템에서 삭제합니다. 이는 필요할 때 아카이브에서 복원할 수 있는 경우 승인 가능합니다. 이 정책은 큐가 손상되고 **MEDIALOG**가 앞으로 이동을 중지할 때 로그 파일 시스템을 채우지 않습니다. 그러나 다시 시작 또는 매체 복원을 위해 필요하지 않은 익스텐트만 아카이브하고 삭제하는 경우에는 큐가 손상되는 경우 로그 파일 시스템이 채우기를 시작합니다.

V 9.0.2 자동 또는 아카이브 로그 관리를 사용하는 경우 큐 관리자는 매체 복원을 위해 여전히 필요한 익스텐트를 아카이브하고 SET LOG ARCHIVED를 사용하여 큐 관리자에게 알린 경우에도 해당 익스텐트를 재사용하지 않습니다. 따라서 큐가 손상되면 로그 파일 시스템이 채우기를 시작합니다.

V 9.0.2 큐가 손상되면 OBJECT DAMAGED FFDC가 작성되고 **MEDIALOG**가 앞으로 이동을 중지합니다. 손상된 오브젝트는 FFDC에서 식별되거나 **runmqsc**에서 상태를 표시할 때 가장 오래된 **MEDIALOG**를 가진 오브젝트이기 때문에 식별될 수 있습니다.

V 9.0.2 로그 파일 시스템이 채워질 때 로그 파일 시스템이 채워지고 있기 때문에 워크로드가 백아웃되는 것을 걱정하는 경우 오브젝트 복구 또는 워크로드 정지를 수행하면 이 문제가 발생하지 않습니다.

IBM MQ 로그 파일 보호

큐 관리자가 실행되고 있을 때에는 로그 파일을 건드리지 않습니다. 복구가 불가능할 수 있습니다. superuser 또는 mqm 권한을 사용하여 의도치 않게 수정하지 않도록 로그 파일을 보호하십시오.

IBM MQ 큐 관리자가 실행 중인 경우 활성 로그 파일을 수동으로 제거하지 마십시오. 큐 관리자가 다시 시작하는 데 필요한 로그 파일을 사용자가 실수로 삭제하면, IBM MQ는 오류를 발행하지 **않고** 지속적 메시지를 포함하는 데이터를 계속 처리합니다. 큐 관리자는 정상적으로 종료되지만 재시작에 실패할 수 있습니다. 그러면 메시지의 복구가 불가능해집니다.

활성 큐 관리자가 사용하고 있는 로그를 삭제할 수 있는 권한이 있는 사용자는 다른 중요한 큐 관리자 자원(예: 큐 파일, 오브젝트 카탈로그 및 IBM MQ 실행 파일)을 삭제할 수 있는 권한도 가집니다. 따라서 이러한 사용자는 미숙함 등으로 인해 IBM MQ가 자체 보호할 수 없는 방식으로 실행 중이거나 중단된 큐 관리자에게 손상을 입힐 수 있습니다.

superuser 또는 mqm 권한을 부여할 때 주의하십시오.

dmpmqlog 명령을 사용하여 로그의 컨텐츠 덤프

`dmpmqlog` 명령을 사용하여 큐 관리자 로그의 컨텐츠를 덤프하는 방법입니다.

`dmpmqlog` 명령을 사용하여 큐 관리자 로그의 컨텐츠를 덤프하십시오. 기본적으로 모든 활성 로그 레코드가 덤프됩니다. 즉, 이 명령은 로그의 헤드(일반적으로 마지막으로 완료된 체크포인트의 시작)부터 덤프를 시작합니다.

이 로그는 일반적으로 큐 관리자가 실행되고 있지 않을 때에만 덤프할 수 있습니다. 큐 관리자는 종료 중 체크포인트를 사용하기 때문에 로그의 활성 부분은 일반적으로 적은 수의 로그 레코드를 포함합니다. 그러나 `dmpmqlog` 명령을 사용하면 다음 옵션 중 하나를 사용하여 덤프의 시작 위치를 변경함으로써 보다 많은 로그 레코드를 덤프할 수 있습니다.

- 로그의 베이스부터 덤프를 시작하십시오. 로그의 베이스는 로그의 헤드를 포함하는 로그 파일에서 첫 번째 로그 레코드입니다. 이 경우에 덤프되는 추가 데이터의 양은 로그 파일에서의 로그 헤드 위치에 따라 다릅니다. 로그 헤드가 로그 파일의 시작부 가까이에 있으면 적은 양의 추가 데이터만 덤프됩니다. 이 헤드가 로그 파일의 끝부분에 가까이 있으면 훨씬 더 많은 데이터가 덤프됩니다.
- 덤프의 시작 위치를 개별 로그 레코드로 지정하십시오. 각 로그 레코드는 고유 로그 순서 번호*(LSN)*로 식별됩니다. 순환 로깅의 경우 이 시작 로그 레코드는 로그의 베이스 앞에 있을 수 없습니다. 이 제한은 선형 로깅에는 적용되지 않습니다. 명령을 실행하기 전에 비활성 로그 파일을 복원해야 될 수도 있습니다. 이전 `dmpmqlog` 출력에서 가져온 유효한 LSN을 시작 위치로 지정해야 합니다.

  예를 들어, 선형 로깅에서는 마지막 `dmpmqlog` 출력으로부터 `nextlsn`을 지정할 수 있습니다. `nextlsn`은 `Log File Header`에 표시되고 기록될 다음 로그 레코드의 LSN을 표시합니다. 이를 로그가 마지막으로 덤프된 이후 기록된 모든 로그 레코드를 형식화하기 위한 시작 위치로 사용하십시오.
- **선형 로깅에만 해당됨**, 임의의 제공된 로그 파일 익스텐트에서 로그 레코드 형식화를 시작하도록 dmpmqlog를 지시할 수 있습니다. 이 경우 dmpmqlog는 이 로그 파일 및 활성 로그 파일과 같은 디렉토리에 있는 각각의 연속 로그 파일을 찾을 것으로 예상합니다. 이 옵션은 dmpmqlog가 로그의 베이스 앞 로그 레코드에 액세스할 수 없는 순환 로그에는 적용되지 않습니다.

`dmpmqlog` 명령의 출력은 `Log File Header` 및 일련의 형식화된 로그 레코드입니다. 큐 관리자는 여러 로그 레코드를 사용하여 해당 데이터에 변경사항을 기록합니다.

형식화되는 정보 중 일부는 내부 전용입니다. 다음 목록에는 가장 유용한 로그 레코드가 들어 있습니다.

로그 파일 헤더
각 로그에는 단일 로그 파일 헤더가 있으며 이 헤더는 항상 `dmpmqlog` 명령에 의해 처음으로 형식화됩니다. 이 헤더에는 다음 필드가 포함됩니다.

logactive	기본 로그 익스텐트의 수입니다.
loginactive	보조 로그 익스텐트의 수입니다.
logsize	익스텐트당 4KB 페이지의 수입니다.
baselsn	로그의 헤드가 포함된 로그 익스텐트에서 첫 번째 LSN입니다.
nextlsn	기록될 다음 로그 레코드의 LSN입니다.
headlsn	로그의 헤드에 있는 로그 레코드의 LSN입니다.
tailsn	로그 끝의 위치를 식별하는 LSN입니다.
hflag1	로그가 CIRCULAR 또는 LOG RETAIN(선형)인지 여부입니다.
HeadExtentID	로그의 헤드가 포함된 로그 익스텐트입니다.

로그 레코드 헤더
로그 내 각 로그 레코드에는 다음 정보를 포함하는 고정된 헤더가 있습니다.

LSN	로그 순서 번호입니다.
LogRecdType	로그 레코드의 유형입니다.

XTranid	이 로그 레코드와 연관된 트랜잭션 ID입니다(있는 경우). *TranType*이 MQI이면 IBM MQ 전용 트랜잭션을 나타냅니다. *TranType*이 XA이면 다른 자원 관리자와 관련됩니다. 같은 작업 단위 내에 포함된 업데이트는 같은 *XTranid*를 가집니다.
QueueName	이 로그 레코드와 연관된 큐입니다(있는 경우).
Qid	큐의 고유 내부 ID입니다.
PrevLSN	같은 트랜잭션 내 이전 로그 레코드의 LSN입니다(있는 경우).

큐 관리자 시작

큐 관리자가 시작되었음을 로깅합니다.

StartDate	큐 관리자가 시작된 날짜입니다.
StartTime	큐 관리자가 시작된 시간입니다.

큐 관리자 중지

큐 관리자가 중지되었음을 로깅합니다.

StopDate	큐 관리자가 중지된 날짜입니다.
StopTime	큐 관리자가 중지된 시간입니다. .
ForceFlag	사용된 시스템 종료의 유형입니다.

체크포인트 시작

큐 관리자 체크포인트의 시작을 나타냅니다.

체크포인트 종료

큐 관리자 체크포인트의 종료를 나타냅니다.

ChkPtLSN	이 체크포인트를 시작한 로그 레코드의 LSN입니다.

메시지 넣기

큐에 넣은 지속 메시지를 로깅합니다. 메시지를 동기점 아래 넣은 경우 로그 레코드 헤더는 널이 아닌 *XTranid*를 포함합니다. 레코드의 나머지는 다음을 포함합니다.

MapIndex	큐에 있는 메시지의 ID입니다. 이는 큐에서 이 메시지를 가져오는 데 사용된 해당 MQGET을 일치시키는 데 사용할 수 있습니다. 이 경우 후속 *Get Message* 로그 레코드가 같은 QueueName 및 MapIndex를 포함하는 것으로 확인될 수 있습니다. 이 때 MapIndex ID는 해당 큐에 대한 후속 넣기 메시지에 재사용할 수 있습니다.
Data	이 로그 레코드의 16진 덤프에는 다양한 내부 데이터가 포함되어 있고 그 뒤에 메시지 디스크립터(eyecatcher MD)의 표현 및 메시지 데이터 자체가 순서대로 포함되어 있습니다.

부분 넣기

단일 로그 레코드에 로깅되기에는 너무 큰 지속 메시지는 뒤에 단일 *Put Message* 레코드가 배치되는 다중 *Put Part* 로그 레코드로 로깅됩니다. *Put Part* 레코드가 있을 경우 *PrevLSN* 필드는 *Put Part* 레코드와 최종 *Put Message* 레코드를 결합합니다.

Data	이전 로그 레코드가 중단된 메시지 데이터가 계속됩니다.

메시지 가져오기

지속 메시지의 가져오기만 로깅됩니다. 동기점 아래로 메시지를 가져온 경우 로그 레코드 헤더는 널이 아닌 *XTranid*를 포함합니다. 레코드의 나머지는 다음을 포함합니다.

MapIndex	큐에서 검색된 메시지를 식별합니다. 같은 `QueueName` 및 `MapIndex`를 포함하는 최근 `Put Message` 로그 레코드는 검색된 메시지를 식별합니다.
QPriority	큐에서 검색되는 메시지의 우선순위입니다.

트랜잭션 시작
새 트랜잭션의 시작을 표시합니다. TranType이 MQI이면 IBM MQ 전용 트랜잭션을 나타냅니다. TranType이 XA이면 다른 자원 관리자가 관련된 트랜잭션을 나타냅니다. 이 트랜잭션이 작성한 모든 업데이트는 같은 `XTranid`를 가집니다.

트랜잭션 준비
큐 관리자가 지정된 `XTranid`와 연관된 업데이트를 커미트할 준비가 되었음을 표시합니다. 이 로그 레코드는 다른 자원 관리자도 관련되는 2단계 커미트의 일부로 기록됩니다.

트랜잭션 커미트
큐 관리자가 트랜잭션에서 작성된 모든 업데이트를 커미트했음을 표시합니다.

트랜잭션 롤백
트랜잭션을 롤백하려는 큐 관리자의 의도를 나타냅니다.

트랜잭션 종료
롤백된 트랜잭션의 종료를 나타냅니다.

트랜잭션 테이블
이 레코드는 동기점 중에 기록됩니다. 이 레코드는 지속적 업데이트를 해온 각 트랜잭션의 상태를 기록합니다. 각 트랜잭션에 대해 다음 정보가 기록됩니다.

XTranid	트랜잭션 ID입니다.
FirstLSN	트랜잭션과 연관된 첫 번째 로그 레코드의 LSN입니다.
LastLSN	트랜잭션과 연관된 마지막 로그 레코드의 LSN입니다.

트랜잭션 참가자
이 로그 레코드는 큐 관리자의 XA 트랜잭션 관리자에 의해 기록됩니다. 이 레코드는 트랜잭션에 참가하는 외부 자원 관리자를 기록합니다. 각 참가자에 대해 다음이 기록됩니다.

RMName	자원 관리자의 이름입니다.
RMID	자원 관리자 ID입니다. 이 ID는 또한 자원 관리자가 참가하는 글로벌 트랜잭션을 기록하는 후속 `Transaction Prepared` 로그 레코드에 로그인합니다.
SwitchFile	이 자원 관리자에 대한 스위치 로드 파일입니다.
XAOpenString	이 자원 관리자에 대한 XA 열기 문자열입니다.
XACloseString	이 자원 관리자에 대한 XA 닫기 문자열입니다.

트랜잭션이 준비됨
이 로그 레코드는 큐 관리자의 XA 트랜잭션 관리자에 의해 기록됩니다. 이 레코드는 지정된 글로벌 트랜잭션이 성공적으로 준비되었음을 표시합니다. 참가하는 자원 관리자 각각은 커미트하도록 지시를 받습니다. 준비된 자원 관리자 각각의 `RMID`가 로그 레코드에 기록됩니다. 큐 관리자 자체가 트랜잭션에 참가하는 경우 `RMID`가 0인 `Participant Entry`가 표시됩니다.

트랜잭션 정보 제거(forget)
이 로그 레코드는 큐 관리자의 XA 트랜잭션 관리자에 의해 기록됩니다. 이 로그 레코드는 커미트 결정이 각 참가자에게 전달 완료될 때 `Transaction Prepared` 로그 레코드 뒤에 옵니다.

큐 영구 제거
MQSC 명령 CLEAR QUEUE 등을 사용하여 큐의 모든 메시지가 영구 되었다는 사실을 로깅합니다.

큐 속성
큐의 속성 초기화 또는 변경을 로깅합니다.

오브젝트 작성
IBM MQ 오브젝트의 작성을 로깅합니다.

ObjName 작성된 오브젝트의 이름입니다.

UserId 작성을 수행하는 사용자 ID입니다.

오브젝트 삭제
IBM MQ 오브젝트의 삭제를 로깅합니다.

ObjName 삭제된 오브젝트의 이름입니다.

IBM MQ 큐 관리자 데이터 백업 및 복원

큐 관리자 및 큐 관리자 데이터를 백업하고 큐 관리자 구성만 백업하며 백업 큐 관리자를 사용하는 방식으로 하드웨어 고장으로 인해 발생 가능한 손상으로부터 큐 관리자를 보고할 수 있습니다.

이 태스크 정보

하드웨어 실패로 인해 발생할 수 있는 손상으로부터 큐 관리자를 보호하기 위한 조치를 주기적으로 수행할 수 있습니다. 큐 관리자를 보호하는 세 가지 방법이 있습니다.

큐 관리자 데이터 백업
하드웨어가 실패하는 경우 큐 관리자 중지가 강제 실행될 수 있습니다. 큐 관리자 로그 데이터가 하드웨어 실패로 인해 손실된 경우 큐 관리자를 재시작하지 못할 수 있습니다. 큐 관리자 데이터를 백업하는 경우 손실된 큐 관리자 데이터의 일부 또는 전체를 복구할 수도 있습니다.

일반적으로 큐 관리자 데이터를 백업하는 빈도가 높을수록 복구 로그의 무결성 손실의 원인이 되는 하드웨어 실패 시 손실되는 데이터가 줄어듭니다.

큐 관리자 데이터를 백업하려면 큐 관리자가 실행 중이면 안됩니다.

큐 관리자 구성만 백업
하드웨어가 실패하는 경우 큐 관리자 중지가 강제 실행될 수 있습니다. 하드웨어 고장으로 인해 큐 관리자 구성과 로그 데이터가 모두 손실된 경우에는 큐 관리자를 재시작하거나 로그에서 복구할 수 없습니다. 큐 관리자 구성을 백업한 경우 큐 관리자와 해당 오브젝트 모두를 저장된 정의에서 다시 작성할 수 있습니다.

큐 관리자 구성을 백업하려면 큐 관리자가 실행 중이어야 합니다.

백업 큐 관리자 사용
하드웨어 실패가 심각한 경우 큐 관리자는 복구 불가능할 수 있습니다. 이러한 상황에서 복구 불가능한 큐 관리자에 전용 백업 큐 관리자가 있을 경우 복구 불가능한 큐 관리자를 대신하여 이 백업 큐 관리자가 활성화될 수 있습니다. 백업 큐 관리자를 정기적으로 업데이트할 경우 복구 불가능한 큐 관리자의 마지막 완료 로그가 포함된 로그 데이터가 백업 큐 관리자 로그에 있을 수 있습니다.

기존 큐 관리자가 실행 중인 동안에도 백업 큐 관리자를 업데이트할 수 있습니다.

프로시저

- 큐 관리자 데이터를 백업 및 복원하려면 다음을 참조하십시오.
  - 501 페이지의 『큐 관리자 데이터 백업』.
  - 502 페이지의 『큐 관리자 데이터 복원』.
- 큐 관리자 구성을 백업 및 복원하려면 다음을 참조하십시오.
  - 503 페이지의 『큐 관리자 구성 백업』
  - 504 페이지의 『큐 관리자 구성 복원』
- 백업 큐 관리자를 작성, 업데이트 및 시작하려면 505 페이지의 『백업 큐 관리자 사용』의 내용을 참조하십시오.

큐 관리자 데이터 백업

큐 관리자 데이터를 백업하면 하드웨어 오류로 인한 데이터 손실에 대해 보호하는 데 유용할 수 있습니다.

시작하기 전에

큐 관리자 백업을 시작하기 전에 큐 관리자가 실행 중이지 않은지 확인하십시오. 실행 중인 큐 관리자를 백업하려고 시도하면 파일이 복사될 때 업데이트가 진행 중이어서 백업이 일치하지 않을 수 있습니다. 가능한 경우, **endmqm -w** 명령 (대기 종료) 을 실행하여 큐 관리자를 중지하십시오. 실패하는 경우에만 **endmqm -i** 명령 (즉시 종료) 을 사용하십시오.

이 태스크 정보

큐 관리자의 데이터 백업 사본을 작성하려면 다음 태스크를 완료하십시오.

프로시저

1. 구성 파일의 정보를 사용하여 큐 관리자가 해당 데이터 및 해당 로그 파일을 매치한 디렉토리를 검색하십시오.

   자세한 정보는 78 페이지의 『IBM MQ 및 큐 관리자 구성 정보 변경』의 내용을 참조하십시오.

   참고: 디렉토리에 표시되는 이름이 IBM MQ를 사용하고 있는 플랫폼과 호환 가능하도록 변환됩니다. 이름 변환에 대한 자세한 정보는 IBM MQ 파일 이름에 대한 이해를 참조하십시오.
2. 모든 서브디렉토리를 포함하여 모든 큐 관리자의 데이터 및 로그 파일 디렉토리의 사본을 작성하십시오.

   477 페이지의 『로그의 표시』에 설명된 모든 파일, 특히 로그 제어 파일과 172 페이지의 『초기화 및 구성 파일』에 설명된 구성 파일이 누락되지 않았는지 확인하십시오. 디렉토리 중 일부는 비어 있을 수 있지만 나중에 백업을 복원하려면 해당 디렉토리 모두 필요합니다.

   순환 로깅의 경우 큐 관리자 데이터 및 로그의 일치 세트를 복원할 수 있도록 큐 관리자 데이터 및 로그 파일 디렉토리를 동시에 백업하십시오.

   선형 로깅의 경우 큐 관리자 데이터 및 로그 파일 디렉토리를 동시에 백업하십시오. 로그 파일의 해당 완료 순서가 사용 가능한 경우 큐 관리자 데이터 파일만 복원할 수 있습니다.
3. 파일의 소유권을 보존하십시오.

   Linux UNIX IBM MQ for UNIX 및 Linux 시스템의 경우 **tar** 명령을 사용하여 이를 수행할 수 있습니다. 2GB를 초과하는 크기의 큐가 있는 경우 **tar** 명령을 사용할 수 없습니다. 자세한 정보는 큰 큐를 사용으로 설정을 참조하십시오.

   참고: IBM WebSphere MQ 7.5 이상으로 업그레이드할 경우 `qm.ini` 파일과 레지스트리 항목을 백업해야 합니다. 큐 관리자 정보는 `qm.ini` 파일에 저장되며 IBM MQ의 이전 버전으로 되돌리는 데 사용할 수 있습니다.

관련 태스크

11 페이지의 『큐 관리자 중지』
endmqm 명령을 사용하여 큐 관리자를 중지할 수 있습니다. 이 명령은 세 가지 방법 즉, 제어된 종료 또는 정상 종료(Quiesced shutdown), 즉시 종료, 강제 종료(Preemptive shutdown)를 사용하여 큐 관리자를 중지합니다. 또는 Windows 및 Linux에서 IBM MQ Explorer를 사용하여 큐 관리자를 중지할 수 있습니다.

10 페이지의 『큐 관리자 작성 후 구성 파일 백업』
IBM MQ 구성 정보는 UNIX, Linux, and Windows의 구성 파일에 저장됩니다. 큐 관리자를 작성한 후 구성 파일을 백업하십시오. 그런 다음 문제점을 초래하는 다른 큐 관리자를 작성하는 경우 문제점의 원인을 제거할 때 백업을 복원할 수 있습니다.

큐 관리자 데이터 복원

다음 단계에 따라 큐 관리자 데이터의 백업을 복원합니다.

시작하기 전에

백업을 시작하기 전에 큐 관리자가 실행 중이지 않은지 확인하십시오.

클러스터에서 큐 관리자의 백업을 복원할 때 자세한 정보는 289 페이지의 『클러스터 큐 관리자 복구』 및 클러스터링: 가용성, 다중 인스턴스 및 재해 복구 를 참조하십시오.

참고: IBM WebSphere MQ 7.5 이상으로 업그레이드할 경우 **.ini** 파일과 레지스트리 항목을 백업해야 합니다. 큐 관리자 정보는 **.ini** 파일에 저장되고 IBM MQ의 이전 버전으로 돌아가는 데 사용될 수 있습니다.

프로시저

1. 구성 파일의 정보를 사용하여 큐 관리자가 해당 데이터 및 해당 로그 파일을 매치한 디렉토리를 찾으십시오.
2. 백업된 데이터를 배치할 디렉토리를 비우십시오.
3. 백업된 큐 관리자 데이터 및 로그 파일을 올바른 위치에 복사하십시오.

   로그 파일과 함께 로그 제어 파일이 있는지 확인하십시오.

   순환 로깅의 경우 큐 관리자 데이터 및 로그의 일치 세트를 복원할 수 있도록 큐 관리자 데이터 및 로그 파일 디렉토리를 동시에 백업하십시오.

   선형 로깅의 경우 큐 관리자 데이터 및 로그 파일 디렉토리를 동시에 백업하십시오. 로그 파일의 해당 완료 순서가 사용 가능한 경우 큐 관리자 데이터 파일만 복원할 수 있습니다.
4. 구성 정보 파일을 업데이트하십시오.

   IBM MQ가 복원된 데이터를 올바른 위치에서 찾을 수 있도록 IBM MQ와 큐 관리자 구성 파일이 일치하는지 확인하십시오.
5. 결과로 생성된 디렉토리 구조를 확인하여 필요한 모든 디렉토리가 있는지 확인하십시오.

   IBM MQ 디렉토리 및 서브디렉토리에 대한 자세한 정보는 Windows 시스템의 디렉토리 구조 및 UNIX and Linux 시스템의 디렉토리 컨텐츠를 참조하십시오.

결과

데이터가 올바르게 백업 및 복원된 경우 큐 관리자가 시작됩니다.

Multi 큐 관리자 구성 백업

큐 관리자 구성을 백업하면 하드웨어 고장으로 인해 큐 관리자 구성과 로그 데이터가 모두 손실되고 큐 관리자를 재시작하거나 로그에서 복구할 수 없는 경우에 해당 정의에서 큐 관리자를 다시 빌드할 수 있습니다.

이 태스크 정보

ULW UNIX, Linux, and Windows에서, **dmpmqcfg** 명령을 사용하여 IBM MQ 큐 관리자의 구성을 덤프할 수 있습니다.

IBM i IBM i에서, MQ 구성 덤프(**DMPMQMCFG**) 명령을 사용하여 큐 관리자에 구성 오브젝트 및 권한을 덤프할 수 있습니다.

프로시저

1. 큐 관리자가 실행 중인지 확인하십시오.
2. 플랫폼에 따라 다음 명령 중 하나를 실행하여 큐 관리자 구성을 백업하십시오.

   - ULW UNIX, Linux, and Windows의 경우: MQSC의 기본 형식화 옵션(-f mqsc)과 모든 속성(-a)을 사용하여 MQ 구성 덤프 명령(**dmpmqcfg**)을 실행하고, 표준 출력 방향 재지정을 사용하여 정의를 파일에 저장하십시오. 예를 들면, 다음과 같습니다.

     ```
     dmpmqcfg -m MYQMGR -a > /mq/backups/MYQMGR.mqsc
     ```

- IBM i IBM i의 경우: 기본 형식화 옵션 OUTPUT(*MQSC) 및 EXPATTR(*ALL)을 사용하여 MQ 구성 덤프 명령(**DMPMQMCFG**)을 실행하고, TOFILE 및 TOMBR을 사용하여 정의를 실제 파일 멤버에 저장하십시오. 예를 들면, 다음과 같습니다.

```
DMPMQMCFG MQMNAME(MYQMGR) OUTPUT(*MQSC) EXPATTR(*ALL) TOFILE(QMQMSAMP/QMQSC)
TOMBR(MYQMGRDEF)
```

관련 태스크

504 페이지의 『큐 관리자 구성 복원』
먼저 큐 관리자가 실행 중인지 확인한 후 사용자의 플랫폼에 적합한 명령을 실행하여 백업에서 큐 관리자에 대한 구성을 복원할 수 있습니다.

관련 정보

dmpmqcfg(큐 관리자 구성 덤프)
MQ 구성 덤프(DMPMQMCFG)

Multi 큐 관리자 구성 복원

먼저 큐 관리자가 실행 중인지 확인한 후 사용자의 플랫폼에 적합한 명령을 실행하여 백업에서 큐 관리자에 대한 구성을 복원할 수 있습니다.

이 태스크 정보

ULW UNIX, Linux, and Windows에서, **runmqsc** 명령을 사용하여 IBM MQ 큐 관리자의 구성을 복원할 수 있습니다.

IBM i IBM i의 경우, **STRMQMMQSC** 명령을 사용하여 큐 관리자의 구성 오브젝트와 권한을 복원할 수 있습니다.

프로시저

1. 큐 관리자가 실행 중인지 확인하십시오.
   데이터 및 로그에 대한 손상을 다른 방법으로 복구할 수 없는 경우 큐 관리자가 재작성되었을 수 있습니다.
2. 플랫폼에 따라 다음 명령 중 하나를 실행하여 큐 관리자 구성을 복원하십시오.
   - ULW UNIX, Linux, and Windows에서 큐 관리자에 대해 **runmqsc**를 실행하고, 표준 입력 경로 재지정을 사용하여 MQ 구성 덤프(**dmpmqcfg**) 명령으로 생성되는 스크립트 파일에서 정의를 복원하십시오(503 페이지의 『큐 관리자 구성 백업』 참조). 예를 들면, 다음과 같습니다.

     ```
     runmqsc MYQMGR < /mq/backups/MYQMGR.mqsc
     ```

   - IBM i IBM i의 경우: 큐 관리자에 대해 **STRMQMMQSC** 명령을 실행하고, **SRCMBR** 및 **SRCFILE** 매개변수를 사용하여 MQ 구성 덤프(**DMPMQMCFG**) 명령으로 생성되는 실제 파일 멤버에서 정의를 복원하십시오(503 페이지의 『큐 관리자 구성 백업』). 예를 들면, 다음과 같습니다.

     ```
     STRMQMMQSC MQMNAME(MYQMGR) SRCFILE(QMQMSAMP/QMQSC) SRCMBR(MYQMGR)
     ```

관련 태스크

503 페이지의 『큐 관리자 구성 백업』
큐 관리자 구성을 백업하면 하드웨어 고장으로 인해 큐 관리자 구성과 로그 데이터가 모두 손실되고 큐 관리자를 재시작하거나 로그에서 복구할 수 없는 경우에 해당 정의에서 큐 관리자를 다시 빌드할 수 있습니다.

관련 정보

dmpmqcfg(큐 관리자 구성 덤프)
runmqsc(MQSC 명령 실행)

MQ 구성 덤프(DMPMQMCFG)
IBM MQ 명령 시작(STRMQMMQSC)

백업 큐 관리자 사용

기존 큐 관리자에는 재해 복구 목적으로 전용 백업 큐 관리자가 있을 수 있습니다.

이 태스크 정보

백업 큐 관리자는 기존 큐 관리자의 비활성 사본입니다. 기존 큐 관리자가 심각한 하드웨어 실패로 인해 복구 불가능하게 되면 백업 큐 관리자가 온라인 상태가 되어 복구 불가능한 큐 관리자를 대체할 수 있습니다.

기존 큐 관리자 로그 파일은 정기적으로 백업 큐 관리자에 복사되어 백업 큐 관리자가 재해 복구를 위해 효과적인 메소드로 남을 수 있도록 해야 합니다. 로그 파일을 복사하기 위해 기존 큐 관리자를 반드시 중지해야 하는 것은 아닙니다. 그러나 큐 관리자가 로그 파일에 기록을 완료한 경우에만 로그 파일을 복사해야 합니다. 기존 큐 관리자 로그는 계속해서 업데이트되기 때문에 기존 큐 관리자 로그와 백업 큐 관리자 로그에 복사되는 로그 데이터 사이에는 항상 약간의 불일치가 있습니다. 백업 큐 관리자에 대한 정기적 업데이트는 이 두 로그 간 불일치를 최소화합니다.

백업 큐 관리자를 온라인으로 가져와야 하는 경우 백업 큐 관리자를 활성화한 다음 시작해야 합니다. 백업 큐 관리자가 시작되기 전에 활성화하는 요구사항은 백업 큐 관리자가 갑자기 시작되지 않도록 보호하기 위한 예방 조치입니다. 백업 큐 관리자가 활성화된 후에는 업데이트할 수 없습니다.

중요사항: 이전 백업 큐 관리자가 새로운 활성 큐 관리자가 되면 백업 큐 관리자는 더 이상 없습니다. 이는 사실상 비동기 복제 양식이므로, 새로운 활성 큐 관리자는 논리적으로 이전 활성 큐 관리자보다 뒤에 생성됩니다. 따라서 이전 활성 큐 관리자는 더 이상 새로운 활성 큐 관리자에 대한 백업으로 사용되지 않습니다.

프로시저

- 백업 큐 관리자를 작성하고 업데이트하고 시작하는 방법에 대한 정보는 다음 주제를 참조하십시오.
  - 505 페이지의 『백업 큐 관리자 작성』
  - 506 페이지의 『백업 큐 관리자 업데이트』
  - 506 페이지의 『백업 큐 관리자 시작』

관련 개념

477 페이지의 『로깅: 메시지가 유실되지 않았는지 확인』
IBM MQ는 큐 관리자에 의해 제어되는 데이터에 대한 모든 중요한 변경사항을 복구 로그에 기록합니다.

백업 큐 관리자 작성

백업 큐 관리자는 기존 큐 관리자의 비활성 사본으로 작성할 수 있습니다.

이 태스크 정보

중요사항: 선형 로깅을 사용할 때에만 백업 큐 관리자를 사용할 수 있습니다.

백업 큐 관리자의 요구사항은 다음과 같습니다.

- 기존 큐 관리자와 같은 속성(예: 큐 관리자 이름, 로깅 유형 및 로그 파일 크기)을 가져야 합니다.
- 기존 큐 관리자와 같은 플랫폼에 있어야 합니다.
- 기존 큐 관리자 이상의 코드 레벨에 있어야 합니다.

프로시저

1. 제어 명령 **`crtmqm`**을 사용하여 기존 큐 관리자의 백업 큐 관리자를 작성하십시오.
2. 501 페이지의 『큐 관리자 데이터 백업』에서 설명된 대로 기존 큐 관리자의 모든 데이터 및 모든 서브디렉토리를 포함한 로그 파일 디렉토리 사본을 작성하십시오.
3. 모든 서브디렉토리를 포함하여 백업 큐 관리자의 데이터 및 로그 파일 디렉토리를 기존 큐 관리자에서 작성한 사본으로 덮어쓰십시오.

4. 다음 예에 표시된 것과 같이 백업 큐 관리자에서 **strmqm** 제어 명령을 실행하십시오.

   ```
   strmqm -r BackupQMName
   ```

   이 명령은 IBM MQ 내에서 큐 관리자를 백업 큐 관리자로 플래그 지정하고, 복사된 모든 로그 익스텐트를 재실행하여 백업 큐 관리자를 기존 큐 관리자와 함께 가져오십시오.

관련 정보

crtmqm(큐 관리자 작성)

strmqm(큐 관리자 시작)

백업 큐 관리자 업데이트

백업 큐 관리자가 재해 복구에 효과적인 방법으로 사용되게 하려면 정기적으로 업데이트해야 합니다.

이 태스크 정보

정기적 업데이트는 백업 큐 관리자 로그와 현재 큐 관리자 로그 간 불일치를 줄입니다. 백업하기 전에 큐 관리자를 중지하지 않아도 됩니다.

경고: 비연속적인 로그 세트를 백업 큐 관리자로 복사할 경우 첫 번째 누락된 로그를 찾은 지점까지의 로그만 재실행됩니다.

프로시저

1. 백업할 큐 관리자에서 다음 스크립트(MQSC) 명령을 발행하십시오.

   ```
   RESET QMGR TYPE(ADVANCELOG)
   ```

   이는 현재 로그에 대한 기록을 중지한 다음 큐 관리자 로깅을 다음 로그 익스텐트로 진행시킵니다. 이렇게 하면 현재까지 로깅된 모든 정보를 백업합니다.
2. 백업할 큐 관리자에서 다음 스크립트(MQSC) 명령을 발행하여 (새) 현재 활성 로그 익스텐트를 확보하십시오.

   ```
   DIS QMSTATUS CURRLOG
   ```

3. 업데이트된 로그 익스텐트 파일을 현재 큐 관리자 로그 디렉토리에서 백업 큐 관리자 로그 디렉토리로 복사하십시오.

   마지막 업데이트 이후의 모든 로그 익스텐트 및 506 페이지의 『2』에 표시된 현재 익스텐트까지(현재 익스텐트는 포함되지 않음) 복사하십시오. "S..."로 시작하는 로그 범위 파일만 복사하십시오.
4. 다음 예에 표시된 것과 같이 백업 큐 관리자에서 **strmqm** 제어 명령을 실행하십시오.

   ```
   strmqm -r BackupQMName
   ```

   이 명령은 복사된 모든 로그 익스텐트를 재실행하고 백업 큐 관리자를 큐 관리자가 있는 단계로 가져옵니다. 재실행이 완료되면 재시작 복구에 필요한 모든 로그 익스텐트와 매체 복구에 필요한 모든 로그 익스텐트를 식별하는 메시지를 수신합니다.

관련 정보

RESET QMGR

DISPLAY QMSTATUS

strmqm(큐 관리자 시작)

백업 큐 관리자 시작

복구 불가능한 큐 관리자를 백업 큐 관리자가 대신할 수 있습니다.

이 태스크 정보

복구 불가능한 큐 관리자에 전용 백업 큐 관리자가 있을 경우 복구 불가능한 큐 관리자를 대신하여 백업 큐 관리자를 활성화할 수 있습니다.

복구 불가능한 큐 관리자를 백업 큐 관리자로 대체할 경우 복구 불가능한 큐 관리자의 일부 큐 관리자 데이터가 손실될 수 있습니다. 손실되는 데이터의 양은 백업 큐 관리자가 얼마나 최근에 마지막으로 업데이트되었는지에 따라 다릅니다. 마지막 업데이트가 최근일수록 큐 관리자 데이터 손실이 적어집니다.

참고: 큐 관리자 데이터와 로그 파일이 서로 다른 디렉토리에 있더라도 이들 디렉토리를 동시에 백업하고 복원해야 합니다. 큐 관리자 데이터 및 로그 파일이 서로 연령이 다른 경우 큐 관리자가 유효한 상태가 아니고 시작되지 않을 수 있습니다. 큐 관리자가 시작되더라도 데이터가 손상될 수 있습니다.

프로시저

1. 다음 예에 표시된 것과 같이 **strmqm** 제어 명령을 실행하여 백업 큐 관리자를 활성화하십시오.

   ```
   strmqm -a BackupQMName
   ```

   백업 큐 관리자가 활성화됩니다. 활성화된 백업 큐 관리자는 더 이상 업데이트할 수 없습니다.
2. 다음 예에 표시된 것과 같이 **strmqm** 제어 명령을 실행하여 백업 큐 관리자를 시작하십시오.

   ```
   strmqm BackupQMName
   ```

   IBM MQ에서는 이를 재시작 복구로 간주하고 백업 큐 관리자의 로그를 사용합니다. 백업 큐 관리자에 대한 마지막 업데이트 중에 재실행이 발생하므로, 마지막으로 기록된 체크포인트의 활성 트랜잭션만 롤백됩니다.
3. 모든 채널을 재시작하십시오.
4. 결과로 생성된 디렉토리 구조를 확인하여 필요한 모든 디렉토리가 있는지 확인하십시오.

   IBM MQ 디렉토리 및 서브디렉토리에 대한 자세한 정보는 파일 시스템 지원 계획을 참조하십시오.
5. 로그 파일과 함께 로그 제어 파일이 있는지 확인하십시오. IBM MQ가 복원된 데이터를 올바른 위치에서 찾을 수 있도록 IBM MQ 및 큐 관리자 구성 파일이 일치하는지도 확인하십시오.

결과

데이터가 올바르게 백업되고 복원된 경우 큐 관리자가 시작됩니다.

관련 태스크

165 페이지의 『중지된 채널 재시작』
채널이 STOPPED 상태가 되면 채널을 수동으로 재시작해야 합니다.

관련 정보

strmqm(큐 관리자 시작)

클러스터 오류 복구 변경사항(z/OS 외의 서버에서)

IBM WebSphere MQ 7.1부터, 큐 관리자는 문제점이 해결될 때까지 문제점의 원인이 되는 조작을 재실행합니다. 5일이 지난 후에도 문제점이 해결되지 않으면 캐시가 더 이상 만료되지 않도록 큐 관리자가 종료됩니다.

IBM WebSphere MQ 7.1 전에는 큐 관리자가 클러스터를 관리하는 로컬 저장소 관리자의 문제점을 발견하면 오류 로그를 업데이트했습니다. 그런 다음 클러스터 관리를 중지하는 경우도 있었습니다. 큐 관리자는 계속해서 클러스터와 애플리케이션 메시지를 교환했으며 이에 따라 클러스터 정의의 만료된 캐시에 더욱 의존했습니다.
IBM WebSphere MQ 7.1부터, 큐 관리자는 문제점이 해결될 때까지 문제점의 원인이 되는 조작을 재실행합니다. 5일이 지난 후에도 문제점이 해결되지 않으면 캐시가 더 이상 만료되지 않도록 큐 관리자가 종료됩니다. 캐시가 더 많이 만료될수록 많은 수의 문제점이 발생하게 됩니다. 7.1 이상의 클러스터 오류에 대한 변경된 동작은 z/OS에 적용되지 않습니다.

클러스터 관리의 모든 측면은 로컬 저장소 관리자 프로세스 amqrrmfa에 의해 큐 관리자에 대해 처리됩니다. 이 프로세스는 모든 큐 관리자에 대해 실행되며 클러스터 정의가 없는 경우에도 마찬가지입니다.

IBM WebSphere MQ 7.1 전에는 큐 관리자가 로컬 저장소 관리자의 문제점을 발견하면 잠시 후에 저장소 관리자를 중지했습니다. 이때 큐 관리자는 계속해서 실행되고 애플리케이션 메시지와 요청을 처리하면서 큐를 열고 토픽을 발행하거나 구독했습니다.

저장소 관리자가 중지되면 큐 관리자가 사용할 수 있는 클러스터 정의의 캐시가 만료되었습니다. 시간이 지나면서 메시지는 올바르지 않은 목적지로 라우트되고 애플리케이션은 실패했습니다. 애플리케이션은 로컬 큐 관리자로 전파되지 않은 클러스터 큐나 발행 토픽을 열지 못했습니다.

관리자가 오류 로그에서 저장소 메시지를 확인하지 않는 한 관리자는 클러스터 구성에 문제점이 있다는 것을 인식하지 못합니다. 오랫동안 이러한 오류가 인식되지 않으면 큐 관리자는 해당 클러스터 멤버십을 갱신하지 않으므로 더 많은 문제점이 발생했습니다. 이러한 불안정한 상황이 클러스터의 모든 큐 관리자에 영향을 미치고 클러스터는 불안정한 상태에 있었습니다.

IBM WebSphere MQ 7.1 이후부터 IBM MQ에서는 클러스터 오류 처리에 대한 다른 방식을 사용합니다. 저장소 관리자를 중지하고 저장소 관리자 없이 작업을 진행하는 것이 아니라 저장소 관리자가 실패한 조작을 다시 실행합니다. 큐 관리자가 저장소 관리자 문제점을 감지하면 다음과 같은 두 가지 조치 중 하나를 수행합니다.

1. 오류로 인해 큐 관리자의 조작이 지장을 받지 않으면 큐 관리자는 메시지를 오류 로그에 씁니다. 큐 관리자는 조작이 성공할 때까지 10분마다 실패한 조작을 다시 실행합니다. 기본적으로 5일 이내에 오류를 처리해야 합니다. 이 기간 동안 오류를 처리하지 못하면 큐 관리자가 메시지를 오류 로그에 쓰고 종료됩니다. 5일 후의 종료를 연기할 수도 있습니다.
2. 오류로 인해 큐 관리자의 조작이 지장을 받으면 큐 관리자는 메시지를 오류 로그에 쓰고 즉시 종료됩니다.

큐 관리자의 조작에 지장을 주는 오류는 큐 관리자가 진단할 수 없는 오류 또는 결과를 예측할 수 없는 오류입니다. 이러한 유형의 오류로 인해 종종 큐 관리자가 FFST 파일을 쓰게 됩니다. 큐 관리자의 조작에 방해가 되는 오류는 IBM MQ의 버그, 관리자 또는 프로그램으로 인해 발생할 수 있으며 IBM MQ 프로세스 종료와 같은 예기치 않은 상황이 나타납니다.

오류 복구 동작의 변경 시점은 큐 관리자가 점점 더 증가하는 불일치 클러스터 정의를 계속해서 실행하는 시간을 제한하기 위한 것입니다. 클러스터 정의의 불일치 수가 증가할수록 비정상적인 애플리케이션 동작의 가능성도 함께 증가합니다.

5일 후 큐 관리자를 종료하는 기본 선택은 불일치 수를 제한하는 것과 문제점이 감지되어 해결될 때까지 큐 관리자를 사용 가능하게 유지하는 것 사이의 절충 방법입니다.

문제점을 수정하거나 계획된 큐 관리자 시스템 종료를 기다리면서 큐 관리자가 무기한 종료되기 전에 시간을 연장할 수 있습니다. 5일 동안 그대로 남아 있으면 큐 관리자가 긴 주말 동안 실행되도록 유지함으로써 문제점에 대처하거나 큐 관리자를 다시 시작하기 전에 시간을 연장할 수 있습니다.

정정 조치

클러스터 오류 복구 문제점을 처리하기 위한 조치를 선택할 수 있습니다. 첫 번째 선택은 문제점을 모니터하고 수정하는 조치이고, 두 번째는 문제점을 모니터하고 수정을 연기하는 조치이며, 마지막 선택은 IBM WebSphere MQ 7.1 전 릴리스의 경우와 같이 클러스터 오류 복구를 계속 관리하는 조치입니다.

1. 오류 메시지 AMQ9448 및 AMQ5008의 큐 관리자 오류 로그를 모니터하고 문제점을 수정하십시오.

   AMQ9448은 저장소 관리자가 명령을 실행한 후 오류를 리턴했음을 나타냅니다. 이 오류는 10분마다 명령 재시도를 시작하도록 표시하며 사용자가 종료를 연기하지 않은 경우 실제로 5일 후 큐 관리자를 중지합니다.

   AMQ5008은 IBM MQ 프로세스가 누락되어 큐 관리자가 중지되었음을 나타냅니다. AMQ5008은 5일 후에 저장소 관리자를 중지한 결과로 나타납니다. 저장소 관리자가 중지되면 큐 관리자도 중지됩니다.
2. 오류 메시지 AMQ9448의 큐 관리자 오류 로그를 모니터하고 문제점 수정을 연기하십시오.

   `SYSTEM.CLUSTER.COMMAND.QUEUE`에서 메시지 가져오기를 사용하지 않을 경우 저장소 관리자가 명령 실행을 중지하고 작업을 처리하지 않은 채로 무기한 계속됩니다. 그러나 저장소 관리자가 큐에 대해 보유하는 핸들이 해제됩니다. 저장소 관리자가 중지되지 않으므로 5일 후 큐 관리자도 중지되지 않습니다.

   다음과 같이 MQSC 명령을 실행하여 `SYSTEM.CLUSTER.COMMAND.QUEUE`에서 메시지 가져오기를 사용하지 않도록 설정합니다.

   ```
   ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(DISABLED)
   ```

   `SYSTEM.CLUSTER.COMMAND.QUEUE`에서 메시지를 다시 수신하려면 MQSC 명령을 실행하십시오.

   ```
   ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(ENABLED)
   ```
3. IBM WebSphere MQ 7.1 전과 동일한 클러스터 오류 복구 동작으로 큐 관리자를 되돌리십시오.

   저장소 관리자가 중지할 경우 큐 관리자가 계속 실행되도록 큐 관리자 성능 조정 매개변수를 설정할 수 있습니다.

   튜닝 매개변수는 `qm.ini` 파일의 `TuningParameters` 스탠자에 있는 `TolerateRepositoryFailure`입니다. 저장소 관리자가 중지될 경우 큐 관리자가 중지되지 않게 하려

면 `TolerateRepositoryFailure`를 TRUE로 설정하십시오. 509 페이지의 그림 88의 내용을 참조하십시오.
큐 관리자를 다시 시작하여 `TolerateRepositoryFailure` 옵션을 사용하도록 설정합니다.
저장소 관리자의 시작을 막는 클러스터 오류가 발생하여 큐 관리자가 시작되지 않으면 `TolerateRepositoryFailure`를 TRUE로 설정하여 저장소 관리자 없이 큐 관리자를 시작합니다.

특수 고려사항

IBM WebSphere MQ 7.1 이전에는 클러스터의 일부가 아닌 큐 관리자를 관리하는 일부 관리자가 `amqrrmfa` 프로세스를 중지했습니다. `amqrrmfa` 중지는 큐 관리자에 영향을 주지 않습니다.

IBM WebSphere MQ 7.1 또는 나중에 `amqrrmfa`을(를) 중지하면 큐 관리자가 큐 관리자 실패로 간주되므로 큐 관리자가 중지됩니다. 큐 관리자 튜닝 매개변수 `TolerateRepositoryFailure`를 설정하지 않는 한, 7.1 이상에서는 `amqrrmfa` 프로세스를 중지하지 않아야 합니다.

예

```
TuningParameters:
        TolerateRepositoryFailure=TRUE
```

그림 *88. qm.ini*에서 *`TolerateRepositoryFailure`*를 *TRUE*로 설정

관련 정보
큐 관리자 구성 파일, qm.ini

JMS 자원 구성

JMS 애플리케이션이 IBM MQ에 연결에 연결하고 메시지 송신 및 수신을 위해 목적지에 액세스하기 위해 필요한 자원을 작성 및 구성할 수 있는 방법 중 하나는 JNDI(Java Naming and Directory Interface)를 사용하여 JNDI 네임스페이스라고 부르는 네이밍 및 디렉토리 서비스 내의 위치에서 관리 대상 오브젝트를 검색하는 것입니다. JMS 애플리케이션이 JNDI 네임스페이스에서 관리 대상 오브젝트를 검색할 수 있기 전에 먼저 관리 대상 오브젝트를 작성 및 구성해야 합니다.

이 태스크 정보

다음 도구 중 하나를 사용하여 IBM MQ에서 관리 대상 오브젝트를 작성 및 구성할 수 있습니다.

IBM MQ Explorer
IBM MQ Explorer를 사용하여 LDAP, 로컬 파일 시스템 또는 기타 위치에 저장되는 JMS 오브젝트 정의를 작성 및 관리할 수 있습니다.

IBM MQ JMS 관리 도구
IBM MQ JMS 관리 도구는 LDAP, 로컬 파일 시스템 또는 다른 위치에 저장되는 IBM MQ JMS 오브젝트를 작성 및 구성하기 위해 사용할 수 있는 명령행 도구입니다. JMS 관리 도구는 **`runmqsc`**와 비슷한 구문을 사용하며 스크립트를 지원합니다.

관리 도구는 구성 파일을 사용하여 특정 특성의 값을 설정합니다. 샘플 구성 파일이 제공되는데, 도구를 사용하여 JMS 자원을 구성하기 시작하기 전에 이를 편집하여 사용자 시스템에 맞게 편집할 수 있습니다. 구성 파일에 대한 자세한 정보는 515 페이지의 『JMS 관리 도구 구성』의 내용을 참조하십시오.

WebSphere Application Server에 배치되는 IBM MQ JMS 애플리케이션은 애플리케이션 서버 JNDI 저장소의 JMS 오브젝트에 액세스해야 합니다. 그러므로 WebSphere Application Server와 IBM MQ 사이에서 JMS 메시징을 사용하는 경우, IBM MQ에서 작성하는 오브젝트에 대응하는 오브젝트를 WebSphere Application Server에서 작성해야 합니다.

IBM MQ Explorer 및 IBM MQ JMS 관리 도구는 WebSphere Application Server에 저장되는 IBM MQ JMS오브젝트를 관리하는 데 사용될 수 없습니다. 대신, 다음 도구 중 하나를 사용하여 WebSphere Application Server에서 관리 대상 오브젝트를 작성 및 구성할 수 있습니다.

WebSphere Application Server 관리 콘솔
WebSphere Application Server 관리 콘솔은 WebSphere Application Server의 IBM MQ JMS 오브젝트를 관리하는 데 사용할 수 있는 웹 기반 도구입니다.

WebSphere Application Server wsadmin 스크립트 클라이언트
WebSphere Application Server wsadmin 스크립트 클라이언트는 WebSphere Application Server의 IBM MQ JMS 오브젝트를 관리하기 위한 특수한 명령을 제공합니다.

JMS 애플리케이션을 사용하여 WebSphere Application Server 내에서 IBM MQ 큐 관리자의 자원에 액세스하기 원하는 경우, WebSphere Application Server에서 IBM MQ 메시징 제공자를 사용해야 하는데, 여기에 IBM MQ classes for JMS의 버전이 포함됩니다. WebSphere Application Server와 함께 제공되는 IBM MQ 자원 어댑터는 IBM MQ 메시징 어댑터와 함께 JMS 메시징을 수행하는 모든 애플리케이션에 의해 사용됩니다. IBM MQ 자원 어댑터는 대개 WebSphere Application Server 수정팩을 적용할 때 자동으로 업데이트되지만, 이전에 수동으로 자원 어댑터를 업데이트한 경우 유지보수가 올바르게 적용되도록 보장하기 위해 구성을 수동으로 업데이트해야 합니다.

관련 정보

IBM MQ classes for JMS 애플리케이션 작성

runmqsc

JNDI 네임스페이스에서 연결 팩토리 및 목적지 구성

JMS 애플리케이션은 Java JNDI(Naming and Directory Interface)를 통해 네이밍 및 디렉토리 서비스의 관리 대상 오브젝트에 액세스합니다. JMS 관리 대상 오브젝트는 JNDI 네임스페이스라고 부르는 네이밍 및 디렉토리 서비스 내의 위치에 저장됩니다. JMS 애플리케이션은 IBM MQ에 연결할 관리 대상 오브젝트를 찾고 메시지 송신 및 수신을 위한 목적지에 액세스할 수 있습니다.

이 태스크 정보

JMS 애플리케이션은 컨텍스트를 사용하여 네이밍 및 디렉토리 서비스에서 JMS 오브젝트의 이름을 찾습니다.

초기 컨텍스트
초기 컨텍스트는 JNDI 네임스페이스의 루트를 정의합니다. 네이밍 및 디렉토리 서비스의 각 위치에 대해 JMS 애플리케이션이 네이밍 및 디렉토리 서비스의 해당 위치에서 관리 대상 오브젝트의 이름을 해석할 수 있는 시작점을 제공할 초기 컨텍스트를 지정해야 합니다.

하위 컨텍스트
컨텍스트가 하나 이상의 하위 컨텍스트를 가질 수 있습니다. 하위 컨텍스트는 JNDI 네임스페이스의 하위 영역이며 연결 팩토리와 목적지 및 기타 하위 컨텍스트 같은 관리 대상 오브젝트를 포함할 수 있습니다. 하위 컨텍스트는 자체로는 오브젝트가 아니며 단순히 하위 컨텍스트의 오브젝트에 대한 이름 지정 규칙의 확장입니다.

IBM MQ Explorer 또는 IBM MQ JMS 관리 도구를 사용하여 컨텍스트를 작성할 수 있습니다.

IBM MQ classes for JMS 애플리케이션이 JNDI 네임스페이스에서 관리 대상 오브젝트를 검색할 수 있기 전에, 먼저 IBM MQ Explorer 또는 IBM MQ JMS 관리 도구를 사용하여 관리 대상 오브젝트를 작성해야 합니다. JMS 오브젝트의 다음 유형을 작성 및 구성할 수 있습니다.

연결 팩토리
JMS 연결 팩토리 오브젝트는 연결을 위한 표준 구성 특성의 세트를 정의합니다. JMS 애플리케이션은 연결 팩토리를 사용하여 IBM MQ에 대한 연결을 작성합니다. 두 메시지 도메인인 포인트-투-포인트 메시지 도메인과 발행/구독 메시지 도메인 중 하나에 특정한 연결 팩토리를 작성할 수 있습니다. 그렇지 않으면 JMS 1.1에서 포인트-투-포인트 및 발행/구독 메시징 모두에 사용될 수 있는 도메인 독립적 연결 팩토리를 작성할 수 있습니다.

목적지
JMS 목적지는 클라이언트가 생성하는 메시지의 대상과 JMS 애플리케이션이 이용하는 메시지의 소스를 나타내는 오브젝트입니다. JMS 애플리케이션은 단일 목적지 오브젝트를 사용하여 메시지를 넣거나(Put) 메시지를 가져오거나(Get), 애플리케이션이 별도의 목적지 오브젝트를 사용할 수 있습니다. 두 가지 유형의 목적지 오브젝트가 있습니다.

- 포인트-투-포인트 메시징에서 사용되는 JMS 큐 목적지
- 발행/구독 메시징에서 사용되는 JMS 토픽 목적지

다음 다이어그램은 IBM MQ JNDI 네임스페이스에서 작성되는 JMS 오브젝트의 예를 보여줍니다.

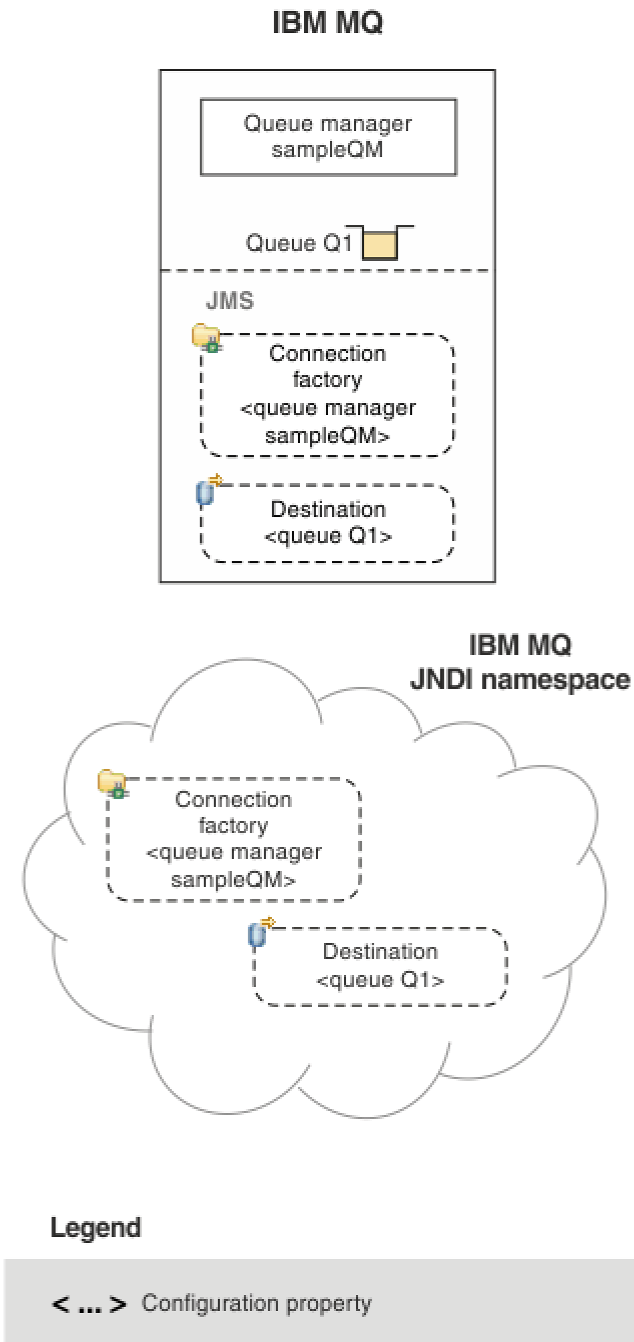


그림 *89. IBM MQ*에서 작성되는 *JMS* 오브젝트

WebSphere Application Server와 IBM MQ 사이에서 JMS 메시징을 사용하는 경우, IBM MQ와 통신하기 위해 사용할 대응하는 오브젝트를 WebSphere Application Server에서 작성해야 합니다. WebSphere Application Server에서 이들 오브젝트 중 하나를 작성할 때, 다음 다이어그램에 표시된 것처럼 WebSphere Application Server JNDI 네임스페이스에 저장됩니다.

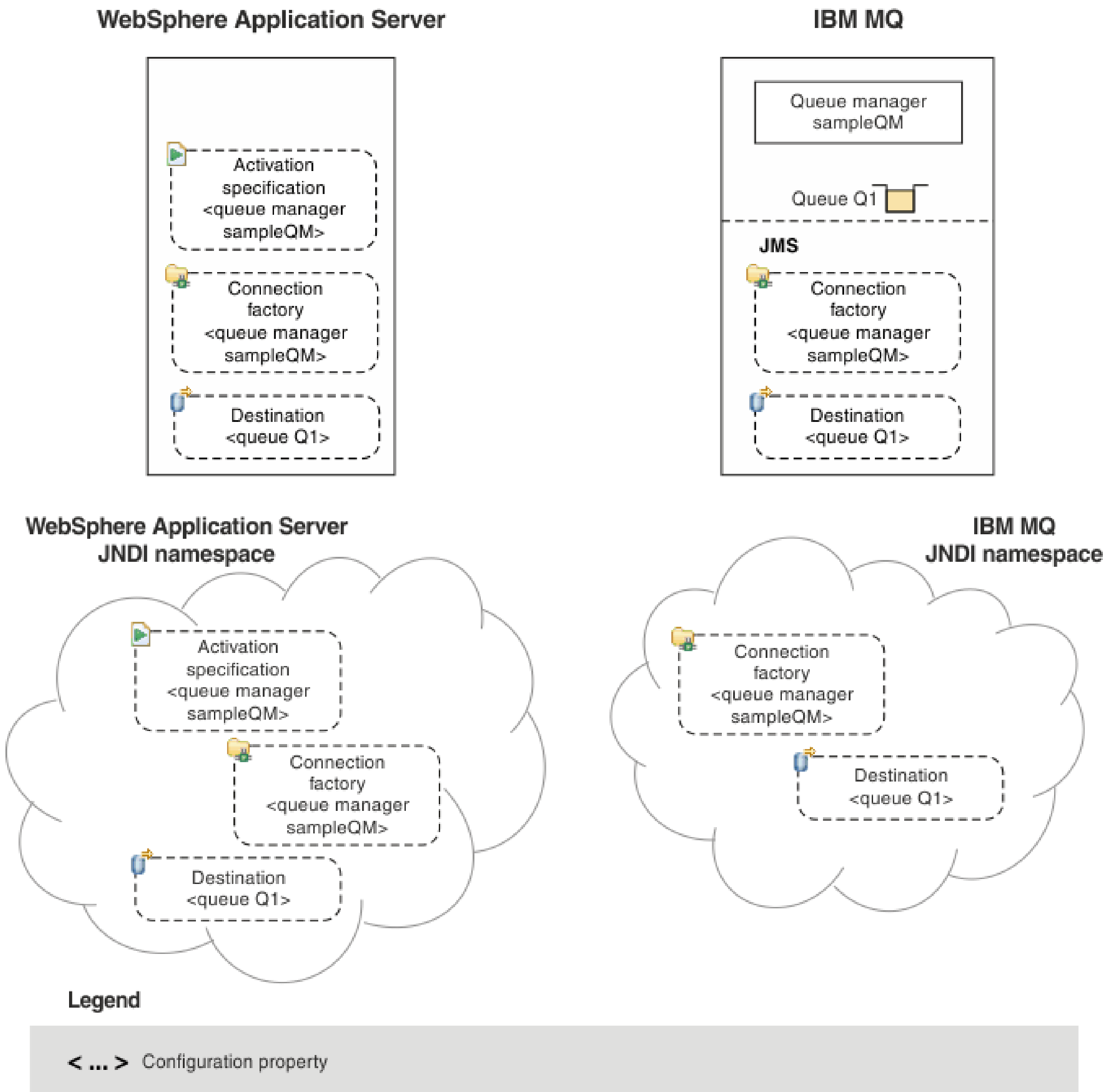


그림 *90. WebSphere Application Server*에서 작성된 오브젝트 및 *IBM MQ*의 해당 오브젝트

애플리케이션이 메시지 구동 Bean(MDB)을 사용하는 경우, 연결 팩토리는 아웃바운드 메시지에만 사용되고 인바운드 메시지는 활성화 스펙에 의해 수신됩니다. 활성화 스펙은 Java EE Connector Architecture 1.5(JCA 1.5) 표준의 일부입니다. JCA 1.5는 IBM MQ 같은 JMS 제공자를 WebSphere Application Server 같은 Java EE 애플리케이션 서버와 통합하는 표준 방법을 제공합니다. JMS 활성화 스펙은 하나 이상의 메시지 구동 Bean(MDB)과 연관될 수 있으며 이들 MDB가 목적지에 도착하는 메시지를 대기하기 위해 필요한 구성을 제공합니다.

WebSphere Application Server 관리 콘솔 또는 wsadmin 스크립트 명령을 사용하여 필요한 JMS 자원을 작성 및 구성할 수 있습니다.

프로시저

- IBM MQ Explorer를 사용하여 IBM MQ에 대한 JMS 오브젝트를 구성하려면 513 페이지의 『IBM MQ Explorer를 사용하여 JMS 오브젝트 구성』의 내용을 참조하십시오.
- IBM MQ JMS 관리 도구를 사용하여 IBM MQ에 대한 JMS 오브젝트를 구성하려면 514 페이지의 『관리 도구를 사용하여 JMS 오브젝트 구성』의 내용을 참조하십시오.

- WebSphere Application Server에 대한 JMS 오브젝트를 구성하려면 522 페이지의 『WebSphere Application Server에서 JMS 자원 구성』의 내용을 참조하십시오.

결과

IBM MQ classes for JMS 애플리케이션은 JNDI 네임스페이스에서 관리 대상 오브젝트를 검색하고, 필요한 경우 IBM JMS 확장이나 IBM MQ JMS 확장을 사용하여 그의 특성 중 하나 이상을 설정 또는 변경할 수 있습니다.

관련 정보

JNDI를 사용하여 JMS 애플리케이션에서 관리 대상 오브젝트 검색

IBM MQ classes for JMS 애플리케이션에서 연결 팩토리 및 목적지 작성 및 구성

IBM MQ Explorer를 사용하여 JMS 오브젝트 구성

다른 IBM MQ 오브젝트 관리 및 모니터링을 위해서 뿐 아니라 IBM MQ 오브젝트에서 JMS 오브젝트를 작성하고 JMS 오브젝트에서 IBM MQ 오브젝트를 작성하려면 IBM MQ Explorer 그래픽 사용자 인터페이스를 사용하십시오.

이 태스크 정보

IBM MQ Explorer는 로컬 컴퓨터에서 호스팅하는지 또는 원격 시스템에서 호스팅하는지에 구애받지 않고 IBM MQ 오브젝트를 관리 및 모니터할 수 있는 그래픽 사용자 인터페이스(GUI)입니다. IBM MQ Explorer는 Windows 및 Linux x86-64에서 실행됩니다. 이는 z/OS를 포함하여 지원되는 플랫폼에서 실행되고 콘솔에서 전체 메시징 백본을 보고, 탐색하고 변경할 수 있는 큐 관리자에게 원격으로 연결할 수 있습니다.

IBM MQ Explorer에서, 모든 연결 팩토리는 적당한 컨텍스트 및 하위 컨텍스트에서 연결 팩토리 폴더에 저장됩니다.

IBM MQ Explorer의 기존 오브젝트로부터 컨텍스트적으로 또는 새 오브젝트 작성 마법사 내에서 IBM MQ Explorer로 다음 유형의 태스크를 수행할 수 있습니다.

- 다음 IBM MQ 오브젝트 중 하나로부터 JMS 연결 팩토리를 작성하십시오.
  - 로컬 컴퓨터 또는 원격 시스템에 있든지 간에, IBM MQ 큐 관리자.
  - IBM MQ 채널.
  - IBM MQ 리스너.
- JMS 연결 팩토리를 사용하여 IBM MQ 큐 관리자를 IBM MQ Explorer에 추가하십시오.
- IBM MQ 큐로부터 JMS 큐를 작성하십시오.
- JMS 큐로부터 IBM MQ 큐를 작성하십시오.
- IBM MQ 오브젝트 또는 동적 토픽일 수 있는 IBM MQ 토픽으로부터 JMS 토픽을 작성하십시오.
- JMS 토픽으로부터 IBM MQ 토픽을 작성하십시오.

프로시저

- 아직 실행 중이 아닌 경우 IBM MQ Explorer를 시작하십시오.

  IBM MQ Explorer가 시작 페이지를 실행하고 표시하는 경우,IBM MQ 오브젝트 관리를 시작하기 위해 시작 페이지를 닫으십시오.
- 아직 수행하지 않은 경우, JMS 오브젝트가 네이밍 및 디렉토리 서비스에서 저장되는 JNDI 네임스페이스의 루트를 정의하는 초기 컨텍스트를 작성하십시오.

  초기 컨텍스트를 IBM MQ Explorer에 추가했을 때, JNDI 네임스페이스에서 연결 팩토리 오브젝트, 목적지 오브젝트 및 하위 컨텍스트를 작성할 수 있습니다.

  초기 컨텍스트는 JMS 관리 대상 오브젝트 폴더의 네비게이터 보기에 표시됩니다. JNDI 네임스페이스의 전체 컨텐츠가 표시되지만 IBM MQ Explorer에서 그곳에 저장되는 IBM MQ classes for JMS 오브젝트만 편집할 수 있음을 참고하십시오. 자세한 정보는 초기 컨텍스트 추가를 참조하십시오.
- 필요한 하위 컨텍스트 및 JMS 관리 대상 오브젝트를 작성하고 구성하십시오.

  자세한 정보는 JMS 관리 대상 오브젝트 작성 및 구성을 참조하십시오.

- IBM MQ를 구성하십시오.
  자세한 정보는 IBM MQ Explorer를 사용한 IBM MQ 구성을 참조하십시오.

관련 정보
소개IBM MQ Explorer
IBM MQ classes for JMS 애플리케이션에서 연결 팩토리 및 목적지 작성 및 구성

관리 도구를 사용하여 JMS 오브젝트 구성

IBM MQ JMS 관리 도구를 사용하여 8가지 유형의 IBM MQ classes for JMS 오브젝트의 특성을 정의하고 JNDI 네임스페이스 안에 저장할 수 있습니다. 그러면 애플리케이션이 JNDI를 사용하여 네임스페이스에서 이들 관리 대상 오브젝트를 검색할 수 있습니다.

이 태스크 정보

다음 표는 verb를 사용하여 작성, 구성 및 조작할 수 있는 관리 대상 오브젝트의 8가지 유형을 보여줍니다. 키워드 열은 514 페이지의 표 33에 표시된 명령에서 *TYPE* 에 대해 대체할 수 있는 문자열을 표시합니다.

표 *33.* 관리 도구가 처리하는 *JMS* 오브젝트 유형

오브젝트 유형	키워드	설명
MQConnectionFactory	CF	JMS ConnectionFactory 인터페이스의 IBM MQ 구현입니다. 이것은 포인트-투-포인트 및 발행/구독 도메인 둘 다에서 연결 작성을 위한 팩토리 오브젝트를 나타냅니다.
MQQueueConnectionFactory	QCF	JMS QueueConnectionFactory 인터페이스의 IBM MQ 구현입니다. 이것은 포인트-투-포인트 도메인에서 연결 작성을 위한 팩토리 오브젝트를 나타냅니다.
MQTopicConnectionFactory	TCF	JMS TopicConnectionFactory 인터페이스의 IBM MQ 구현입니다. 이것은 발행/구독 도메인에서 연결 작성을 위한 팩토리 오브젝트를 나타냅니다.
MQQueue	Q	JMS Queue 인터페이스의 IBM MQ 구현입니다. 이것은 포인트-투-포인트 도메인에서 메시지의 목적지를 나타냅니다.
MQTopic	T	JMS Topic 인터페이스의 IBM MQ 구현입니다. 이것은 발행/구독 도메인에서 메시지의 목적지를 나타냅니다.
MQXAConnectionFactory 515 페이지의 『1』	XACF	JMS XAConnectionFactory 인터페이스의 IBM MQ 구현입니다. 이것은 포인트-투-포인트 및 발행/구독 도메인 둘 다에서 연결 작성을 위한 팩토리 오브젝트 및 연결이 JMS 클래스의 XA 버전을 사용하는 위치를 나타냅니다.
MQXAQueueConnectionFactory 515 페이지의 『1』	XAQCF	JMS XAQueueConnectionFactory 인터페이스의 IBM MQ 구현입니다. 이것은 JMS 클래스의 XA 버전을 사용하는 포인트-투-포인트 도메인에서 연결 작성을 위한 팩토리 오브젝트를 나타냅니다.
MQXATopicConnectionFactory 515 페이지의 『1』	XATCF	JMS XATopicConnectionFactory 인터페이스의 IBM MQ 구현입니다. 이것은 JMS 클래스의 XA 버전을 사용하는 발행/구독 도메인에서 연결 작성을 위한 팩토리 오브젝트를 나타냅니다.

참고:

1. 이들 클래스는 애플리케이션 서버의 벤더가 사용하기 위해 제공됩니다. 애플리케이션 프로그래머에게 직접 유용할 가능성은 없습니다.

이들 오브젝트를 구성하는 방법에 대한 자세한 정보는 521 페이지의 『JMS 오브젝트 구성』의 내용을 참조하십시오.

이 도구를 사용하기 위해 필요한 특성 유형 및 값은 IBM MQ classes for JMS 오브젝트의 특성에 나열되어 있습니다.

또한 519 페이지의 『하위 컨텍스트 구성』에서 설명한대로 이 도구를 사용하여 JNDI 내에서 디렉토리 네임스페이스 하위 컨텍스트를 조작할 수도 있습니다.

또한 IBM MQ Explorer로 JMS 관리 대상 오브젝트를 작성 및 구성할 수도 있습니다.

관련 정보

IBM MQ classes for JMS 애플리케이션에서 연결 팩토리 및 목적지 작성 및 구성
JNDI를 사용하여 JMS 애플리케이션에서 관리 대상 오브젝트 검색

JMS 관리 도구 구성

IBM MQ JMS 관리 도구는 구성 파일을 사용하여 특정 구성의 값을 설정합니다. 샘플 구성 파일이 제공되는데, 사용자 시스템에 맞게 편집할 수 있습니다.

이 태스크 정보

구성 파일은 등호(=)로 구분된 키-값 쌍 세트로 구성되는 일반 텍스트 파일입니다. 구성 파일에 정의된 세 개의 특성에 대한 값을 설정하여 관리 도구를 구성합니다. 다음 예는 이런 세 특성을 보여줍니다.

```
#Set the service provider
INITIAL_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory
#Set the initial context
PROVIDER_URL=ldap://polaris/o=ibm_us,c=us
#Set the authentication type
SECURITY_AUTHENTICATION=none
```

(이 예에서 행의 첫 번째 열에 있는 해시 부호(#)는 주석 또는 사용되지 않는 행을 표시합니다.)

기본 구성 파일로 사용되는 샘플 구성 파일이 IBM MQ와 함께 제공됩니다. 샘플 파일은 `JMSAdmin.config`라고 하며, *MQ_JAVA_INSTALL_PATH*/bin 디렉토리에 있습니다. 이 샘플 파일을 편집하여 시스템에 필요한 설정을 정의하거나, 사용자 자신의 구성 파일을 작성할 수 있습니다.

관리 도구를 시작할 때 517 페이지의 『관리 도구 시작』에서 설명하는 대로 `-cfg` 명령행 매개변수를 사용하여 사용하려는 구성 파일을 지정할 수 있습니다. 도구를 호출할 때 구성 파일을 지정하지 않는 경우 도구는 기본 구성 파일(`JMSAdmin.config`)을 로드하려고 시도합니다. 먼저 현재 디렉토리에서 이 파일을 검색한 후 *MQ_JAVA_INSTALL_PATH*/bin 디렉토리에서 검색하는데, *MQ_JAVA_INSTALL_PATH*는 IBM MQ classes for JMS 설치에 대한 경로입니다.

LDAP 환경에 저장되는 JMS 오브젝트의 이름은 LDAP 이름 지정 규칙을 준수해야 합니다. 이들 규칙 중 하나는 오브젝트 및 컨텍스트 이름이 cn=(공통 이름) 또는 ou=(조직 단위) 같은 접두부를 포함해야 한다는 것입니다. 관리 도구를 사용하면 접두부 없이 오브젝트 및 컨텍스트 이름을 나타낼 수 있어 LDAP 서비스 제공자를 쉽게 사용할 수 있습니다. 접두부를 제공하지 않으면 도구가 자동으로 사용자가 제공하는 이름에 기본 접두부를 추가합니다. LDAP의 경우, `cn=`입니다. 필요한 경우 구성 파일에서 **`NAME_PREFIX`** 특성을 설정하여 기본 접두부를 변경할 수 있습니다.

참고: Java 오브젝트를 저장하려면 LDAP 서버를 구성해야 할 수 있습니다. 자세한 정보는 LDAP 서버 문서를 참조하십시오.

프로시저

1. **`INITIAL_CONTEXT_FACTORY`** 특성을 구성하여 도구가 사용하는 서비스 제공자를 정의하십시오.

   이 특성에 대해 지원되는 값은 다음과 같습니다.

- com.sun.jndi.ldap.LdapCtxFactory(LDAP의 경우)
- com.sun.jndi.fscontext.RefFSContextFactory(파일 시스템 컨텍스트의 경우)
- z/OS com.ibm.jndi.LDAPCtxFactory는 z/OS에서만 지원되며, LDAP 서버에 대한 액세스를 제공합니다. 그러나 이 클래스는 하나의 InitialContextFactory를 사용하여 작성되는 오브젝트가 다른 것을 사용하여 읽거나 수정될 수 없다는 점에서 com.sun.jndi.ldap.LdapCtxFactory와 호환되지 않습니다.

또한 관리 도구를 사용하여 JMSAdmin 구성 파일에서 정의되는 세 개의 매개변수를 사용하여 다른 JNDI 컨텍스트에 연결할 수 있습니다. 다른 InitialContextFactory를 사용하려면 다음을 수행하십시오.

a) **INITIAL_CONTEXT_FACTORY** 특성을 필수 클래스 이름으로 설정하십시오.

b) **USE_INITIAL_DIR_CONTEXT**, **NAME_PREFIX** 및 **NAME_READABILITY_MARKER** 특성을 사용하여 InitialContextFactory의 작동을 정의하십시오.

이들 특성의 설정은 샘플 구성 파일 주석에서 설명됩니다.

지원되는 **INITIAL_CONTEXT_FACTORY** 값 중 하나를 사용하는 경우 **USE_INITIAL_DIR_CONTEXT**, **NAME_PREFIX** 및 **NAME_READABILITY_MARKER** 특성을 정의할 필요가 없습니다. 그러나 시스템 기본값을 대체하려는 경우 이들 특성에 값을 제공할 수 있습니다. 예를 들어, 오브젝트가 LDAP 환경에서 저장되는 경우, **NAME_PREFIX** 특성을 필수 접두부로 설정하여 도구가 오브젝트 및 컨텍스트 이름에 추가하는 기본 접두부를 변경할 수 있습니다.

세 InitialContextFactory 특성의 하나 이상을 생략하는 경우, 관리 도구가 다른 특성의 값을 바탕으로 적당한 기본값을 제공합니다.

2. **PROVIDER_URL** 특성을 구성하여 세션의 초기 컨텍스트의 URL을 정의하십시오.

   이 URL은 도구가 수행하는 모든 JNDI 조작의 루트입니다. 이 특성의 다음 두 가지 양식이 지원됩니다.

   - ldap://hostname/contextname
   - file:[drive:]/pathname

   LDAP URL의 형식은 LDAP 제공자에 따라 다를 수 있습니다. 자세한 정보는 LDAP 문서를 참조하십시오.

3. **SECURITY_AUTHENTICATION** 특성을 구성하여 JNDI가 서비스 제공자에게 보안 신임 정보를 전달하는지 여부를 정의하십시오.

   이 특성은 LDAP 서비스 제공자가 사용될 때만 사용되며 다음 세 값 중 하나를 가질 수 있습니다.

   none(익명 인증)
   이 매개변수를 none으로 설정하는 경우, JNDI는 서비스 제공자로 어떤 보안 신임 정보도 전달하지 않으며 익명 인증이 수행됩니다.

   simple(단순 인증)
   매개변수를 `simple`로 설정하는 경우, 보안 신임 정보가 JNDI를 통해 기본 서비스 제공자로 전달됩니다. 이 보안 신임 정보는 사용자 식별 이름(사용자 DN)과 비밀번호로 구성됩니다.

   CRAM-MD5(CRAM-MD5 인증 메커니즘)
   매개변수를 `CRAM-MD5`로 설정하는 경우, 보안 신임 정보가 JNDI를 통해 기본 서비스 제공자로 전달됩니다. 이 보안 신임 정보는 사용자 식별 이름(사용자 DN)과 비밀번호로 구성됩니다.

   SECURITY_AUTHENTICATION 특성에 대한 올바른 값을 제공하지 않는 경우 특성은 none으로 기본 설정됩니다.

   보안 신임 정보가 필수인 경우 도구가 초기화할 때 해당 정보가 프롬프트됩니다. JMSAdmin 구성 파일에서 **PROVIDER_USERDN** 및 **PROVIDER_PASSWORD** 특성을 설정하여 이를 피할 수 있습니다.

   참고: 이들 특성을 사용하지 않는 경우 비밀번호를 포함하여 입력된 텍스트가 화면에 반향됩니다. 이것은 보안 영향을 가질 수 있습니다.

   도구는 자신을 인증하지 않습니다. 즉 인증 태스크가 LDAP 서버에 위임됩니다. LDAP 서버 관리자는 디렉토리의 다른 파트에 대한 액세스 권한을 설정하고 유지보수해야 합니다. 자세한 정보는 LDAP 문서를 참조하십시오. 인증이 실패하면 도구는 적절한 오류 메시지를 표시하고 종료합니다.

   보안 및 JNDI에 관한 자세한 내용은 Oracle의 Java 웹 사이트(Oracle Technology Network for Java Developers)의 문서에 있습니다.

관리 도구 시작

관리 도구는 대화식으로 또는 배치 프로세스를 시작하기 위해 사용할 수 있는 명령행 인터페이스를 갖습니다.

이 태스크 정보

대화식 모드는 관리 명령을 입력할 수 있는 명령 프롬프트를 제공합니다. 배치 모드에서 도구를 시작하는 명령은 관리 명령 스크립트를 포함하는 파일의 이름을 포함합니다.

프로시저

대화식 모드

- 대화식 모드에서 도구를 시작하려면 다음 명령을 입력하십시오.

```
JMSAdmin [-t] [-v] [-cfg config_filename]
```

설명:

-t
추적을 사용으로 설정합니다(기본값은 추적 끄기입니다).

추적 파일이 "%MQ_JAVA_DATA_PATH%"\errors(Windows)에 생성됩니다. 또는 /var/mqm/trace(UNIX)입니다. 추적 파일의 이름은 다음과 같습니다.

mqjms_*PID*.trc

여기서 *PID*는 JVM의 프로세스 ID입니다.

-v
상세 출력을 생성합니다(기본값은 간결 출력입니다).

-cfg config_filename
대체 구성 파일의 이름을 지정합니다. 이 매개변수가 생략되면 기본 구성 파일인 JMSAdmin.config가 사용됩니다. 구성 파일에 대한 자세한 정보는 515 페이지의 『JMS 관리 도구 구성』의 내용을 참조하십시오.

명령 프롬프트가 표시되며, 도구가 관리 명령을 허용할 준비가 되었음을 표시합니다. 이 프롬프트는 초기에 다음과 같이 나타납니다.

```
InitCtx>
```

현재 컨텍스트(즉, 모든 네이밍 및 디렉토리 조작이 현재 참조하는 JNDI 컨텍스트)가 **PROVIDER_URL** 구성 매개변수에서 정의되는 초기 컨텍스트임을 표시합니다. 이 매개변수에 대한 자세한 정보는 515 페이지의 『JMS 관리 도구 구성』의 내용을 참조하십시오.

디렉토리 네임스페이스를 가로지를 때, 프롬프트는 항상 현재 컨텍스트를 표시하도록 프롬프트가 이를 반영하도록 변합니다.

배치 모드

- 배치 모드에서 도구를 시작하려면 다음 명령을 입력하십시오.

```
JMSAdmin test.scp
```

여기서 *test.scp*는 관리 명령을 포함하는 스크립트 파일입니다. 자세한 정보는 517 페이지의 『관리 명령 사용』의 내용을 참조하십시오. 파일의 마지막 명령은 END 명령이어야 합니다.

관리 명령 사용

관리 도구는 관리 verb와 그의 적절한 매개변수로 구성된 명령을 허용합니다.

이 태스크 정보

다음 표는 관리 도구로 명령을 입력할 때 사용할 수 있는 관리 verb를 나열한 것입니다.

표 *34.* 관리 *verb*

동사	축약형	설명
ALTER	ALT	관리 대상 오브젝트의 특성 중 하나 이상을 변경합니다.
DEFINE	DEF	관리 대상 오브젝트를 작성 및 저장하거나, 하위 컨텍스트를 작성합니다.
DISPLAY	DIS	저장된 하나 이상의 관리 대상 오브젝트의 특성 또는 현재 컨텍스트의 컨텐츠를 표시합니다.
DELETE	DEL	하나 이상의 관리 대상 오브젝트를 이름 공간에서 제거하거나 비어 있는 하위 컨텍스트를 제거합니다.
CHANGE	CHG	사용자가 초기 컨텍스트 아래 임의 위치에 있는 디렉토리 이름 공간을 탐색할 수 있도록 현재 컨텍스트를 변경합니다(보안 기준 준수).
COPY	CP	저장된 관리 대상 오브젝트의 사본을 작성하여 다른 이름으로 저장합니다.
MOVE	MV	관리 대상 오브젝트가 저장되는 이름을 변경합니다.
END		관리 도구를 닫습니다.

프로시저

- 관리 도구가 아직 시작되지 않은 경우, 517 페이지의 『관리 도구 시작』의 설명에 따라 시작하십시오.

  도구가 관리 명령을 허용할 준비가 되었음을 표시하는 명령 프롬프트가 표시됩니다. 이 프롬프트는 초기에 다음과 같이 나타납니다.

  ```
  InitCtx>
  ```

  현재 컨텍스트를 변경하려면 519 페이지의 『하위 컨텍스트 구성』에서 설명하는 대로 CHANGE verb를 사용하십시오.
- 다음 양식으로 명령을 입력하십시오.

  ```
  verb [param]*
  ```

  여기서 **verb**는 518 페이지의 표 34에 나열된 관리 verb 중 하나입니다. 모든 유효한 명령은 하나의 verb를 포함하는데, 이것은 명령의 시작에 표준 또는 단축 양식으로 나타납니다. Verb 이름은 대소문자를 구분하지 않습니다.
- 명령을 종료하려면 여러 명령을 함께 입력하려는 경우가 아니면 Enter를 누르십시오. 여러 명령을 입력하는 경우 Enter를 누르기 직전에 더하기 부호(+)를 입력하십시오.

  일반적으로 명령을 종료하려면 Enter를 누릅니다. 그러나 Enter를 누르기 직전에 더하기 부호(+)를 입력하여 이를 대체할 수 있습니다. 그러면 다음 예에서 보는 것처럼 다중행 명령을 입력할 수 있습니다.

  ```
  DEFINE Q(BookingsInputQueue) +
  QMGR(QM.POLARIS.TEST) +
  QUEUE(BOOKINGS.INPUT.QUEUE) +
  PORT(1415) +
  CCSID(437)
  ```
- 관리 도구를 닫으려면 **END** verb를 사용하십시오.

  이 verb는 어떤 매개변수도 가질 수 없습니다.

하위 컨텍스트 구성

CHANGE, **DEFINE**, **DISPLAY** 및 **DELETE** verb를 사용하여 디렉토리 네임스페이스 하위 컨텍스트를 구성할 수 있습니다.

이 태스크 정보

이들 verb의 사용은 다음 표에서 설명되어 있습니다.

표 *35.* 하위 컨텍스트를 조작하는 데 사용되는 명령 구문 및 설명

명령 구문	설명
DEFINE CTX(ctxName)	이름이 ctxName인 현재 컨텍스트의 하위 컨텍스트를 작성하려고 시도합니다. 보안 위반이 있는 경우, 하위 컨텍스트가 이미 존재하는 경우 또는 제공되는 이름이 올바르지 않은 경우 실패합니다.
DISPLAY CTX	현재 컨텍스트의 컨텐츠를 표시합니다. 관리 대상 오브젝트가 [D]를 갖는 a 하위 컨텍스트로 어노테이션됩니다. 각 오브젝트의 Java 유형도 표시됩니다.
DELETE CTX(ctxName)	이름이 ctxName인 현재 컨텍스트의 하위 컨텍스트를 삭제하려고 시도합니다. 해당 컨텍스트가 없거나 비어 있지 않거나 보안 위반이 있는 경우 실패합니다.
CHANGE CTX(ctxName)	현재 컨텍스트가 이름이 ctxName인 하위 컨텍스트를 참조하도록 변경합니다. ctxName의 두 특수 값 중 하나가 제공될 수 있습니다. **=UP** 현재 컨텍스트의 상위로 이동합니다. **=INIT** 초기 컨텍스트로 직접 이동합니다. 지정된 컨텍스트가 없거나 보안 위반이 있는 경우 실패합니다.

LDAP 환경에 저장되는 JMS 오브젝트의 이름은 LDAP 이름 지정 규칙을 준수해야 합니다. 이들 규칙 중 하나는 오브젝트 및 컨텍스트 이름이 cn=(공통 이름) 또는 ou=(조직 단위) 같은 접두부를 포함해야 한다는 것입니다. 관리 도구를 사용하면 접두부 없이 오브젝트 및 컨텍스트 이름을 나타낼 수 있어 LDAP 서비스 제공자를 쉽게 사용할 수 있습니다. 접두부를 제공하지 않으면 도구가 자동으로 사용자가 제공하는 이름에 기본 접두부를 추가합니다. LDAP의 경우, cn=입니다. 필요한 경우 구성 파일에서 **NAME_PREFIX** 특성을 설정하여 기본 접두부를 변경할 수 있습니다. 추가 정보는 515 페이지의 『JMS 관리 도구 구성』의 내용을 참조하십시오.

참고: Java 오브젝트를 저장하려면 LDAP 서버를 구성해야 할 수 있습니다. 자세한 정보는 LDAP 서버 문서를 참조하십시오.

JMS 오브젝트 작성

JMS 연결 팩토리 및 목적지 오브젝트를 작성하고 이들을 JNDI 네임스페이스에 저장하려면 DEFINE verb를 사용하십시오. 오브젝트를 LDAP 환경에 저장하려면 특정 규칙을 따르는 이름을 제공해야 합니다. 관리 도구는 오브젝트 이름에 기본 접두부를 추가하여 LDAP 이름 지정 규칙을 따르는 데 도움이 될 수 있습니다.

이 태스크 정보

DEFINE verb는 사용자가 지정하는 유형, 이름 및 특성을 갖는 관리 대상 오브젝트를 작성합니다. 새 오브젝트는 현재 컨텍스트에 저장됩니다.

LDAP 환경에 저장되는 JMS 오브젝트의 이름은 LDAP 이름 지정 규칙을 준수해야 합니다. 이들 규칙 중 하나는 오브젝트 및 컨텍스트 이름이 cn=(공통 이름) 또는 ou=(조직 단위) 같은 접두부를 포함해야 한다는 것입니다. 관리 도구를 사용하면 접두부 없이 오브젝트 및 컨텍스트 이름을 나타낼 수 있어 LDAP 서비스 제공자를 쉽게 사용할 수 있습니다. 접두부를 제공하지 않으면 도구가 자동으로 사용자가 제공하는 이름에 기본 접두부를 추가합니

다. LDAP의 경우, cn=입니다. 필요한 경우 구성 파일에서 **NAME_PREFIX** 특성을 설정하여 기본 접두부를 변경할 수 있습니다. 추가 정보는 515 페이지의 『JMS 관리 도구 구성』의 내용을 참조하십시오.

참고: Java 오브젝트를 저장하려면 LDAP 서버를 구성해야 할 수 있습니다. 자세한 정보는 LDAP 서버 문서를 참조하십시오.

프로시저

1. 관리 도구가 아직 시작되지 않은 경우, 517 페이지의 『관리 도구 시작』의 설명에 따라 시작하십시오.
   도구가 관리 명령을 허용할 준비가 되었음을 표시하는 명령 프롬프트가 표시됩니다.
2. 명령 프롬프트가 새 오브젝트를 작성하려는 컨텍스트를 표시하고 있는지 확인하십시오.
   관리 도구를 시작할 때 프롬프트가 초기에 다음과 같이 나타납니다.

   ```
   InitCtx>
   ```

   현재 컨텍스트를 변경하려면 519 페이지의 『하위 컨텍스트 구성』에서 설명하는 대로 CHANGE verb를 사용하십시오.
3. 연결 팩토리, 큐 목적지 또는 토픽 목적지를 작성하려면 다음 명령 구문을 사용하십시오.

   ```
   DEFINE TYPE (name) [property]*
   ```

   즉, DEFINE verb, *TYPE* (name) 관리 대상 오브젝트 참조, 0개 이상의 특성을 차례로 입력하십시오(IBM MQ classes for JMS 오브젝트의 특성을 참조하십시오).
4. 연결 팩토리, 큐 목적지 또는 토픽 목적지를 작성하려면 다음 명령 구문을 사용하십시오.

   ```
   DEFINE TYPE (name) [property]*
   ```

5. 새로 작성된 오브젝트를 표시하려면 DISPLAY verb를 다음 명령 구문과 함께 사용하십시오.

   ```
   DISPLAY TYPE (name)
   ```

예

다음 예는 DEFINE verb를 사용하여 초기 컨텍스트에서 작성된 testQueue라는 큐를 표시합니다. 이 오브젝트는 LDAP 환경에 저장되고 있으므로, 오브젝트 이름 `testQueue`가 접두부와 함께 입력되지 않더라도 도구는 LDAP 이름 지정 규칙을 준수하도록 자동으로 하나를 추가합니다. 명령 `DISPLAY Q(testQueue)`를 제출해도 이 접두부가 추가됩니다.

```
InitCtx> DEFINE Q(testQueue)

InitCtx> DISPLAY CTX

Contents of InitCtx

a cn=testQueue       com.ibm.mq.jms.MQQueue

1 Object(s)
0 Context(s)
1 Binding(s), 1 Administered
```

JMS 오브젝트를 작성하는 샘플 오류 조건

오브젝트를 작성할 때 많은 공통 오류 조건이 발생할 수 있습니다.

다음은 이러한 오류 조건의 예입니다.

CipherSpec이 CipherSuite에 맵핑됨

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) SSLCIPHERSUITE(RC4_MD5_US)
```

```
WARNING: Converting CipherSpec RC4_MD5_US to
CipherSuite SSL_RSA_WITH_RC4_128_MD5
```

오브젝트에 대한 올바르지 않은 특성

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) PRIORITY(4)
Unable to create a valid object, please check the parameters supplied
Invalid property for a QCF: PRI
```

특성 값에 대한 올바르지 않은 유형

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) CCSID(english)
Unable to create a valid object, please check the parameters supplied
Invalid value for CCS property: English
```

특성 충돌 - 클라이언트/바인딩

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) HOSTNAME(polaris.hursley.ibm.com)
Unable to create a valid object, please check the parameters supplied
Invalid property in this context: Client-bindings attribute clash
```

특성 충돌 - 초기화 종료

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) SECEXITINIT(initStr)
Unable to create a valid object, please check the parameters supplied
Invalid property in this context: ExitInit string supplied
without Exit string
```

특성 값이 올바른 범위를 벗어남

```
InitCtx/cn=Trash> DEFINE Q(testQ) PRIORITY(12)
Unable to create a valid object, please check the parameters supplied
Invalid value for PRI property: 12
```

알 수 없는 특성

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) PIZZA(ham and mushroom)
Unable to create a valid object, please check the parameters supplied
Unknown property: PIZZA
```

다음은 JMS 애플리케이션에서 JNDI 관리 대상 오브젝트를 찾을 때 Windows에서 발생할 수 있는 오류 조건의 예입니다.

1. WebSphere JNDI 제공자인 com.ibm.websphere.naming.WsnInitialContextFactory를 사용 중인 경우, 포워드 슬래시(/)를 사용하여 하위 컨텍스트에서 정의된 관리 대상 오브젝트(예: jms/MyQueueName)에 액세스해야 합니다. 백슬래시(\)를 사용하는 경우 InvalidNameException이 발생합니다.
2. Oracle JNDI 제공자인 com.sun.jndi.fscontext.RefFSContextFactory를 사용 중인 경우 백슬래시(\)를 사용하여 하위 컨텍스트에서 정의된 관리 대상 오브젝트(예: ctx1\\fred)에 액세스해야 합니다. 포워드 슬래시(/)를 사용하는 경우 NameNotFoundException이 발생합니다.

JMS 오브젝트 구성

동사 ALTER, DEFINE, DISPLAY, DELETE, COPY, MOVE를 사용하여 디렉토리 네임스페이스의 관리 대상 오브젝트를 조작할 수 있습니다.

이 태스크 정보

522 페이지의 표 36에서는 이들 동사의 사용을 요약한 것입니다. *TYPE* 을 514 페이지의 『관리 도구를 사용하여 JMS 오브젝트 구성』에서 설명한 대로 필수 관리 대상 오브젝트를 나타내는 키워드로 대체하십시오.

표 36. 관리 대상 오브젝트를 조작하는 데 사용되는 명령의 구문 및 설명	
명령 구문	**설명**
ALTER *TYPE* (name) [property]*	관리 대상 오브젝트의 특성을 제공되는 것으로 업데이트하려고 합니다. 보안 위반이 있는 경우, 지정된 오브젝트를 찾을 수 없는 경우 또는 제공되는 새 특성이 올바르지 않은 경우 실패합니다.
DEFINE *TYPE* (name) [property]*	제공되는 특성을 갖는 *TYPE* 유형의 관리 대상 오브젝트를 작성하고 이를 현재 컨텍스트의 name 이름 아래에 저장하려고 합니다. 보안 위반이 있는 경우, 제공되는 이름이 올바르지 않은 경우 또는 제공되는 특성이 올바르지 않은 경우 실패합니다.
DISPLAY *TYPE* (name)	현재 컨텍스트의 이름 name 아래에 바운드된 *TYPE* 유형의 관리 대상 오브젝트의 특성을 표시합니다. 오브젝트가 없거나 보안 위반이 있는 경우 실패합니다.
DELETE *TYPE* (name)	현재 컨텍스트가 이름이 `name`인 *TYPE* 유형의 관리 대상 오브젝트를 제거하려고 합니다. 오브젝트가 없거나 보안 위반이 있는 경우 실패합니다.
COPY *TYPE* (nameA) *TYPE* (nameB)	이름이 `nameA`인 *TYPE* 유형의 관리 대상 오브젝트 사본을 작성하고, 사본 이름을 `nameB`로 지정합니다. 모두 현재 컨텍스트 범위에서 발생합니다. 복사될 오브젝트가 존재하지 않는 경우, 이름 nameB의 오브젝트가 존재하는 경우 또는 보안 위반이 있는 경우 실패합니다.
MOVE *TYPE* (nameA) *TYPE* (nameB)	이름이 `nameA`인 *TYPE* 유형의 관리 대상 오브젝트를 `nameB`로 이동(이름 바꾸기)합니다. 모두 현재 컨텍스트 범위에서 발생합니다. 이동될 오브젝트가 존재하지 않는 경우, 이름 nameB의 오브젝트가 존재하는 경우 또는 보안 위반이 있는 경우 실패합니다.

WebSphere Application Server에서 JMS 자원 구성

WebSphere Application Server에서 JMS 자원을 구성하려면 관리 콘솔이나 wsadmin 명령 중 하나를 사용할 수 있습니다.

이 태스크 정보

JMS(Java Message Service) 애플리케이션은 일반적으로 애플리케이션이 그의 JMS 제공자에게 연결하는 방법 및 액세스하는 목적지를 설명하는 외부에서 구성된 오브젝트에 의존합니다. JMS 애플리케이션은 JNDI(Java Naming Directory Interface)를 사용하여 런타임 시에 다음 유형의 오브젝트에 액세스합니다.

- 활성화 스펙(Java EE 애플리케이션 서버에서 사용)
- 연합 연결 팩토리(JMS 1.1에서, 도메인 독립(연합) 연결 팩토리는 도메인 특정 큐 연결 팩토리와 토픽 연결 팩토리에 선호됨)
- 토픽 연결 팩토리(JMS 1.0 애플리케이션에서 사용)
- 큐 연결 팩토리(JMS 1.0 애플리케이션에서 사용)
- 큐
- 토픽

WebSphere Application Server의 IBM MQ 메시징 제공자를 통해, JMS(Java Message Service) 메시징 애플리케이션이 사용자의 IBM MQ 시스템을 JMS 메시징 자원의 외부 제공자로서 사용할 수 있습니다. 이 방식을 사용할 수 있기 위해서는 IBM MQ 네트워크의 모든 큐 관리자에 연결하기 위한 JMS 자원을 정의하도록 WebSphere Application Server의 IBM MQ 메시징 제공자를 구성합니다.

WebSphere Application Server를 사용하여 애플리케이션(예: 큐 연결 팩토리)에 대한 IBM MQ 자원을 구성하고 JMS 목적지와 연관된 메시지 및 구독을 관리할 수 있습니다. IBM MQ를 통해 보안을 관리합니다.

WebSphere Application Server 버전 8.5.5에 대한 관련 정보

IBM MQ 메시징 제공자를 사용한 상호 운용

IBM MQ 메시징 제공자를 사용한 메시징 관리

관리 콘솔 패널 이름을 명령 이름 및 IBM MQ 이름에 맵핑

WebSphere Application Server 8.0 관련 정보

IBM MQ 메시징 제공자를 사용한 상호 운용

IBM MQ 메시징 제공자를 사용한 메시징 관리

관리 콘솔 패널 이름을 명령 이름 및 IBM MQ 이름에 맵핑

WebSphere Application Server 7.0 관련 정보

IBM MQ 메시징 제공자를 사용한 상호 운용

IBM MQ 메시징 제공자를 사용한 메시징 관리

관리 콘솔 패널 이름을 명령 이름 및 IBM MQ 이름에 맵핑

관리 콘솔을 사용한 JMS 자원 구성

WebSphere Application Server 관리 콘솔을 사용하여 IBM MQ JMS 제공자에 대한 활성화 스펙, 연결 팩토리 및 목적지를 구성할 수 있습니다.

이 태스크 정보

WebSphere Application Server 관리 콘솔을 사용하여 다음 자원을 작성, 보기 또는 수정할 수 있습니다.

- 활성화 스펙
- 도메인 독립적 연결 팩토리(JMS 1.1 이상)
- 큐 연결 팩토리
- 토픽 연결 팩토리
- 큐
- 토픽

다음 단계는 관리 콘솔을 사용하여 IBM MQ 메시징 제공자와 함께 사용할 JMS 자원을 구성할 수 있는 방법의 개요를 제공합니다. 각 단계는 추가 정보를 위해 참조할 수 있는 WebSphere Application Server 제품 문서에 있는 주제의 이름을 포함합니다. WebSphere Application Server 8.5.5, 8.0 및 7.0 제품 문서의 해당 주제에 대한 링크는 관련 링크 를 참조하십시오.

혼합 버전 WebSphere Application Server 셀에서는 모든 버전의 노드의 IBM MQ 자원을 관리할 수 있습니다. 그러나, 일부 특성은 모든 버전에 사용 가능하지 않습니다. 이 경우, 해당 특정 노드의 특성만 관리 콘솔에 표시됩니다.

프로시저

IBM MQ 메시징 제공자와 함께 사용할 활성화 스펙을 작성 또는 구성하려면 다음을 수행하십시오.

- 활성화 스펙을 작성하려면 IBM MQ JMS 자원 작성 마법사를 사용하십시오.

  마법사를 사용하여 활성화 스펙에 대한 모든 세부사항을 지정하거나, 클라이언트 채널 정의 테이블(CCDT)을 사용하여 IBM MQ에 대한 연결 세부사항을 지정할 것을 선택할 수 있습니다. 마법사를 사용하여 연결 세부사항을 지정할 때, 호스트 및 포트 정보를 분리해서 입력할 것을 선택하거나, 다중 인스턴스 큐 관리자를 사용 중인 경우 연결 이름 목록의 양식으로 호스트 및 포트 정보를 입력할 것을 선택할 수 있습니다. 자세한 정보는 *IBM MQ* 메시징 제공자를 위한 활성화 스펙 작성을 참조하십시오.
- 활성화 스펙의 구성 특성을 보거나 변경하려면 관리 콘솔 IBM MQ 메시징 제공자 연결 팩토리 설정 패널을 사용하십시오.

  이 구성 특성은 연관된 큐 및 토픽에 대한 연결 작성 방법을 제어합니다. 자세한 정보는 *IBM MQ* 메시징 제공자를 위한 활성화 스펙 구성을 참조하십시오.

IBM MQ 메시징 제공자와 함께 사용할 통합 연결 팩토리, 큐 연결 팩토리 또는 토픽 연결 팩토리를 작성 또는 구성하려면 다음을 수행하십시오.

- 연결 팩토리를 작성하려면 먼저 작성할 연결 팩토리의 유형을 선택한 후 IBM MQ JMS 자원 작성 마법사를 사용하여 세부사항을 지정하십시오.
  - JMS 애플리케이션이 포인트-투-포인트 메시징만을 사용하려는 경우, 특별히 포인트-투-포인트 메시징을 위한 연결 작성에 사용될 수 있는 포인트-투-포인트 메시징 도메인에 대한 도메인 특정 연결 팩토리를 작성하십시오.
  - JMS 애플리케이션이 발행/구독 메시징만을 사용하려는 경우, 특별히 발행/구독 메시징을 위한 연결 작성에 사용될 수 있는 발행/구독 메시징 도메인에 대한 도메인 특정 연결 팩토리를 작성하십시오.
  - JMS 1.1 이상의 경우, 애플리케이션이 동일한 트랜잭션에서 포인트-투-포인트 및 발행/구독 작업을 둘 다 수행할 수 있도록 하기 위해 포인트-투-포인트 메시징 및 발행/구독 메시징 둘 다에 사용될 수 있는 도메인 독립적 연결 팩토리를 작성하십시오.

  연결 팩토리에 대한 모든 세부사항을 지정하기 위해 마법사를 사용할지 여부를 선택하거나, 클라이언트 채널 정의 테이블(CCDT)을 사용하여 IBM MQ에 대한 연결 세부사항을 지정할 것을 선택할 수 있습니다. 마법사를 사용하여 연결 세부사항을 지정할 때, 호스트 및 포트 정보를 분리해서 입력할 것을 선택하거나, 다중 인스턴스 큐 관리자를 사용 중인 경우 연결 이름 목록의 양식으로 호스트 및 포트 정보를 입력할 것을 선택할 수 있습니다. 자세한 정보는 *IBM MQ* 메시징 제공자를 위한 연결 팩토리 작성을 참조하십시오.

연결 팩토리의 구성 특성을 보거나 변경하려면 다음을 수행하십시오.

- 구성하려는 연결 팩토리의 유형에 대한 관리 콘솔 연결 팩토리 설정 패널을 사용하십시오.

  구성 특성은 연관된 큐 및 토픽에 대한 연결을 작성하는 방법을 제어합니다. 자세한 정보는 *IBM MQ* 메시징 제공자를 위한 콜렉션 팩토리 구성 또는 *IBM MQ* 메시징 제공자에 대한 큐 콜렉션 팩토리 구성 또는 *IBM MQ* 메시징 제공자에 대한 토픽 콜렉션 팩토리 구성을 참조하십시오.

IBM MQ 메시징 제공자를 사용한 포인트-투-포인트 메시징에 대한 JMS 큐 목적지를 구성하려면 다음을 수행하십시오.

- 관리 콘솔 IBM MQ 메시징 제공자 큐 설정 패널을 사용하여 다음 유형의 특성을 정의하십시오.
  - 관리 및 IBM MQ 큐 특성을 포함한 일반 특성.
  - 큐를 호스팅하는 큐 관리자에 연결하는 방법을 지정하는 연결 특성.
  - IBM MQ 메시징 제공자 목적지에 작성된 연결의 작동을 제어하는 고급 특성.
  - 큐 목적지에 대한 모든 사용자 정의 특성.

  자세한 정보는 *IBM MQ* 메시징 제공자에 대한 큐 구성을 참조하십시오.

IBM MQ 메시징 제공자를 사용하는 발행/구독 메시징에 대한 JMS 토픽 목적지를 작성 또는 구성하려면 다음을 수행하십시오.

- IBM MQ 메시징 제공자 토픽 설정 패널을 사용하여 다음 유형의 특성을 정의하십시오.
  - 관리 및 IBM MQ 토픽 특성을 포함한 일반 특성.
  - IBM MQ 메시징 제공자 목적지에 작성된 연결의 작동을 제어하는 고급 특성.
  - 큐 목적지에 대한 모든 사용자 정의 특성.

  자세한 정보는 *IBM MQ* 메시징 제공자에 대한 토픽 구성을 참조하십시오.

관련 개념

36 페이지의 『클라이언트 채널 정의 테이블』

클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

396 페이지의 『다중 인스턴스 큐 관리자』

다중 인스턴스 큐 관리자는 여러 서버에서 구성된 동일한 큐 관리자의 인스턴스입니다. 큐 관리자의 한 인스턴스는 활성 인스턴스로 정의되고 다른 인스턴스는 대기 인스턴스로 정의됩니다. 활성 인스턴스가 실패하면 다중 인스턴스 큐 관리자가 대기 서버에서 자동으로 시작됩니다.

관련 태스크

326 페이지의 『발행/구독 메시징 구성』

큐 대기된 발행/구독의 상태를 시작, 중지 또는 표시할 수 있습니다. 스트림을 추가하거나 제거할 수 있고 브로커 계층에서 큐 관리자를 추가하고 삭제할 수도 있습니다.

WebSphere Application Server traditional 9.0에 대한 관련 정보
IBM MQ 메시징 제공자 활성화 스펙
IBM MQ 메시징 제공자에 대한 활성화 스펙 작성
IBM MQ 메시징 제공자에 대한 활성화 스펙 구성
IBM MQ 메시징 제공자에 대한 연결 팩토리 작성
IBM MQ 메시징 제공자에 대한 통합 연결 팩토리 구성
IBM MQ 메시징 제공자에 대한 큐 연결 팩토리 구성
IBM MQ 메시징 제공자에 대한 토픽 연결 팩토리 구성
IBM MQ 메시징 제공자에 대한 큐 구성
IBM MQ 메시징 제공자에 대한 토픽 구성
WebSphere Application Server 8.5.5에 대한 관련 정보
IBM WebSphere MQ 메시징 제공자 활성화 스펙
IBM WebSphere MQ 메시징 제공자에 대한 활성화 스펙 작성
IBM WebSphere MQ 메시징 제공자에 대한 활성화 스펙 구성
IBM WebSphere MQ 메시징 제공자에 대한 연결 팩토리 작성
IBM WebSphere MQ 메시징 제공자에 대한 통합 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 큐 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 토픽 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 큐 구성
IBM WebSphere MQ 메시징 제공자에 대한 토픽 구성
WebSphere Application Server 8.0에 대한 관련 정보
IBM WebSphere MQ 메시징 제공자 활성화 스펙
IBM WebSphere MQ 메시징 제공자에 대한 활성화 스펙 작성
IBM WebSphere MQ 메시징 제공자에 대한 활성화 스펙 구성
IBM WebSphere MQ 메시징 제공자에 대한 연결 팩토리 작성
IBM WebSphere MQ 메시징 제공자에 대한 통합 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 큐 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 토픽 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 큐 구성
IBM WebSphere MQ 메시징 제공자에 대한 토픽 구성
WebSphere Application Server 7.0에 대한 관련 정보
IBM WebSphere MQ 메시징 제공자 활성화 스펙
IBM WebSphere MQ 메시징 제공자에 대한 활성화 스펙 작성
IBM WebSphere MQ 메시징 제공자에 대한 활성화 스펙 구성
IBM WebSphere MQ 메시징 제공자에 대한 연결 팩토리 작성
IBM WebSphere MQ 메시징 제공자에 대한 통합 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 큐 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 토픽 연결 팩토리 구성
IBM WebSphere MQ 메시징 제공자에 대한 큐 구성
IBM WebSphere MQ 메시징 제공자에 대한 토픽 구성

wsadmin 스크립트 명령을 사용한 JMS 자원 구성

WebSphere Application Server wsadmin 스크립트 명령을 사용하여 JMS 활성화 스펙, 연결 팩토리, 큐 및 토픽에 관한 정보를 작성, 수정, 삭제 또는 표시할 수 있습니다. 또한 IBM MQ 자원 어댑터에 대한 설정을 표시 및 관리할 수 있습니다.

이 태스크 정보

다음 단계는 WebSphere Application Server wsadmin 명령을 사용하여 IBM MQ 메시징 제공자와 함께 사용할 JMS 자원을 구성할 수 있는 방법의 개요를 제공합니다. 이러한 명령을 사용하는 방법에 대한 자세한 정보는 WebSphere Application Server 8.5.5, 8.0 및 7.0 제품 문서에 대한 링크는 관련 링크 를 참조하십시오.

명령을 실행하려면 wsadmin 스크립트 클라이언트의 AdminTask 오브젝트를 사용하십시오.

명령을 사용하여 새 오브젝트를 작성하거나 변경을 작성한 후, 변경을 마스터 구성에 저장하십시오. 예를 들면, 다음 명령을 사용하십시오.

```
AdminConfig.save()
```

사용 가능한 IBM MQ 메시징 제공자 관리 명령과 각 명령의 간략한 설명을 보려면 wsadmin 프롬프트에 다음 명령을 입력하십시오.

```
print AdminTask.help('WMQAdminCommands')
```

주어진 명령에 대한 개요 도움말을 보려면 wsadmin 프롬프트에 다음 명령을 입력하십시오.

```
print AdminTask.help('command_name')
```

프로시저

명령이 실행되는 범위에서 정의되는 모든 IBM MQ 메시징 제공자 자원을 나열하려면 다음 명령을 사용하십시오.

- 활성화 스펙을 나열하려면 **listWMQActivationSpecs** 명령을 사용하십시오.
- 연결 팩토리를 나열하려면 **listWMQConnectionFactories** 명령을 사용하십시오.
- 큐 유형 목적지를 나열하려면 **listWMQQueues** 명령을 사용하십시오.
- 토픽 유형 목적지를 나열하려면 **listWMQTopics** 명령을 사용하십시오.

특정 범위에서 IBM MQ 메시징 제공자에 대한 JMS 자원을 작성하려면 다음 명령을 사용하십시오.

- 활성화 스펙을 작성하려면 **createWMQActivationSpec** 명령을 사용하십시오.

  연결 설정에 사용될 모든 매개변수를 지정하여 활성화 스펙을 작성하거나, 클라이언트 채널 정의 테이블 (CCDT)을 사용하여 연결할 큐 관리자를 찾도록 활성화 스펙을 작성할 수 있습니다.
- 연결 팩토리를 작성하려면 **createWMQConnectionFactory** 명령을 사용하고 **-type** 매개변수를 사용하여 작성하려는 연결 팩토리의 유형을 지정하십시오.
  - JMS 애플리케이션이 포인트-투-포인트 메시징만을 사용하려는 경우, 특별히 포인트-투-포인트 메시징을 위한 연결 작성에 사용될 수 있는 포인트-투-포인트 메시징 도메인에 대한 도메인 특정 연결 팩토리를 작성하십시오.
  - JMS 애플리케이션이 발행/구독 메시징만을 사용하려는 경우, 특별히 발행/구독 메시징을 위한 연결 작성에 사용될 수 있는 발행/구독 메시징 도메인에 대한 도메인 특정 연결 팩토리를 작성하십시오.
  - JMS 1.1 이상의 경우, 애플리케이션이 동일한 트랜잭션에서 포인트-투-포인트 및 발행/구독 작업을 둘 다 수행할 수 있도록 하기 위해 포인트-투-포인트 메시징 및 발행/구독 메시징 둘 다에 사용될 수 있는 도메인 독립적 연결 팩토리를 작성하십시오.

  기본 유형은 도메인 독립적 연결 팩토리입니다.
- 큐 유형 목적지를 작성하려면 **createWMQQueue** 명령을 사용하십시오.
- 토픽 유형 목적지를 작성하려면 **createWMQTopic** 명령을 사용하십시오.

특정 범위에 있는 IBM MQ 메시징 제공자에 대한 JMS 자원을 수정하려면 다음 명령을 사용하십시오.

- 활성화 스펙을 수정하려면 **modifyWMQActivationSpec** 명령을 사용하십시오.

  활성화 스펙의 유형을 변경할 수 없습니다. 예를 들어, 모든 구성 정보를 수동으로 입력하는 활성화 스펙을 작성한 후 CCDT를 사용하도록 수정할 수 없습니다.

- 연결 팩토리를 수정하려면 **modifyWMQConnectionFactory** 명령을 사용하십시오.
- 큐 유형 목적지를 수정하려면 **modifyWMQQueue** 명령을 사용하십시오.
- 토픽 유형 목적지를 수정하려면 **modifyWMQTopic** 명령을 사용하십시오.

특정 범위에서 IBM MQ 메시징 제공자에 대한 JMS 자원을 삭제하려면 다음 명령을 사용하십시오.

- 활성화 스펙을 삭제하려면 **deleteWMQActivationSpec** 명령을 사용하십시오.
- 연결 팩토리를 삭제하려면 **deleteWMQConnectionFactory** 명령을 사용하십시오.
- 큐 유형 목적지를 삭제하려면 **deleteWMQQueue** 명령을 사용하십시오.
- 토픽 유형 목적지를 삭제하려면 **deleteWMQTopic** 명령을 사용하십시오.

특정 IBM MQ 메시징 제공자 자원에 관한 정보를 표시하려면 다음 명령을 사용하십시오.

- 특정 활성화 스펙과 연관된 모든 매개변수 및 해당 값을 표시하려면 **showWMQActivationSpec** 명령을 사용하십시오.
- 특정 연결 팩토리와 연관된 모든 매개변수 및 해당 값을 표시하려면 **showWMQConnectionFactory** 명령을 사용하십시오.
- 특정 큐 유형 목적지와 연관된 모든 매개변수 및 해당 값을 표시하려면 **showWMQQueue** 명령을 사용하십시오.
- 토픽 유형 목적지와 연관된 모든 매개변수 및 해당 값을 표시하려면 **deleteWMQTopic** 명령을 사용하십시오.

IBM MQ 자원 어댑터 또는 IBM MQ 메시징 제공자에 대한 설정을 관리하려면 다음 명령을 사용하십시오.

- 특정 범위에서 설치되는 IBM MQ 자원 어댑터의 설정을 관리하려면 **manageWMQ** 명령을 사용하십시오.
- **manageWMQ** 명령으로 설정할 수 있는 모든 매개변수 및 해당 값을 표시하려면 **showWMQ** 명령을 사용하십시오. 이러한 설정은 IBM MQ 자원 어댑터 또는 IBM MQ 메시징 제공자와 관련됩니다. **showWMQ** 명령은 또한 IBM MQ 자원 어댑터에서 설정되는 모든 사용자 정의 특성도 표시합니다.

관련 개념

36 페이지의 『클라이언트 채널 정의 테이블』
클라이언트 채널 정의 테이블(CCDT)은 클라이언트 애플리케이션이 큐 관리자에 연결하기 위해 사용하는 인증 정보 및 채널 정의를 판별합니다. 멀티플랫폼에서 CCDT는 자동으로 작성됩니다. 그러면 이 CCDT를 클라이언트 애플리케이션이 사용할 수 있도록 설정해야 합니다.

396 페이지의 『다중 인스턴스 큐 관리자』
다중 인스턴스 큐 관리자는 여러 서버에서 구성된 동일한 큐 관리자의 인스턴스입니다. 큐 관리자의 한 인스턴스는 활성 인스턴스로 정의되고 다른 인스턴스는 대기 인스턴스로 정의됩니다. 활성 인스턴스가 실패하면 다중 인스턴스 큐 관리자가 대기 서버에서 자동으로 시작됩니다.

관련 태스크

326 페이지의 『발행/구독 메시징 구성』
큐 대기된 발행/구독의 상태를 시작, 중지 또는 표시할 수 있습니다. 스트림을 추가하거나 제거할 수 있고 브로커 계층에서 큐 관리자를 추가하고 삭제할 수도 있습니다.

WebSphere Application Server 버전 8.5.5에 대한 관련 정보

createWMQActivationSpec 명령

createWMQConnectionFactory 명령

createWMQQueue 명령

createWMQTopic 명령

deleteWMQActivationSpec 명령

deleteWMQConnectionFactory 명령

deleteWMQQueue 명령

deleteWMQTopic 명령

listWMQActivationSpecs 명령

listWMQConnectionFactories 명령

listWMQQueues 명령

`listWMQTopics` 명령
`modifyWMQActivationSpec` 명령
`modifyWMQConnectionFactory` 명령
`modifyWMQQueue` 명령
`modifyWMQTopic` 명령
`showWMQActivationSpec` 명령
`showWMQConnectionFactory` 명령
`showWMQQueue` 명령
`showWMQTopic` 명령
`showWMQ` 명령
`manageWMQ` 명령

WebSphere Application Server 7.0에 대한 관련 정보

`createWMQActivationSpec` 명령
`createWMQConnectionFactory` 명령
`createWMQQueue` 명령
`createWMQTopic` 명령
`deleteWMQActivationSpec` 명령
`deleteWMQConnectionFactory` 명령
`deleteWMQQueue` 명령
`deleteWMQTopic` 명령
`listWMQActivationSpecs` 명령
`listWMQConnectionFactories` 명령
`listWMQQueues` 명령
`listWMQTopics` 명령
`modifyWMQActivationSpec` 명령
`modifyWMQConnectionFactory` 명령
`modifyWMQQueue` 명령
`modifyWMQTopic` 명령
`showWMQActivationSpec` 명령
`showWMQConnectionFactory` 명령
`showWMQQueue` 명령
`showWMQTopic` 명령
`showWMQ` 명령
`manageWMQ` 명령

JMS 2.0 공유 구독 사용

WebSphere Application Server traditional 9.0에서 IBM MQ 9.0과 함께 JMS 2.0 공유 구독을 구성 및 사용할 수 있습니다.

이 태스크 정보

JMS 2.0 스펙에서는 한 명 이상의 사용자가 단일 구독을 열 수 있는 공유 구독 개념을 소개합니다. 메시지는 모든 사용자가 공유합니다. 사용자가 동일한 큐 관리자에 연결되어 있는 동안 제한이 없습니다.

공유 구독은 비공유 구독을 참조하는 동일한 시맨틱을 사용하여 지속 가능 또는 지속 불가능할 수 있습니다.

사용자가 사용할 구독을 식별하려면 구독 이름을 제공해야 합니다. 비공유 지속 가능 구독과 유사하지만 공유 구독이 필요한 모든 경우에 구독 이름이 필요합니다. 그러나 지속 가능한 공유 구독의 경우 클라이언트 ID는 필요하지 않습니다. 제공할 수는 있지만 필수는 아닙니다

공유 구독을 로드 밸런싱 메커니즘으로 생각할 수 있지만 IBM MQ 또는 JMS 2.0 스펙에서는 사용자간에 메시지를 분배하는 방식에 대한 기능이 없습니다.

WebSphere Application Server traditional 9.0 내에서 IBM MQ 9.0 자원 어댑터가 사전 설치됩니다.

다음 단계에서는 WebSphere Application Server traditional 관리 콘솔을 사용하여 공유 지속 가능 또는 공유 지속 불가능 구독을 사용하도록 활성화 스펙을 구성하는 방법을 보여줍니다.

프로시저

먼저 JNDI에서 오브젝트를 작성하십시오.

1. JNDI에서 정상적으로 토픽 목적지를 작성하십시오(523 페이지의 『관리 콘솔을 사용한 JMS 자원 구성』 참조).
2. 활성화 스펙을 작성하십시오(523 페이지의 『관리 콘솔을 사용한 JMS 자원 구성』 참조).

   필요한 특성을 사용하여 활성화 스펙을 작성할 수 있습니다. 지속 가능한 구독을 사용할 경우 작성시 이를 선택하고 이름을 지정할 수 있습니다. 지속 불가능 구독을 사용할 경우 지금 이름을 지정할 수 없습니다. 그렇지 않으면, 구독 이름에 대한 사용자 정의 특성을 작성해야 합니다.

필수 사용자 정의 특성을 사용하여 작성한 활성화 스펙을 업데이트하십시오. 지정해야 하는 두 개의 사용자 정의 특성이 있습니다.

- 모든 경우 사용자 정의 특성을 작성하여 이 활성화 스펙에서 공유 구독을 사용하도록 지정하십시오.
- 구독이 지속 불가능 구독으로 작성된 경우 구독 이름 특성을 사용자 정의 특성으로 설정해야 합니다.

다음 표에서는 각 사용자 정의 특성에 지정할 수 있는 유효한 값을 보여줍니다.

특성 이름	유형	올바른 값
sharedSubscription	문자열	true, false
subscriptionName	문자열	0이외 길이의 Java 문자열

3. **활성화 스펙 콜렉션** 양식에 표시된 목록에서 활성화 스펙을 선택하십시오.

   활성화 스펙의 세부 정보가 **IBM MQ 메시징 제공자 활성자 스펙 설정** 양식에 표시됩니다.
4. **IBM MQ 메시징 제공자 활성화 스펙 설정** 양식에서 **사용자 정의 속성**을 클릭하십시오.

   사용자 정의 특성 양식이 표시됩니다.
5. 지속 불가능 구독을 사용할 경우 subscriptionName 사용자 정의 특성을 작성하십시오.

   활성 스펙의 **사용자 정의 특성** 패널에서 **새로 만들기**를 클릭한 다음, 다음 세부정보를 입력하십시오.

   이름

   사용자 정의 특성의 이름(이 경우 `subscriptionName`)입니다.

   가치

   사용자 정의 특성의 값입니다. **값** 필드에 JNDI 이름을 사용할 수 있습니다 (예: `WASSharedSubOne`).

   유형

   사용자 정의 특성의 유형입니다. 목록에서 사용자 정의 특성 유형을 선택하십시오. 이 경우 `java.lang.String`입니다.
6. 공유 지속 가능 및 공유 지속 불가능 구독의 경우 sharedSubscription 사용자 정의 특성을 작성하십시오.

   활성 스펙의 **사용자 정의 특성** 패널에서 **새로 만들기**를 클릭한 다음, 다음 세부정보를 입력하십시오.

   이름

   사용자 정의 특성의 이름(이 경우 `sharedSubscription`)입니다.

   가치

   사용자 정의 특성의 값입니다. 활성 스펙에서 공유 구독을 사용하도록 지정하려면 값을 `true`로 설정하십시오. 나중에 이 활성 스펙에 대한 공유 구독 사용을 중지하려면 이 사용자 정의 특성의 값을 `false`로 설정하여 중지할 수 있습니다.

유형

사용자 정의 특성의 유형입니다. 목록에서 사용자 정의 특성 유형을 선택하십시오. 이 경우 `java.lang.String`입니다.

7. 특성이 설정되면 애플리케이션 서버를 다시 시작하십시오.

   메시지가 도착하면 활성 스펙의 MDB가 구동되지만, MDB만 전송된 메시지를 공유합니다.

관련 정보

복제 및 공유된 구독
구독 지속성
인바운드 통신용 자원 어댑터 구성
WebSphere Application Server traditional 9.0에 대한 관련 정보
IBM MQ 메시징 제공자에 대한 토픽 구성
IBM MQ 메시징 제공자 활성화 스펙
IBM MQ 메시징 제공자에 대한 활성화 스펙 작성
IBM MQ 메시징 제공자에 대한 활성화 스펙 구성
IBM MQ 메시징 제공자 JMS 자원에 대한 사용자 정의 특성 구성

JMS 2.0 ConnectionFactory 및 Destination Lookup 특성 사용

WebSphere Application Server traditional 9.0에서 활성화 스펙의 ConnectionFactoryLookup 및 DestinationLookup 특성이 환경 설정에서 사용되는 관리 개체의 JNDI 이름과 함께 제공됩니다.

이 태스크 정보

JMS 2.0 스펙은 MDB를 구동하는 데 사용되는 활성화 스펙에 두 개의 추가 특성을 지정합니다. 이전에는 메시징 시스템에 연결하고 메시지를 가져올 목적지를 정의하는 데 필요한 세부사항을 정의하기 위해 각 벤더가 활성화 스펙에 대한 사용자 정의 특성을 지정해야 했습니다.

새로운 표준 connectionFactoryLookup 및 destinationLookup 특성을 사용하여 관련 오브젝트의 JNDI 이름을 제공할 수 있습니다. WebSphere Application Server traditional 9.0 내에서 IBM MQ 9.0 자원 어댑터가 사전 설치되어 있습니다.

다음 단계에서는 WebSphere Application Server traditional 관리 콘솔을 사용하여 이 두가지 특성을 사용자 정의 및 사용하는 방법을 보여줍니다.

프로시저

먼저 JNDI에서 오브젝트를 작성하십시오.

1. JNDI에서 ConnectionFactory를 정상적으로 작성하십시오(523 페이지의 『관리 콘솔을 사용한 JMS 자원 구성』 참조).
2. JNDI에서 정상적으로 목적지를 작성하십시오(523 페이지의 『관리 콘솔을 사용한 JMS 자원 구성』 참조).

   목적지 오브젝트에 올바른 값이 있어야 합니다.
3. 필요한 값을 사용하여 활성화 스펙을 작성하십시오(523 페이지의 『관리 콘솔을 사용한 JMS 자원 구성』 참조).

   필요한 특성을 사용하여 활성화 스펙을 작성할 수 있습니다. 그러나 다음을 고려해야 합니다.

   - IBM MQ 자원 어댑터가 Java EE 연결 팩토리 및 목적지 조회 속성을 사용하도록 할 경우, 활성화 스펙 작성 시 사용되는 특성과 상관이 적습니다(ActivationSpec ConnectionFactoryLookup and DestinationLookup properties 참조).
   - 그러나 연결 팩토리 또는 목적지에 아직 정의되지 않은 특성을 활성화 스펙에도 지정해야 합니다. 따라서 연결 이용자 특성 및 추가 특성, 그리고 연결을 실제로 작성할 때 사용되는 인증 정보를 정의해야 합니다.
   - 연결 팩토리에 정의된 특성중 ClientID 특성은 특수 처리합니다. 공통 시나리오에서 여러 활성화 스펙과 함께 단일 연결 팩토리를 사용하기 때문입니다. 이 경우 관리는 단순하지만 JMS 스펙에서 고유한 클라이언트 ID를 요구합니다. 활성화 스펙에서 연결 팩토리에 설정된 값을 대체할 수 있는 기능이 필요하기 하기 때문입니다. 활성화 스펙에 클라이언트 ID가 설정되지 않은 경우 연결 팩토리의 값이 사용됩니다.

WebSphere Application Server 관리 콘솔(532 페이지의 『4』 단계에서 설명) 또는 어노테이션(532 페이지의 『5』 단계에서 설명)을 사용하여 2개의 새로운 사용자 정의 특성으로 작성한 활성화 스펙을 업데이트하십시오.

4. WebSphere Application Server 관리 콘솔에서 활성화 스펙을 업데이트하십시오.

   활성화 스펙의 사용자 정의 특성 패널에서 이 두 개의 특성을 설정해야 합니다. 이 특성은 기본 활성화 스펙 패널이나 활성화 스펙 작성 마법사에는 없습니다.

   a) **활성화 스펙 콜렉션** 양식에 표시된 목록에서 활성화 스펙을 선택하십시오.

      활성화 스펙의 세부 정보가 **IBM MQ 메시징 제공자 활성자 스펙 설정** 양식에 표시됩니다.

   b) **IBM MQ 메시징 제공자 활성화 스펙 설정** 양식에서 **사용자 정의 속성**을 클릭하십시오.

      사용자 정의 특성 양식이 표시됩니다.

   c) **사용자 정의 특성** 양식에서 사용자 정의 특성 2개를 새로 작성하십시오(둘 다 java.lang.String 유형).

      새로 만들기를 클릭하고 사용자 정의 특성마다 다음 세부사항을 입력하십시오.

      이름

      사용자 정의 특성의 이름입니다(`connectionFactoryLookup` 또는 `destinationLookup`).

      가치

      사용자 정의 특성의 값입니다. **값** 필드에 JNDI 이름을 사용할 수 있습니다 (예: `QuoteCF` 및 `QuoteQ`).

      유형

      사용자 정의 특성의 유형입니다. 목록에서 사용자 정의 특성 유형을 선택하십시오. 이 경우 `java.lang.String`입니다.

   배치된 MDB가 이 값을 사용하여 연결 팩토리 및 목적지를 작성합니다. MDB 배치시 JNDI 값 구성을 설정할 필요가 없습니다.

5. 활성화 스펙 대신 어노테이션을 사용하십시오.

   MDB 코드의 어노테이션을 사용하여 값을 지정할 수 있습니다. 예를 들어 JNDI 이름 `QuoteCF` 및 `QuoteQ`를 사용하면 코드가 다음과 같습니다.

```
@MessageDriven(activationConfig = {
        @ActivationConfigProperty(propertyName =  "destinationType" , propertyValue =
"javax.jms.Topic" ),
        @ActivationConfigProperty(propertyName =  "destinationLookup" , propertyValue =
"QuoteQ" ),
        @ActivationConfigProperty(propertyName =  "connectionFactoryLookup" , propertyValue
=  "QuoteCF" )}, mappedName =  "LookupMDB" )
        @TransactionAttribute(TransactionAttributeType.REQUIRED)
        @TransactionManagement(TransactionManagementType.CONTAINER)
        publicclass LookupMDB implements MessageListener {
```

관련 정보

인바운드 통신용 자원 어댑터 구성

WebSphere Application Server traditional 9.0에 대한 관련 정보

IBM MQ 메시징 제공자에 대한 통합 연결 팩토리 구성

IBM MQ 메시징 제공자에 대한 토픽 구성

IBM MQ 메시징 제공자 활성화 스펙

IBM MQ 메시징 제공자에 대한 활성화 스펙 작성

IBM MQ 메시징 제공자에 대한 활성화 스펙 구성

IBM MQ 메시징 제공자 JMS 자원에 대한 사용자 정의 특성 구성

최신 자원 어댑터 유지보수 레벨을 사용하도록 애플리케이션 서버 구성

WebSphere Application Server 수정팩을 적용할 때 IBM MQ 자원 어댑터가 자동으로 최신 사용 가능한 유지보수 레벨로 업데이트되도록 하려면 각 노드의 설치에 적용한 WebSphere Application Server 수정팩에 포함된 자원 어댑터의 최신 버전을 사용하도록 사용자 환경의 모든 서버를 구성할 수 있습니다.

시작하기 전에

중요사항: 플랫폼에서 WebSphere Application Server 7.0, 8또는 8.5 를 사용하는 경우, IBM MQ 8.0 자원 어댑터를 애플리케이션 서버에 설치하지 마십시오. IBM MQ 8.0 자원 어댑터는 JMS 2.0을 지원하는 애플리케이션 서버에만 배치될 수 있습니다. 그러나 WebSphere Application Server 7.0, 8및 8.5 는 JMS 1.1만 지원합니다. WebSphere Application Server의 이들 버전은 IBM WebSphere MQ 7.0 자원 어댑터와 함께 제공하는데, 이 자원 어댑터는 BINDINGS 또는 CLIENT 전송 중 하나로 IBM MQ 8.0 큐 관리자에 연결하는 데 사용할 수 있습니다.

이 태스크 정보

다음 상황 중 하나가 사용자 구성에 적용되고 사용자 환경의 모든 서버를 IBM MQ 자원 어댑터의 최신 버전을 사용하도록 구성하려는 경우,

- WebSphere Application Server 7.0.0 Fix Pack 1 이상이 적용된 후 사용자 환경에 있는 애플리케이션 서버의 JVM 로그가 다음 IBM MQ 자원 어댑터 버전 정보를 표시합니다.

  ```
  WMSG1703I:RAR implementation Version 7.0.0.0-k700-L080820
  ```

- 사용자 환경에 있는 임의의 애플리케이션 서버의 JVM 로그가 다음 입력 항목을 포함합니다.

  ```
  WMSG1625E: It was not possible to detect
  지정된 경로에서 IBM MQ 메시징 제공자 코드
  ```

- 하나 이상의 노드가 지금은 현재 WebSphere Application Server 유지보수 레벨에 포함된 자원 어댑터의 최신 버전에 의해 대체되는 IBM MQ 유지보수 레벨의 특정 유지보수 레벨을 사용하도록 이전에 수동으로 업데이트 되었습니다.

예가 참조하는 *profile_root* 디렉토리는 WebSphere Application Server 프로파일에 대한 홈 디렉토리입니다(예: `C:\Program Files\IBM\WebSphere\AppServer1`).

사용자 환경의 모든 셀 및 단일 서버 설치에 대해 다음 단계를 수행했을 때, 새 WebSphere Application Server 수정팩이 적용될 때 서버가 자동으로 IBM MQ 자원 어댑터에 대한 유지보수를 수신합니다.

프로시저

1. 애플리케이션 서버를 시작하십시오. 프로파일이 Network Deployment 구성의 일부인 경우 배치 관리자 및 모든 노드 에이전트를 시작하십시오. 프로파일에 관리 에이전트가 있는 경우 관리 에이전트를 시작하십시오.
2. IBM MQ 자원 어댑터의 유지보수 레벨을 확인하십시오.
   a) 명령 프롬프트 창을 열고 *profile_root*`\bin` 디렉토리로 변경하십시오.

      예를 들어, `cd C:\Program Files\IBM\WebSphere\AppServer1\bin`을 입력하십시오.

   b) `wsadmin.bat -lang jython`을 입력하여 wsadmin 도구를 시작한 후, 프롬프트될 때 사용자 이름과 비밀번호를 입력하십시오.

   c) 다음 명령을 입력한 후 Return을 두 번 누르십시오.

```
wmqInfoMBeansUnsplit = AdminControl.queryNames("WebSphere:type=WMQInfo,*")
wmqInfoMBeansSplit = AdminUtilities.convertToList(wmqInfoMBeansUnsplit)
for wmqInfoMBean in wmqInfoMBeansSplit: print wmqInfoMBean; print AdminControl.invoke(wmqInfoMBean,
'getInfo', '')
```

      Jacl에서도 이 명령을 실행할 수 있습니다. 이를 수행하는 방법에 관한 추가 정보는 WebSphere Application Server 제품 문서에 있는 서버가 사용 가능한 최신 *IBM MQ* 자원 어댑터 유지보수 레벨을 사용하는지 확인을 참조하십시오.

   d) 명령의 표시되는 출력에서 `WMSG1703I` 메시지를 찾고 자원 어댑터 레벨을 확인하십시오.

      예를 들어, WebSphere Application Server 7.0.1 Fix Pack 5의 경우 메시지는 다음과 같습니다.

      ```
      WMSG1703I: RAR implementation Version 7.0.1.3-k701-103-100812
      ```

      이 메시지는 이 수정팩에 올바른 자원 어댑터 레벨인 7.0.1.3-k701-103-100812 버전임을 보여줍니다. 그러나, 다음 메시지가 대신 표시되는 경우 이것은 수정팩 15에 대한 올바른 유지보수 레벨로 자원 어댑터를 조정해야 함을 의미합니다.

      ```
      WMSG1703I: RAR implementation Version 7.0.0.0-k700-L080820
      ```

3. 다음 Jython 스크립트를 `convertWMQRA.py`라는 파일에 복사한 후, 프로파일 루트 디렉토리(예: `C:\Program Files\IBM\WebSphere\AppServer1\bin`)에 저장하십시오.

```
ras = AdminUtilities.convertToList(AdminConfig.list('J2CResourceAdapter'))

for ra in ras :
  desc = AdminConfig.showAttribute(ra, "description")
  if (desc == "WAS 7.0 Built In IBM MQ Resource Adapter") or (desc == "WAS 7.0.0.1 Built In IBM MQ 
Resource Adapter"):
    print "Updating archivePath and classpath of " + ra
    AdminConfig.modify(ra, [['archivePath', "${WAS_INSTALL_ROOT}/installedConnectors/wmq.jmsra.rar"]])
    AdminConfig.unsetAttributes(ra, ['classpath'])
    AdminConfig.modify(ra, [['classpath', "${WAS_INSTALL_ROOT}/installedConnectors/wmq.jmsra.rar"]])
    AdminConfig.save()
  #end if
#end for
```

팁: 파일을 저장할 때 파일이 텍스트 파일이 아닌 Python 파일로 저장되는지 확인하십시오.

4. WebSphere Application Server wsadmin 도구를 사용하여 방금 작성한 Jython 스크립트를 실행하십시오.

   명령 프롬프트를 열고 WebSphere Application Server에 대한 홈 디렉토리의 `\bin` 디렉토리(예: `C:\Program Files\IBM\WebSphere\AppServer1\bin` 디렉토리)로 이동한 후 다음 명령을 입력하고 Return을 누르십시오.

   ```
   wsadmin -lang jython -f convertWMQRA.py
   ```

   프롬프트될 때 사용자 이름과 비밀번호를 입력하십시오.

   참고: Network Deployment 구성의 일부인 프로파일에 대해 스크립트를 실행하는 경우, 스크립트는 해당 구성에서 업데이트가 필요한 모든 프로파일을 업데이트합니다. 기존 구성 파일이 불일치하는 경우 전체 재동기화가 필요할 수 있습니다.

5. Network Deployment 구성에서 실행 중인 경우, 노드 에이전트가 완전히 다시 동기화되었는지 확인하십시오. 자세한 정보는 wsadmin 스크립트 도구를 사용한 노드 동기화 또는 노드 추가, 관리 및 제거를 참조하십시오.
6. 프로파일의 모든 서버를 중지하십시오. 프로파일이 Network Deployment 구성의 일부인 경우 구성의 모든 클러스터 멤버를 중지시키고 구성의 모든 노드 에이전트를 중지시킨 후 배치 관리자를 중지시키십시오. 프로파일에 관리 에이전트가 있는 경우 관리 에이전트를 중지하십시오.
7. *profile_root*/bin 디렉토리에서 **osgiCfgInit** 명령을 실행하십시오.

   `osgiCfgInit` 명령은 OSGi 런타임 환경에서 사용하는 클래스 캐시를 재설정합니다. 프로파일이 Network Deployment 구성의 일부인 경우, 구성의 일부인 모든 프로파일의 `profile_root/bin` 디렉토리에서 **osgiCfgInit** 명령을 실행하십시오.
8. 프로파일의 모든 서버를 다시 시작하십시오. 프로파일이 Network Deployment 구성의 일부인 경우 구성의 모든 클러스터 멤버를 재시작하고 구성의 모든 노드 에이전트를 재시작한 후 배치 관리자를 재시작하십시오. 프로파일에 관리 에이전트가 있는 경우 관리 에이전트를 재시작하십시오.
9. 2단계를 반복하여 자원 어댑터가 지금 올바른 레벨에 있는지 확인하십시오.

다음에 수행할 작업

이 주제에 설명된 단계를 수행한 후 문제점이 계속 발생하고 이전에 WebSphere Application Server 관리 콘솔의 JMS 제공자 설정 패널에서 **자원 어댑터 갱신** 단추를 사용하여 사용자 환경의 노드에서 IBM MQ 자원 어댑터를 갱신한 경우, APAR PM10308에 설명된 문제가 발생할 수 있습니다.

관련 정보

IBM MQ 자원 어댑터 사용

WebSphere Application Server 8.5.5에 대한 관련 정보

사용 가능한 최신 IBM MQ 자원 어댑터 유지보수 레벨을 사용하는지 확인

wsadmin 스크립트 도구를 사용하여 노드 동기화

노드 추가, 관리 및 제거

JMS 제공자 설정

WebSphere Application Server 8.0에 대한 관련 정보

사용 가능한 최신 IBM MQ 자원 어댑터 유지보수 레벨을 사용하는지 확인

wsadmin 스크립트 도구를 사용하여 노드 동기화

노드 추가, 관리 및 제거
JMS 제공자 설정
WebSphere Application Server 7.0에 대한 관련 정보
사용 가능한 최신 IBM MQ 자원 어댑터 유지보수 레벨을 사용하는지 확인
wsadmin 스크립트 도구를 사용하여 노드 동기화
노드 추가, 관리 및 제거
JMS 제공자 설정

JMS PROVIDERVERSION 특성 구성

IBM MQ 메시징 제공자에는 3개의 조작 모드 즉, 정상 모드, 제한적 정상 모드, 마이그레이션 모드가 있습니다. JMS 애플리케이션이 발행 및 구독하기 위해 사용하는 모드를 선택하기 위해 JMS **PROVIDERVERSION** 특성을 설정할 수 있습니다.

이 태스크 정보

IBM MQ 메시징 제공자 작동 모드의 선택은 주로 PROVIDERVERSION 연결 팩토리 특성을 설정하여 제어될 수 있습니다. 작동 모드는 또한 모드가 지정되지 않은 경우 자동으로 선택될 수도 있습니다.

PROVIDERVERSION 특성은 세 가지 IBM MQ 메시징 제공자 작동 모드 사이를 구별합니다.

IBM MQ 메시징 제공자 정상 모드
: 정상 모드는 IBM MQ 큐 관리자의 모든 기능을 사용하여 JMS를 구현합니다. 이 모드는 JMS 2.0 API 및 기능을 사용하도록 최적화되어 있습니다.

제한이 있는 IBM MQ 메시징 제공자 정상 모드
: 제한이 있는 정상 모드는 JMS 2.0 API를 사용하지만, 새로운 기능인 공유 구독, 지연 전달, 비동기 송신은 사용하지 않습니다.

IBM MQ 메시징 제공자 마이그레이션 모드
: 마이그레이션 모드를 사용하면 IBM MQ 8.0 이상 큐 관리자에 연결할 수 있지만, 미리 읽기 및 스트리밍 같은 IBM WebSphere MQ 7.0 이상 큐 관리자의 어떤 기능도 사용되지 않습니다.

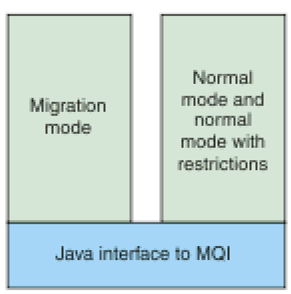


그림 *91.* 메시징 제공자 모드

프로시저

특정 연결 팩토리에 대한 **PROVIDERVERSION** 특성을 구성하려면 다음을 수행하십시오.

- IBM MQ Explorer를 사용하여 **PROVIDERVERSION** 특성을 구성하려면 큐 관리자 및 오브젝트 구성을 참조하십시오.
- JMS 관리 도구를 사용하여 **PROVIDERVERSION** 특성을 구성하려면 큐 관리자 및 오브젝트 구성을 참조하십시오.
- IBM JMS 확장 또는 IBM MQ JMS 확장을 사용하는 JMS 애플리케이션에서 **PROVIDERVERSION** 특성을 구성하려면 IBM MQ classes for JMS 애플리케이션에서 연결 팩토리 및 목적지 작성 및 구성을 참조하십시오.

JVM의 모든 연결 팩토리에 대한 연결 팩토리 제공자 모드 설정을 대체하려면, 다음을 수행하십시오.

- 연결 팩토리 제공자 모드 설정을 대체하려면 `com.ibm.msg.client.wmq.overrideProviderVersion` 특성을 사용하십시오.

  사용 중인 연결 팩토리를 변경할 수 없는 경우 `com.ibm.msg.client.wmq.overrideProviderVersion` 특성을 사용하여 연결 팩토리의 설정을 대체할 수 있습니다. 이 대체는 JVM의 모든 연결 팩토리에 적용되지만 실제 연결 팩토리 오브젝트는 수정되지 않습니다.

관련 정보

PROVIDERVERSION

JMS 제공자 버전 문제점 해결

연결 팩토리 특성

IBM MQ classes for JMS 오브젝트 특성의 종속성

IBM MQ 메시징 제공자 작동 모드

연결 팩토리에 대한 PROVIDERVERSION 특성을 적절한 값으로 설정하여 JMS 애플리케이션이 발행 및 구독하기 위해 사용하는 작동 모드 중 IBM MQ 메시징 제공자 모드를 선택할 수 있습니다. 일부 경우에는 PROVIDERVERSION 특성이 지정되지 않음으로 설정되는데, 이 경우 JMS 클라이언트는 사용할 작동 모드를 판별하는 알고리즘을 사용합니다.

PROVIDERVERSION 특성 값

연결 팩토리 **PROVIDERVERSION** 특성을 다음 값 중 하나로 설정할 수 있습니다.

8 - 정상 모드
: JMS 애플리케이션이 정상 모드를 사용합니다. 이 모드는 IBM MQ 큐 관리자의 모든 기능을 사용하여 JMS를 구현합니다.

7 - 제한이 있는 정상 모드
: JMS 애플리케이션은 제한이 있는 정상 모드를 사용합니다. 이 모드는 JMS 2.0 API를 사용하지만 공유 구독, 지연 전달 또는 비동기 송신과 같은 새 기능은 사용하지 않습니다.

6 - 마이그레이션 모드
: JMS 애플리케이션은 마이그레이션 모드를 사용합니다. 마이그레이션 모드에서, IBM MQ classes for JMS는 IBM WebSphere MQ 6.0에서 제공되는 것과 비슷한 기능 및 알고리즘을 사용합니다.

unspecified(기본값)
: JMS 클라이언트는 사용되는 작동 모드를 판별하는 알고리즘을 사용합니다.

PROVIDERVERSION 특성에 대해 지정한 값은 문자열이어야 합니다. 8, 7 또는 6 옵션을 지정하는 경우, 다음 형식으로 이를 수행할 수 있습니다.

- V.R.M.F
- V.R.M
- V.R
- V

V, R, M 및 F는 0 이상의 정수 값입니다. 추가 R, M 및 F 값은 선택적이며, 미세한 제어가 필요한 경우에 사용할 수 있습니다. 예를 들어 **PROVIDERVERSION** 레벨 7을 사용하려는 경우 **PROVIDERVERSION**=7, `7.0`, `7.0.0` 또는 `7.0.0.0`을 설정할 수 있습니다.

연결 팩토리 오브젝트의 유형

연결 팩토리 오브젝트의 다음 유형에 대해 **PROVIDERVERSION** 특성을 설정할 수 있습니다.

- MQConnectionFactory
- MQQueueConnectionFactory
- MQTopicConnectionFactory
- MQXAConnectionFactory

- MQXAQueueConnectionFactory
- MQXAQueueConnectionFactory
- MQXAQueueConnectionFactory
- MQXATopicConnectionFactory

이런 여러 가지 유형의 연결 팩토리에 대한 자세한 정보는 514 페이지의 『관리 도구를 사용하여 JMS 오브젝트 구성』의 내용을 참조하십시오.

관련 정보
JMS용 IBM MQ 클래스 아키텍처

PROVIDERVERSION 정상 모드

정상 모드는 IBM MQ 큐 관리자의 모든 기능을 사용하여 JMS를 구현합니다. 이 모드는 JMS 2.0 API 및 기능을 사용하도록 최적화되어 있습니다.

다음 플로우차트는 JMS 클라이언트가 정상 모드 연결이 작성될 수 있는지 여부를 판별하기 위해 수행하는 검사를 표시합니다.

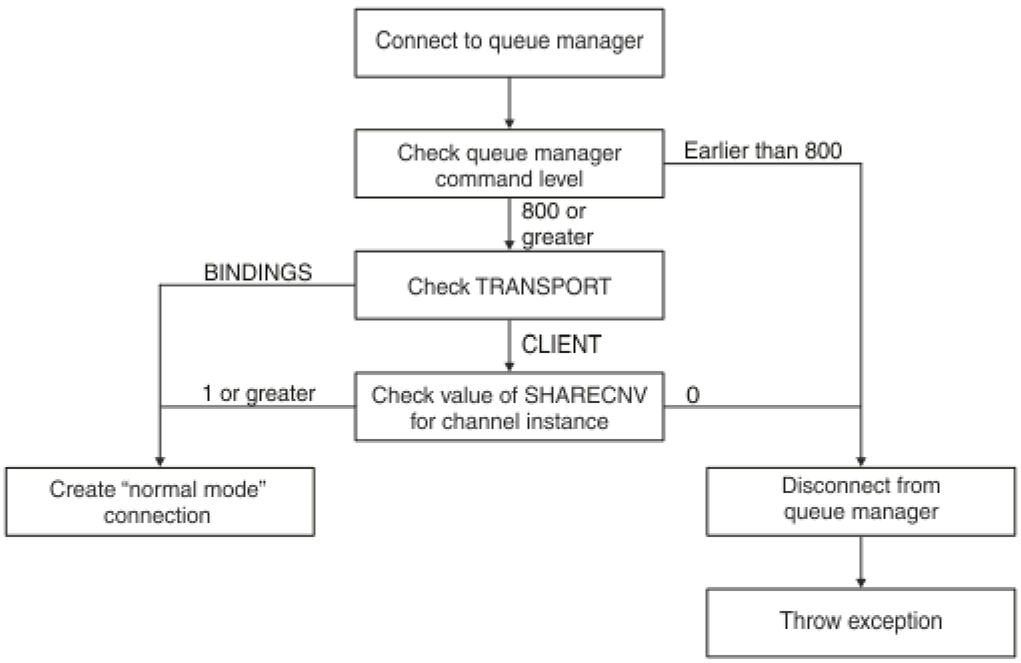


그림 92. *PROVIDERVERSION* 정상 모드

연결 팩토리 설정에서 지정된 큐 관리자가 800 이상의 명령 레벨을 갖고 연결 팩토리의 **TRANSPORT** 특성이 BINDINGS으로 설정되는 경우, 정상 모드 연결은 추가 특성을 검사하지 않고 작성됩니다.

연결 팩토리 설정에서 지정된 큐 관리자가 800 이상의 명령 레벨을 갖고 **TRANSPORT** 특성이 CLIENT로 설정되는 경우, 서버 연결 채널의 **SHARECNV** 특성도 검사됩니다. 이 검사는 IBM MQ 메시징 제공자 정상 모드가 공유 대화 기능을 사용하기 때문에 필요합니다. 그러므로 정상 모드 연결 시도가 성공하기 위해서는 공유될 수 있는 대화 수를 제어하는 **SHARECNV** 특성이 1 이상의 값을 가져야 합니다.

플로우차트에 표시된 모든 검사가 성공하는 경우, 큐 관리자에 대한 정상 모드 연결이 작성되고 모든 JMS 2.0 API 및 기능(즉, 비동기 송신, 지연 전달 및 공유 구독)도 사용할 수 있습니다.

다음 이유 중 하나 때문에 정상 모드 연결을 작성하려는 시도가 실패합니다.

- 연결 팩토리 설정에서 지정되는 큐 관리자가 800보다 낮은 명령 레벨을 갖습니다. 이 경우 `createConnection` 메소드는 예외 `JMSFMQ0003`로 실패합니다.
- 서버 연결 채널의 **SHARECNV** 특성은 `0`(으)로 설정됩니다. 이 특성에 1이상의 값이 없으면 `createConnection` 메소드가 `JMSCC5007` 예외와 함께 실패합니다.

관련 정보
IBM MQ classes for JMS 오브젝트 특성의 종속성
DEFINE CHANNEL(SHARECNV 특성)
TRANSPORT

제한이 있는 *PROVIDERVERSION* 정상 모드

제한이 있는 정상 모드는 JMS 2.0 API를 사용하지만, 공유 구독, 지연 전달 또는 비동기 송신 같은 새 IBM MQ 8.0 이상 기능을 사용하지 않습니다.

다음 플로우차트는 JMS 클라이언트가 제한이 있는 정상 모드 연결이 작성될 수 있는지 여부를 판별하기 위해 수행하는 검사를 표시합니다.

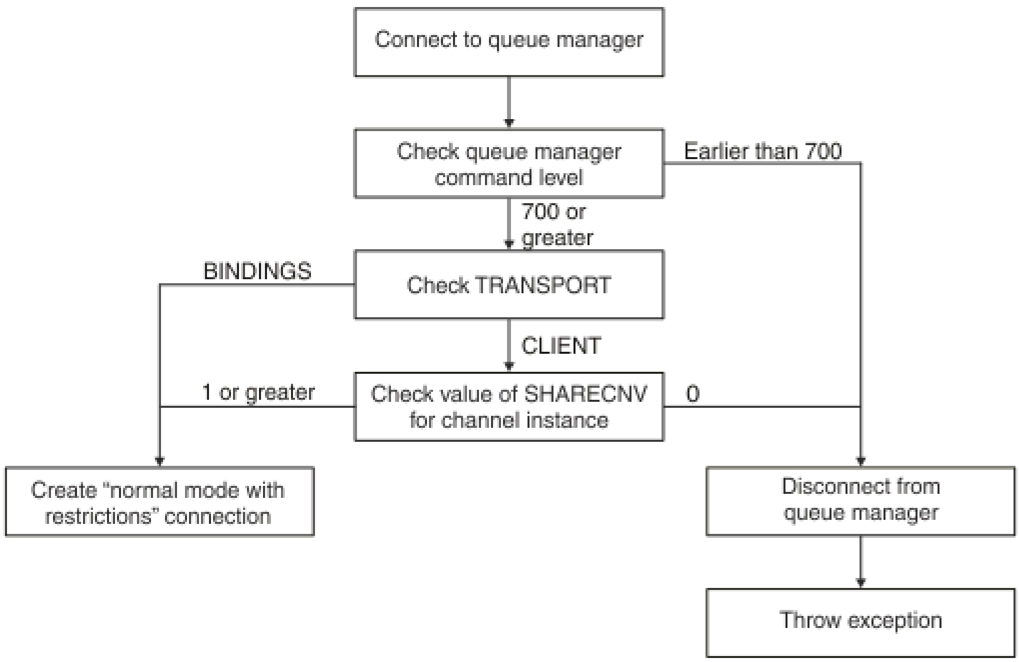


그림 *93.* 제한이 있는 *PROVIDERVERSION* 정상 모드

연결 팩토리 설정에서 지정된 큐 관리자가 700 이상의 명령 레벨을 갖고 연결 팩토리의 **TRANSPORT** 특성이 BINDINGS으로 설정되는 경우, 정상 모드 연결은 추가 특성을 검사하지 않고 작성됩니다.

연결 팩토리 설정에서 지정된 큐 관리자가 700 이상의 명령 레벨을 갖고 **TRANSPORT** 특성이 CLIENT로 설정되는 경우, 서버 연결 채널의 **SHARECNV** 특성도 검사됩니다. 이 검사는 IBM MQ 메시징 제공자 제한이 있는 정상 모드가 공유 대화 기능을 사용하기 때문에 필요합니다. 그러므로 제한이 있는 정상 모드 연결 시도가 성공하기 위해서는 공유될 수 있는 대화 수를 제어하는 **SHARECNV** 특성이 1 이상의 값을 가져야 합니다.

플로우차트에 표시된 모든 검사가 성공하는 경우, 큐 관리자에 대한 제한이 있는 정상 모드 연결이 작성되고 사용자는 JMS 2.0 API를 사용할 수 있지만 비동기 송신, 지연 전달 또는 공유 구독 기능은 사용할 수 없습니다.

다음 이유 중 하나 때문에 제한이 있는 정상 모드 연결을 작성하려는 시도가 실패합니다.

- 연결 팩토리 설정에서 지정되는 큐 관리자가 700보다 낮은 명령 레벨을 갖습니다. 이 경우에 `createConnection` 메소드가 JMSFCC5008 예외와 함께 실패합니다.
- 서버 연결 채널의 **SHARECNV** 특성은 `0`(으)로 설정됩니다. 이 특성에 1이상의 값이 없으면 `createConnection` 메소드가 JMSCC5007 예외와 함께 실패합니다.

관련 정보
IBM MQ classes for JMS 오브젝트 특성의 종속성
DEFINE CHANNEL(SHARECNV 특성)
TRANSPORT

PROVIDERVERSION 마이그레이션 모드

마이그레이션 모드의 경우 IBM MQ classes for JMS는 큐 처리된 발행/구독, 클라이언트 측에서 구현되는 선택, 비멀티플렉스 채널, 리스너를 구현하는 데 사용되는 폴링 같이 IBM WebSphere MQ 6.0에서 제공되는 것과 비슷한 기능 및 알고리즘을 사용합니다.

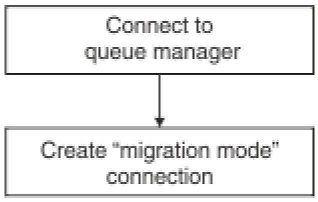


그림 *94. PROVIDERVERSION* 마이그레이션 모드

IBM MQ Enterprise Transport 6.0을 사용하여 WebSphere Message Broker 6.0 또는 6.1 에 연결하려면 마이그레이션 모드를 사용해야 합니다.

마이그레이션 모드를 사용하여 IBM MQ 8.0 큐 관리자에 연결할 수 있지만, IBM MQ classes for JMS 큐 관리자의 새로운 기능(예: 미리 읽기 또는 스트리밍)은 사용되지 않습니다. 분산 플랫폼의 IBM MQ 8.0 또는 이후 큐 관리자에 연결하는 IBM MQ 8.0 이상의 클라이언트가 있는 경우, z/OS 또는 z/OS의 IBM MQ 8.0 이후에 큐 관리자를 연결하면 , 메시지 선택은 클라이언트 시스템이 아닌 큐 관리자에 의해 수행됩니다.

IBM MQ 메시징 제공자 마이그레이션 모드가 지정되고 IBM MQ classes for JMS가 임의의 JMS 2.0 API를 사용하려고 시도하는 경우, API 메소드 호출은 `JMSCC5007` 예외와 함께 실패합니다.

관련 정보

IBM MQ classes for JMS 오브젝트 특성의 종속성
TRANSPORT

지정되지 않은 *PROVIDERVERSION*

연결 팩토리의 **PROVIDERVERSION** 특성이 지정되지 않을 때, JMS 클라이언트는 알고리즘을 사용하여 큐 관리자에 연결하는 데 사용되는 작동 모드를 판별합니다. IBM MQ classes for JMS의 이전 버전과 함께 JNDI 네임스페이스에서 작성된 연결 팩토리는 연결 팩토리가 IBM MQ classes for JMS의 새 버전과 함께 사용될 때 지정되지 않은 값을 취합니다.

PROVIDERVERSION 특성이 지정되지 않은 경우 `createConnection` 메소드가 호출될 때 알고리즘이 사용됩니다. 알고리즘은 IBM MQ 메시징 제공자 정상 모드, 제한이 있는 정상 모드 또는 IBM MQ 메시징 제공자 마이그레이션 모드가 필요한지 여부를 판별하기 위해 연결 팩토리 특성의 수를 확인합니다. 정상 모드가 항상 첫 번째로 시도된 후 제한이 있는 정상 모드가 시도됩니다. 이들 작동 모드의 어느 것도 작성할 수 없는 경우, JMS 클라이언트는 큐 관리자에서 연결을 끊은 후 마이그레이션 모드 연결을 시도하기 위해 큐 관리자와 다시 연결합니다.

BROKERVER, BROKERQMGR, PSMODE 및 BROKERCONQ 특성의 확인

특성 값의 검사는 그림 1에 표시된 것처럼 **BROKERVER** 특성으로 시작합니다.

BROKERVER 특성이 V1으로 설정되면 그림 2에 표시된 대로 **TRANSPORT** 특성이 선택됩니다. 그러나 **BROKERVER** 특성이 V2로 설정된 경우 그림 1에 표시된 추가 검사는 **TRANSPORT** 특성을 확인하기 전에 수행됩니다.

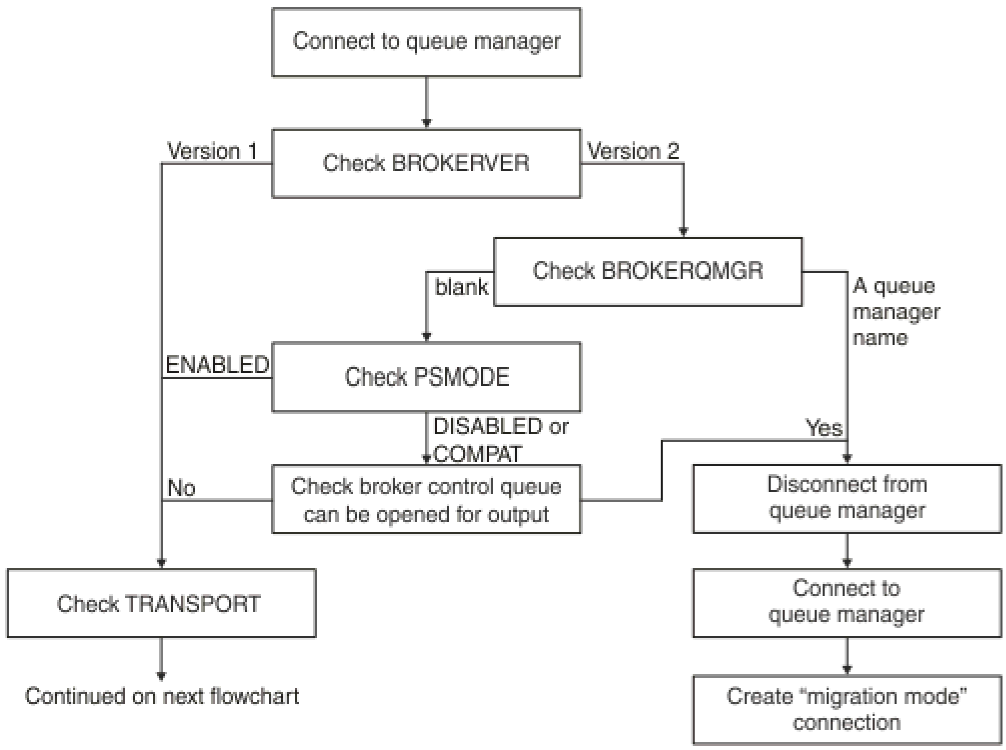


그림 *95.* 지정되지 않은 *PROVIDERVERSION*

BROKERVER 특성이 V2로 설정되는 경우 정상 모드 연결이 가능하기 위해서는 **BROKERQMGR** 특성이 공백이어야 합니다. 추가로 큐 관리자의 **PSMODE** 속성이 ENABLED로 설정되어야 하거나 **BROKERCONQ** 특성에 의해 지정되는 브로커 제어 큐가 출력을 위해 열릴 수 있지 않아야 합니다.

특성 값이 일반 모드 연결에 필요한 대로 설정된 경우 그림 2에 표시된 대로 다음 이동을 **TRANSPORT** 특성으로 확인하십시오.

특성 값이 정상 모드 연결을 위해 필요한 대로 설정되지 않은 경우, JMS 클라이언트가 큐 관리자에서 연결을 끊은 후 다시 연결하고 마이그레이션 모드 연결을 작성합니다. 이것은 다음 경우에 발생합니다.

- **BROKERQMGR** 특성이 공백이고 큐 관리자에 대한 **PSMODE** 속성이 COMPAT 또는 DISABLED로 설정되고 **BROKERCONQ** 특성에 의해 지정되는 브로커 제어 큐가 출력을 위해 열릴 수 있는 경우(즉, 출력에 대한 MQOPEN이 성공함).
- **BROKERQMGR** 특성이 큐 이름을 지정하는 경우.

TRANSPORT 특성 및 명령 레벨의 검사

그림 2는 큐 관리자의 **TRANSPORT** 특성 및 명령 레벨에 대해 작성되는 검사를 보여줍니다.

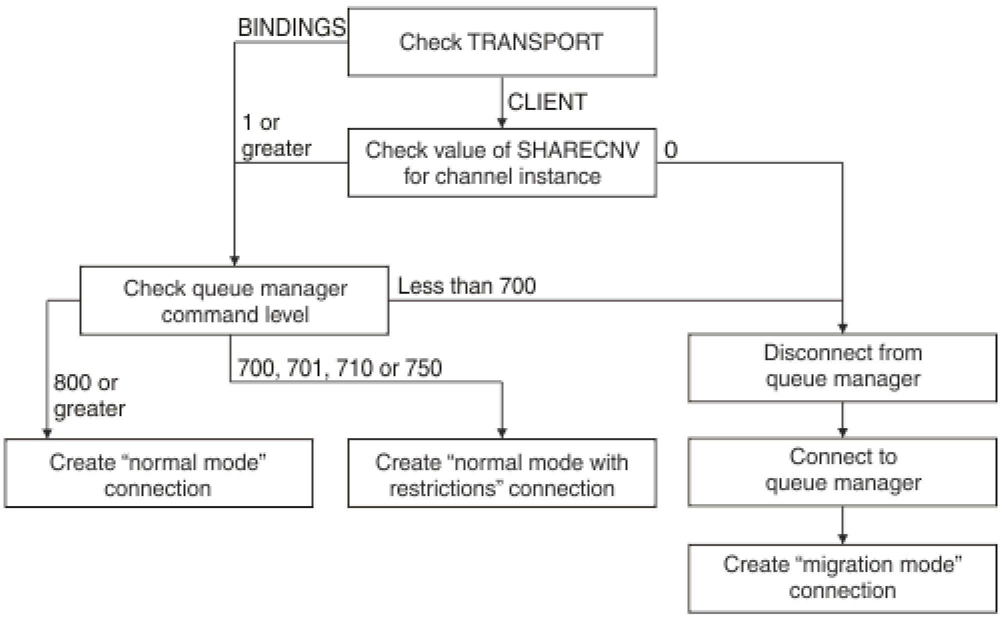


그림 *96.* 지정되지 않은 *PROVIDERVERSION(*계속*)*

다음 경우에 정상 모드 연결이 작성됩니다.

- 연결 팩토리의 **TRANSPORT** 특성이 BINDINGS로 설정되고, 큐 관리자가 800 이상의 명령 레벨을 갖습니다.
- **TRANSPORT** 특성이 CLIENT로 설정되고, 서버 연결 채널의 **SHARECNV** 특성이 1 이상의 값을 갖고, 큐 관리자가 800 이상의 명령 레벨을 갖습니다.

큐 관리자가 700, 701, 710 또는 750의 명령 레벨을 갖는 경우, 큐 관리자에 대한 제한이 있는 정상 모드 연결이 작성됩니다.

TRANSPORT 특성이 CLIENT로 설정되고 서버 연결 채널의 **SHARECNV** 특성이 0의 값을 갖는 경우에도 마이그레이션 모드 연결이 작성됩니다.

관련 정보

IBM MQ classes for JMS 오브젝트 특성의 종속성
ALTER QMGR(PSMODE 속성)
BROKERCONQ
BROKERQMGR
BROKERVER
DEFINE CHANNEL(SHARECNV 특성)
TRANSPORT

PROVIDERVERSION 기본 설정을 대체할 시기

IBM MQ classes for JMS의 이전 버전으로 JNDI 네임스페이스에서 작성된 연결 팩토리가 IBM MQ classes for JMS의 새 버전에서 사용되는 경우, 연결 팩토리에 대한 **PROVIDERVERSION** 특성이 unspecified의 기본값으로 설정되며 사용되는 IBM MQ 메시징 제공자 작동 모드를 판별하기 위한 알고리즘이 사용됩니다. 그러나, IBM MQ classes for JMS가 올바르게 작업할 수 있도록 **PROVIDERVERSION** 특성의 기본 선택을 대체해야 하는 두 가지 경우가 있습니다.

참고: 이 주제에서 설명하는 이주 모드는 IBM WebSphere MQ 6.0 에서 7.0으로 이주하기 위한 것입니다. 나중 릴리스로부터의 마이그레이션에는 적용되지 않습니다.

IBM WebSphere MQ 6.0, WebSphere Application Server 6.0.x 및 WebSphere Message Broker 6은 지원되지 않으므로, 이 주제는 참조 목적으로만 포함됩니다.

PROVIDERVERSION 특성이 unspecified의 기본값으로 설정될 때, 539 페이지의 『지정되지 않은 PROVIDERVERSION』에서 설명하는 대로 사용할 작동 모드를 판별하기 위한 알고리즘이 사용됩니다. 그러나, 다음 두 시나리오에서는 이 알고리즘을 사용할 수 없습니다.

1. WebSphere Message Broker 및 WebSphere Event Broker 가 호환성 모드에 있는 경우, WebSphere Message Broker 및 WebSphere Event Broker 에 대한 **PROVIDERVERSION** 특성 값을 지정하여 올바르게 작업해야 합니다.
2. WebSphere Application Server 6.0.1, 6.0.2 또는 6.1을 사용하는 경우 연결 팩토리는 WebSphere Application Server 관리 콘솔을 사용하여 정의됩니다.

   WebSphere Application Server에서 연결 팩토리의 **BROKERVER** 특성의 기본값은 V2입니다. JMS 관리 도구 **JMSAdmin** 또는 IBM MQ Explorer 를 사용하여 작성된 연결 팩토리에 대한 **BROKERVER** 특성의 기본값은 V1입니다. 이 특성은 이제 IBM MQ에서 지정되지 않습니다.

BROKERVER 특성이 V2로 설정된 경우, WebSphere Application Server 에 의해 작성되었거나 연결 팩토리가 이전에 발행/구독에 사용되었고 **BROKERCONQ** 특성이 정의되어 있는 기존 큐 관리자가 있는 경우 (이전에 발행/구독 메시징에 사용되었기 때문에), IBM MQ 메시징 제공자 이주 모드가 사용됩니다.

그러나 응용프로그램이 피어 투 피어 통신을 사용하고 응용프로그램이 발행/구독에 사용된 적이 있는 기존 큐 관리자를 사용 중이고 **BROKERVER** 가 2로 설정된 연결 팩토리를 갖는 경우 (WebSphere Application Server에서 연결 팩토리가 작성된 경우 기본 설정), IBM MQ 메시징 제공자 이주 모드가 사용됩니다. 이 경우에 IBM MQ 메시징 제공자 마이그레이션 모드 사용은 불필요합니다. 대신 IBM MQ 메시징 제공자 정상 모드를 사용하십시오. 다음 방법 중 하나를 사용하여 이를 해결할 수 있습니다.

- **BROKERVER**를 1 또는 unspecified로 설정하십시오. 사용자가 선택하는 옵션은 애플리케이션에 따라 다릅니다.
- **PROVIDERVERSION**을 8 또는 7(WebSphere Application Server 6.1의 사용자 정의 특성)로 설정하십시오.

  그렇지 않으면, 클라이언트 구성 특성을 사용하거나 **BROKERCONQ** 특성 세트를 갖지 않도록 연결된 큐 관리자를 수정하거나, 큐를 사용 불가능하게 만드십시오.

WebSphere Application Server에서 제공자 버전 정보 구성

WebSphere Application Server에서 제공자 버전 정보를 구성하기 위해 관리 콘솔 또는 wsadmin 명령을 사용할 수 있습니다.

프로시저

WebSphere Application Server에서 IBM MQ 연결 팩토리 또는 활성화 스펙 오브젝트에 대한 제공자 버전 정보를 구성하려면, WebSphere Application Server 제품 문서의 추가 정보에 대한 링크는 관련 정보를 참조하십시오.

WebSphere Application Server 8.5.5에 대한 관련 정보
IBM MQ 메시징 제공자 연결 팩토리 설정
createWMQConnectionFactory 명령
IBM MQ 메시징 제공자 활성화 스펙 설정
createWMQActivationSpec 명령
WebSphere Application Server 8.0.0에 대한 관련 정보
IBM MQ 메시징 제공자 연결 팩토리 설정
createWMQConnectionFactory 명령
IBM MQ 활성화 스펙 설정
createWMQActivationSpec 명령
WebSphere Application Server 7.0.0에 대한 관련 정보
IBM MQ 메시징 제공자 연결 팩토리 설정
createWMQConnectionFactory 명령
IBM MQ 활성화 스펙 설정
createWMQActivationSpec 명령

WebSphere Application Server 지속 가능 구독 제거

WebSphere Application Server 7.0 및 8.0와 함께 IBM MQ 메시징 제공자를 사용하는 경우 활성화 스펙에 바인드된 메시지 구동 Bean 애플리케이션에서 작성된 지속 가능한 등록은 제거되지 않습니다. 지속 가능 구독은 IBM MQ Explorer 또는 IBM MQ 명령행 유틸리티를 사용하여 제거될 수 있습니다.

이 태스크에 대한 정보

애플리케이션이 IBM MQ에 연결하는 데 WebSphere MQ 메시징 제공자 정상 모드 를 사용하는 WebSphere Application Server 7.0 또는 8.0 인스턴스 내부에서 실행 중인 경우, 지속 가능한 등록을 제거하는 메시지 구동 Bean 애플리케이션을 리스너 포트 또는 활성화 스펙을 사용하도록 구성할 수 있습니다.

메시지 구동 Bean 애플리케이션이 리스너 포트에 바인딩되면 애플리케이션이 처음 시작될 때 IBM MQ 메시징 제공자가 애플리케이션에 대한 지속 가능 구독을 작성합니다. 지속 가능 구독은 메시지 구동 Bean 애플리케이션을 애플리케이션 서버에서 설치 제거하고 애플리케이션 서버를 재시작하면 제거됩니다.

활성화 스펙에 바인딩된 메시지 구동 Bean 애플리케이션은 약간 다른 방식으로 작동합니다. 지속 가능 구독은 애플리케이션이 처음 시작될 때 애플리케이션에 대해 작성됩니다. 그러나 지속 가능 구독은 애플리케이션이 설치 제거되고 애플리케이션 서버를 재시작할 때 제거되지 않습니다.

이로 인해 WebSphere Application Server 시스템에 더 이상 설치되어 있지 않은 애플리케이션에 대해 IBM MQ 발행/구독 엔진에 다수의 지속 가능 구독이 남아 있을 수 있습니다. 이러한 구독은 "고아 구독"이라고 알려져 있으며 발행/구독 엔진이 실행 중일 때 큐 관리자에 문제를 일으킬 수 있습니다.

메시지가 토픽에 대해 발행되면 IBM MQ 발행/구독 엔진은 해당 토픽에 등록된 각 지속 가능 구독에 대해 해당 메시지의 사본을 작성하고 이를 내부 큐에 넣습니다. 지속 가능 독을 사용하는 애플리케이션은 이 내부 큐에서 메시지를 가져와서 이용합니다.

지속 가능 구독을 사용 중이었던 메시지 구동 Bean 애플리케이션이 더 이상 설치되어 있지 않으면 애플리케이션에 대해 발행된 메시지 사본이 계속해서 작성됩니다. 하지만 이러한 메시지는 처리되지 않으므로 내부 큐에 절대 제거되지 않는 다수의 메시지가 남아 있게 됩니다.

시작하기 전 주의사항

IBM MQ 발행/구독 엔진에 등록된 구독은 연관된 구독 이름을 갖게 됩니다.

활성화 스펙에 바인딩된 메시지 구동 Bean에 대한 WebSphere Application Server IBM MQ 메시징 제공자에서 작성한 지속 가능 구독은 다음 형식의 구독 이름을 갖습니다.

```
JMS:queue manager name:client identifier:subscription name
```

여기서:

queue manager name
: 이는 발행/구독 엔진이 실행되는 IBM MQ 큐 관리자의 이름입니다.

client identifier
: 이는 메시지 구동 Bean이 바인딩된 활성화 스펙의 클라이언트 ID 특성 값입니다.

subscription name
: 이는 메시지 구동 Bean 애플리케이션이 사용하도록 구성된 활성화 스펙에 대한 활성화 스펙 특성 구독 이름의 값입니다.

예를 들어, 큐 관리자 testQM에 연결하도록 설정된 활성화 스펙이 있다고 가정합니다. 활성화 스펙에는 다음과 같은 특성이 설정되어 있습니다.

- Client ID = testClientID
- Subscription name = durableSubscription1

지속 가능 구독을 갖는 메시지 구동 Bean이 이 활성화 스펙에 바인딩되면 다음 구독 이름을 갖는 큐 관리자 testQM에 대해 구독이 IBM MQ 발행/구독 엔진에서 작성됩니다.

- JMS:testQM:testClientID:durableSubscription1

제공된 큐 관리자에 대해 IBM MQ 발행/구독 엔진에 등록된 구독은 다음 방법 중 하나로 볼 수 있습니다.

- 첫 번째 옵션은 IBM MQ Explorer를 사용하는 것입니다. IBM MQ Explorer 가 발행/구독 작업에 사용되는 큐 관리자에 연결된 경우, 탐색 분할창에서 `IBM WebSphere MQ ->`*`queue manager name`*`-> Subscriptions` 항목을 클릭하여 현재 발행/구독 엔진에 등록된 등록자 목록을 볼 수 있습니다.
- 발행/구독 엔진에 등록된 등록을 보는 다른 방법은 IBM MQ 명령행 유틸리티 **runmqsc** 를 사용하고 **display sub**명령을 실행하는 것입니다. 이를 수행하려면 명령 프롬프트를 가져오고 *WebSphere MQ*`\bin` 디렉토리로 변경한 후 다음 명령을 입력하여 **runmqsc**를 시작하십시오.

  – `runmqsc` *`queue manager name`*

  runmqsc 유틸리티가 시작되면 다음 명령을 입력하여 **runmqsc**가 연결한 큐 관리자에서 실행 중인 발행/구독 엔진에 현재 등록된 모든 지속 가능 구독을 나열하십시오.

  – `display sub(*) durable`

발행/구독 엔진에 등록된 지속 가능 구독이 여전히 활성인지 확인하려면 다음을 수행하십시오.

1. 발행/구독 엔진에 등록된 지속 가능 구독 목록을 생성하십시오.
2. 각 지속 가능 구독에 대해 다음을 수행하십시오.
   - 지속 가능 구독자의 구독 이름을 확인하고 *client identifier* 및 *subscription name* 값에 유의하십시오.
   - 이 발행/구독 엔진에 연결 중인 WebSphere Application Server 시스템을 확인하십시오. *client identifier* 값과 일치하는 클라이언트 ID 특성과 *subscription name*과 일치하는 구독 이름 특성이 있는 활성화 스펙이 정의되어 있는지 확인하십시오.
   - IBM MQ 구독 이름의 *client identifier* 및 *subscription name* 필드와 일치하는 클라이언트 ID와 구독 이름 특성이 있는 활성화 스펙이 없는 경우 이 지속 가능 구독을 사용하는 활성화 스펙이 없습니다. 지속 가능 구독은 삭제될 수 있습니다.
   - 지속 가능 구독 이름과 일치하는 활성화 스펙이 정의되어 있으면 이 활성화 스펙을 사용하는 메시지 구동 Bean 애플리케이션이 있는지 확인하기 위해 최종 검사를 수행해야 합니다. 이를 수행하려면:
     – 현재 확인하고 있는 지속 가능 구독을 갖는 활성화 스펙에 대한 JNDI 이름을 기록해 두십시오.
     – 설치된 각 메시지 구동 Bean 애플리케이션에 대해 WebSphere Application Server 관리 콘솔에서 구성 분할창을 불러오십시오.
     – 구성 분할창에서 메시지 구동 Bean 리스너 바인딩 링크를 클릭하십시오.
     – 메시지 구동 Bean 애플리케이션에 대한 정보가 있는 테이블이 표시됩니다. 활성화 스펙 단일 선택 단추가 바인딩 열에서 선택되고 대상 자원 JNDI 이름 필드에 지속 가능 구독을 갖는 활성화 스펙에 대한 JNDI 이름이 포함되는 경우, 구독이 여전히 사용 중이고 삭제되지 않습니다.
     – 활성화 스펙을 사용 중인 메시지 구동 Bean 애플리케이션을 발견할 수 없으면 지속 가능 구독을 삭제할 수 있습니다.

프로시저

"고아" 지속 가능한 등록이 식별되면 IBM MQ Explorer 또는 IBM MQ 명령행 유틸리티 **runmqsc**를 사용하여 삭제할 수 있습니다.

IBM MQ Explorer를 사용하여 "고아" 지속 가능 구독을 삭제하려면 다음을 수행하십시오.

1. 구독에 대한 입력 항목을 강조표시하십시오.
2. 항목을 마우스 오른쪽 단추로 클릭하고 **삭제 ...** 를 선택하십시오. 선택하십시오. 확인 창이 표시됩니다.
3. 확인 창에 표시되는 구독 이름이 올바른지 확인하고 **예**를 클릭하십시오.

이제 IBM MQ Explorer가 발행/구독 엔진에서 구독을 삭제하고 이와 연관된 내부 자원(예: 지속 가능 구독이 등록된 토픽에 대해 발행된 처리되지 않은 메시지)을 정리합니다.

IBM MQ 명령행 유틸리티 **runmqsc**를 사용하여 "고아" 지속 가능한 등록을 삭제하려면 **delete sub** 명령을 실행해야 합니다.

1. 명령 프롬프트 세션을 여십시오.
2. *WebSphere MQ*`\bin` 디렉토리로 이동하십시오.

3. 다음 명령을 입력하여 **runmqsc**를 시작하십시오.

```
runmqsc queue manager name
```

4. **runmqsc** 유틸리티가 시작되면 다음을 입력하십시오.

```
delete sub(Subscription name)
```

여기서 *Subscription name*은 다음 형식을 사용하는 지속 가능 구독의 구독 이름입니다.

- `JMS:queue manager name:client identifier:subscription name`

V 9.0.1 IBM MQ Console 및 REST API 구성

IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에 기본 구성이 제공됩니다. 이 컴포넌트 중 하나를 사용하려면 사용자의 로그인을 허용하는 보안 구성 같은 여러 구성 태스크를 수행해야 합니다. 이 주제에서는 사용 가능한 모든 구성 옵션에 대해 설명합니다.

프로시저

- 545 페이지의 『보안 구성』
- 547 페이지의 『HTTP 호스트 이름 구성』
- 548 페이지의 『HTTP 및 HTTPS 포트 구성』
- 549 페이지의 『응답 제한시간 구성』
- 550 페이지의 『자동시작 구성』
- 552 페이지의 『로깅 구성』
- 554 페이지의 『LTPA 토큰 만기 간격 구성』
- 556 페이지의 『messaging REST API 구성』
- 546 페이지의 『CSRF 보호 구성』

V 9.0.1 보안 구성

`mqwebuser.xml` 파일을 편집하여 IBM MQ Console 및 REST API 에 대한 보안을 구성할 수 있습니다. 기본 사용자 레지스트리 또는 LDAP 레지스트리 또는 WebSphere Application Server Liberty와 함께 제공되는 다른 레지스트를 구성하여 사용자를 구성 및 인증할 수 있습니다. 그런 다음 사용자 및 그룹에 역할을 지정하여 권한 부여할 수 있습니다. IBM MQ 9.0.1에서 REST API에 대한 보안이 없습니다. IBM MQ 9.0.2에서는 REST API의 보안을 구성할 수 있습니다.

이 태스크 정보

IBM MQ Console 및 REST API의 보안을 구성하려면 사용자 및 그룹을 구성해야 합니다. 그런 다음 이 사용자 및 그룹에 IBM MQ Console, REST API 또는 둘 다를 사용할 수 있는 권한을 부여할 수 있습니다. 사용자 및 그룹 구성과 사용자 인증 및 권한 부여에 대한 자세한 정보는 IBM MQ Console 및 REST API 보안을 참조하십시오.

사용자가 IBM MQ Console을 사용하여 인증될 경우 LTPA 토큰이 생성됩니다. REST API에 대해 토큰 기본 인증을 사용하는 경우, HTTP POST 메소드에서 `/login` REST API 자원을 사용하여 사용자가 로그인할 때 다른 LTPA 토큰이 생성됩니다. 이 토큰을 사용하면 토큰이 만기될 때까지 재인증 없이도 사용자가 IBM MQ Console을 사용할 수 있습니다. 토큰 만료 시점을 사용자가 구성할 수 있습니다. 추가 정보는 554 페이지의 『LTPA 토큰 만기 간격 구성』 의 내용을 참조하십시오.

프로시저

- IBM MQ Console 및 REST API 보안
- 554 페이지의 『LTPA 토큰 만기 간격 구성』

V 9.0.4 CSRF 보호 구성

CSRF (Cross-Site Request Forgery) 는 악의적인 웹 사이트가 사용자의 브라우저로 하여금 사용자가 현재 인증된 신뢰할 수 있는 사이트에서 원하지 않는 동작을 수행하게 할 때 발생하는 공격의 한 유형이다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

V 9.0.4 CSRF 보호의 현재 구성을 보려면 다음 명령을 사용하십시오.

```
dspmqweb properties -a
```

`mqRestCsrfValidation` 필드는 CSRF 유효성 검사가 수행되는지 여부를 표시합니다. 자세한 정보는 dspmqweb을 참조하십시오.

참고: V 9.0.5 CSRF 만기 시간을 표시하기 위해 IBM MQ 9.0.4 에 소개된 `mqRestCsrfExpirationInMinutes` 필드가 IBM MQ 9.0.5에 더 이상 존재하지 않습니다.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 `crtmqweb.sh`로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

이 태스크 정보

V 9.0.5 IBM MQ 9.0.5 전에서는 IBM MQ Console 및 REST API가 CSRF 공격으로부터 보호하기 위해 동기화 프로그램 토큰을 사용합니다 IBM MQ 9.0.4에서만, 관리 REST API용 CSRF 동기화 프로그램 토큰이 정기적으로 다시 생성됩니다. IBM MQ 9.0.5부터는 CSRF 동기화 프로그램 토큰이 사용되지 않습니다. 그 대신 사용자 정의 HTTP 헤더를 설정해야 하며 이는 동기화 프로그램 토큰을 사용하는 것과 동일한 보호를 제공합니다.

setmqweb properties 명령을 사용하여 REST API 에 대한 CSRF 보호 구성을 수정할 수 있습니다.

프로시저

- REST API에 대해 CSRF 토큰 유효성 검증을 구성하려면 다음 방법을 사용하십시오.
  - IBM MQ 9.0.4에서만 토큰 만기를 변경하기 위해 **setmqweb properties** 명령을 사용하십시오.

    setmqweb properties -k mqRestCsrfExpirationInMinutes -v *time*

    여기서 *time*은 CSRF 토큰 만기 전 시간(분)을 지정합니다. 토큰은 만기된 후 다음 HTTP POST, PATCH 또는 DELETE 메소드에 대해 유효한 상태로 남아 있습니다. 그 이후에는 새 토큰이 쿠키로 리턴되고 이전 토큰 값은 무효화됩니다. 시간 값 -1은 CSRF 토큰 만기를 사용 안함으로 설정하고, 값 0은 POST, PATCH 또는 DELETE 요청이 있을 때마다 토큰을 변경합니다. 기본값은 30분입니다.
  - **setmqweb properties** 명령을 사용하여 CSRF 유효성 검증 검사를 제거하십시오.

    setmqweb properties -k mqRestCsrfValidation -v *boolean*

    여기서 *boolean*은 CSRF 유효성 검증 검사를 수행할지 여부를 지정합니다. 값 false는 CSRF 토큰 유효성 검증 검사를 제거합니다. 특히 사용자가 웹 브라우저를 사용하여 REST API에 액세스하는 경우에는 토큰

유효성 검증이 권장됩니다. 기본값은 true입니다. REST API를 통해 모든 HTTP POST, PATCH 및 DELETE 요청에 대해 CSRF 토큰의 유효성을 검증합니다.

V 9.0.1 HTTP 호스트 이름 구성

기본적으로 IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버는 로컬 연결만 허용하도록 구성되어 있습니다. 즉 IBM MQ Console 및 REST API는 IBM MQ Console 및 REST API가 설치된 시스템에서만 액세스할 수 있습니다. V 9.0.4 IBM MQ 9.0.4부터는 **setmqweb** 명령을 사용하여 원격 연결이 허용되도록 호스트 이름을 구성할 수 있습니다. IBM MQ 9.0.3에서 `mqwebuser.xml` 파일을 편집하여 원격 연결을 허용하도록 호스트 이름을 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

V 9.0.4 IBM MQ 9.0.4에서는 다음 명령을 사용하여 HTTP 호스트 이름의 현재 구성을 볼 수 있습니다.

```
dspmqweb properties -a
```

`httpHost` 필드에는 HTTP 호스트 이름이 표시됩니다. 자세한 정보는 dspmqweb을 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *`WLP_user_directory`*는 `crtmqweb.sh`로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

프로시저

- V 9.0.4

  다음 방법 중 하나를 사용하여 호스트 이름을 구성하십시오.

  - IBM MQ 9.0.4에서 **setmqweb properties** 명령을 사용하십시오.

    `setmqweb properties -k httpHost -v` *`hostName`*

    여기서 *hostName*은 IBM MQ가 설치된 서버의 IP 주소, DNS(도메인 이름 서버) 호스트 이름(도메인 이름 접미부 포함) 또는 DNS 호스트 이름을 지정합니다. 사용 가능한 모든 네트워크 인터페이스를 지정하려면 별표를 큰따옴표로 묶어 사용하십시오. 로컬 연결만 허용하려면 `localhost` 값을 사용하십시오.
  - IBM MQ 9.0.3 및 이전 버전의 경우 `mqwebuser.xml` 파일을 편집하십시오.

    1. `mqwebuser.xml` 파일을 여십시오.

       `mqwebuser.xml`은 다음 디렉토리 중 하나에 있습니다.

       - ULW UNIX, Linux, and Windows의 경우: *`MQ_DATA_DIRECTORY`*`/web/installations/`*`installationName`*`/servers/mqweb`
       - z/OS z/OS: *`WLP_user_directory`*`/servers/mqweb`

       여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.

2. mqweb 서버를 구성하십시오.

- mqweb 서버에 대한 원격 연결을 허용하려면 mqwebuser.xml 파일의 <server> 태그 내에 다음 행을 추가하십시오.

  ```
  <variable name="httpHost" value="hostName"/>
  ```

  여기서 *hostName*은 IBM MQ가 설치된 서버의 IP 주소, DNS(도메인 이름 서버) 호스트 이름(도메인 이름 접미부 포함) 또는 DNS 호스트 이름을 지정합니다. 사용 가능한 모든 네트워크 인터페이스를 지정하려면 별표(*)를 사용하십시오.

- mqweb 서버에 로컬 연결만 허용하려면 mqwebuser.xml 파일에서 다음 행을 제거하거나 값을 localhost로 설정하십시오.

  ```
  <variable name="httpHost" value="hostName"/>
  ```

V 9.0.1 HTTP 및 HTTPS 포트 구성

기본적으로 IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버는 HTTPS 포트 9443을 사용합니다. HTTP 연결과 연관된 포트는 사용할 수 없습니다. HTTP 포트를 사용으로 설정하거나, 다른 HTTP 포트를 구성하거나, HTTP 또는 HTTPS 포트를 사용 안함으로 설정할 수 있습니다. V 9.0.4 IBM MQ 9.0.4부터는 **setmqweb** 명령을 사용하여 포트를 구성할 수 있습니다. IBM MQ 9.0.3에서는 mqwebuser.xml 파일을 편집하여 포트를 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

HTTP 포트와 HTTPS 포트를 모두 사용으로 설정하려면 HTTPS 요청에 대해 발생되는 LTPA 토큰을 브라우저의 HTTP 요청에 재사용할 수 있습니다. 이 작동을 방지하기 위해 mqweb 서버를 구성하고 다음 행을 mqwebuser.xml 파일에 추가하여 환경의 보안을 강화할 수 있습니다.

```
<webAppSecurity ssoRequiresSSL="true"/>
```

V 9.0.4 IBM MQ 9.0.4에서는 다음 명령을 사용하여 HTTP 및 HTTPS 포트의 현재 구성을 볼 수 있습니다.

```
dspmqweb properties -a
```

httpPort 필드에는 HTTP 포트가 표시되고, httpsPort 필드에는 HTTPS 포트가 표시됩니다. 자세한 정보는 dspmqweb을 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

프로시저

- V 9.0.4
  다음 방법 중 하나를 사용하여 포트를 구성하십시오.
  - IBM MQ 9.0.4에서 **setmqweb properties** 명령을 사용하십시오.

- HTTP 포트를 사용으로 설정하거나 구성하려면 다음 명령을 사용하십시오.

  `setmqweb properties -k httpPort -v portNumber`

  여기서 *portNumber*는 HTTP 연결에 사용할 포트를 지정합니다. -1 값을 사용하여 포트를 사용 불가능하게 설정할 수 있습니다.
- HTTPS 포트를 구성하려면 다음 명령을 사용하십시오.

  `setmqweb properties -k httpsPort -v portNumber`

  여기서 *portNumber*는 HTTPS 연결에 사용할 포트를 지정합니다. -1 값을 사용하여 포트를 사용 불가능하게 설정할 수 있습니다.

• IBM MQ 9.0.3 및 이전 버전의 경우 mqwebuser.xml 파일을 편집하십시오.

1. mqwebuser.xml 파일을 여십시오.

   mqwebuser.xml은 다음 디렉토리 중 하나에 있습니다.

   - ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/servers/mqweb
   - z/OS z/OS: *WLP_user_directory*/servers/mqweb

     여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.
2. 포트를 구성하십시오.

   - HTTP 포트를 사용 설정 또는 구성하려면 mqwebuser.xml 파일에서 <server> 태그 내부에 다음 행을 추가하거나 편집하십시오.

     `<variable name="httpPort" value="portNumber" />`

     여기서 *portNumber*는 HTTP 연결에 사용할 포트를 지정합니다. -1 값을 사용하여 포트를 사용 불가능하게 설정할 수 있습니다.
   - HTTPS 포트를 구성하려면 mqwebuser.xml 파일의 <server> 태그 내부에서 다음 행을 추가하거나 편집하십시오.

     `<variable name="httpsPort" value="portNumber" />`

     여기서 *portNumber*는 HTTPS 연결에 사용할 포트를 지정합니다. -1 값을 사용하여 포트를 사용 불가능하게 설정할 수 있습니다.

V 9.0.1 응답 제한시간 구성

기본적으로 응답을 클라이언트에 다시 전송하는 데 걸리는 시간이 30초보다 길 경우 IBM MQ Console 및 REST API의 제한시간이 초과됩니다. V 9.0.4 IBM MQ 9.0.4에서 **setmqweb** 명령을 사용하여 다른 제한시간 값을 사용하도록 IBM MQ Console 및 REST API 를 구성할 수 있습니다. IBM MQ 9.0.3에서 mqwebuser.xml 파일을 편집하여 다른 제한시간 값을 사용하도록 IBM MQ Console 및 REST API 를 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

V 9.0.4 IBM MQ 9.0.4부터는 다음 명령을 사용하여 REST API 응답 제한시간의 현재 구성을 볼 수 있습니다.

```
dspmqweb properties -a
```

mqRestRequestTimeout 필드에는 응답 제한시간의 현재 값이 표시됩니다. 자세한 정보는 dspmqweb을 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

프로시저

- V 9.0.4
  다음 방법 중 하나를 사용하여 제한시간을 구성하십시오.
  - IBM MQ 9.0.4에서 **setmqweb properties** 명령을 사용하십시오.

    `setmqweb properties -k mqRestRequestTimeout -v timeout`

    여기서 *timeout*은 제한시간이 초과되기 전의 시간(초)을 지정합니다.
  - IBM MQ 9.0.3 및 이전 버전의 경우 mqwebuser.xml 파일을 편집하십시오.
    1. mqwebuser.xml 파일을 여십시오.

       mqwebuser.xml은 다음 디렉토리 중 하나에 있습니다.

       - ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/servers/mqweb
       - z/OS z/OS: *WLP_user_directory*/servers/mqweb

         여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.
    2. mqwebuser.xml 파일의 <server> 태그 내에서 다음 행을 추가하거나 편집하여 제한시간을 구성하십시오.

       `<variable name="mqRestRequestTimeout" value="timeout" />`

       여기서 *timeout*은 제한시간이 초과되기 전의 시간(초)을 지정합니다.

V 9.0.1 자동시작 구성

기본적으로 mqweb 서버가 시작되면 IBM MQ Console이 자동으로 시작됩니다. IBM MQ 9.0.1에서는 REST API가 자동으로 시작되지 않습니다. IBM MQ 9.0.2에서 REST API는 mqweb 서버가 시작할 때 자동으로 시작됩니다. V 9.0.4 IBM MQ 9.0.4에서 **setmqweb** 명령을 사용하여 IBM MQ Console 및 REST API 가 자동으로 시작되는지 여부를 구성할 수 있습니다. IBM MQ 9.0.3에서 mqwebuser.xml 파일을 편집하여 IBM MQ Console 및 REST API 가 자동으로 시작되는지 여부를 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

V 9.0.4 IBM MQ 9.0.4에서는 다음 명령을 사용하여 REST API 자동시작의 현재 구성을 볼 수 있습니다.

```
dspmqweb properties -a
```

mqRestAutostart 필드는 REST API 가 자동으로 시작되는지 여부를 표시하고 mqConsoleAutostart 필드는 IBM MQ Console 가 자동으로 시작되는지 여부를 표시합니다. 자세한 정보는 dspmqweb을 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

프로시저

- V 9.0.4

  다음 방법 중 하나를 사용하여 IBM MQ Console 및 REST API의 자동 시작 여부를 구성하십시오.

  - IBM MQ 9.0.4에서 **setmqweb properties** 명령을 사용하십시오.
    - 다음 명령을 사용하여 IBM MQ Console을 자동으로 시작할지 여부를 구성하십시오.

      ```
      setmqweb properties -k mqconsoleAutostart -v start
      ```

      여기서 *start* 는 IBM MQ Console 가 자동으로 시작되도록 하려는 경우의 값 True 이고 그렇지 않으면 False 입니다.
    - 다음 명령을 사용하여 REST API를 수동으로 시작할지 여부를 구성하십시오.

      ```
      setmqweb properties -k mqRestAutostart -v start
      ```

      여기서 *start* 는 REST API 가 자동으로 시작되도록 하려는 경우의 값 True 이고 그렇지 않으면 False 입니다.
  - IBM MQ 9.0.3 및 이전 버전의 경우 mqwebuser.xml 파일을 편집하십시오.
    1. mqwebuser.xml 파일을 여십시오.

       mqwebuser.xml은 다음 디렉토리 중 하나에 있습니다.

       - ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/servers/mqweb
       - z/OS z/OS: *WLP_user_directory*/servers/mqweb

       여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.
    2. 자동시작을 구성하십시오.
       - <server> 태그 내에서 mqwebuser.xml 파일에 다음 행을 추가하거나 업데이트하여 IBM MQ Console 에 수동 시작이 필요한지 여부를 구성하십시오.

         ```
         <variable name="mqConsoleAutostart" value="start"/>
         ```

         여기서 *start* 는 IBM MQ Console 가 자동으로 시작되도록 하려는 경우의 값 True 이고 그렇지 않으면 False 입니다.
       - <server> 태그 내에서 mqwebuser.xml 파일에 다음 행을 추가하거나 업데이트하여 REST API 에 수동 시작이 필요한지 여부를 구성하십시오.

         ```
         <variable name="mqRestAutostart" value="start"/>
         ```

여기서 *start* 는 REST API 가 자동으로 시작되도록 하려는 경우의 값 True 이고 그렇지 않으면 False 입니다.

V 9.0.1 로깅 구성

IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에서 사용하는 로그 파일의 로깅 레벨, 최대 로그 파일 크기 및 최대 로그 파일 수를 구성할 수 있습니다. V 9.0.4 IBM MQ 9.0.4에서는 **setmqweb** 명령을 사용하여 로깅을 구성할 수 있습니다. IBM MQ 9.0.3에서 mqwebuser.xml 파일을 편집하여 로깅을 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

V 9.0.4 IBM MQ 9.0.4부터는 다음 명령을 사용하여 REST API 로깅의 현재 구성을 볼 수 있습니다.

```
dspmqweb properties -a
```

maxTraceFileSize 필드에는 최대 추적 파일 크기가 표시되고, maxTraceFiles 필드에는 최대 추적 파일 수가 표시되며, traceSpec 필드에는 사용된 추적 레벨이 표시됩니다. 자세한 정보는 dspmqweb을 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

이 태스크 정보

mqweb 서버에 대한 로그 파일은 다음 디렉토리 중 하나에 있습니다.

- ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/servers/mqweb/logs
- z/OS z/OS: *WLP_user_directory*/servers/mqweb/logs

  여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.

IBM MQ Console 및 REST API의 추적 사용에 대한 자세한 정보는 IBM MQ Console 및 REST API 추적을 참조하십시오.

프로시저

- V 9.0.4

  다음 방법 중 하나를 사용하여 로깅을 구성하십시오.

  - IBM MQ 9.0.4에서 **setmqweb properties** 명령을 사용하십시오.
    - 최대 로그 파일 크기를 설정하려면 다음 명령을 사용하십시오.

```
setmqweb properties -k maxTraceFileSize -v size
```

여기서 *size*는 각 로그 파일이 도달할 수 있는 크기(MB)를 지정합니다. 기본값은 20입니다.

- 로깅에 사용할 최대 파일 수를 설정하려면 다음 명령을 사용하십시오.

```
setmqweb properties -k maxTraceFiles -v max
```

여기서 *max*는 최대 파일 수를 지정합니다. 기본값은 2입니다.

- 사용되는 로깅 레벨을 구성하려면 다음 명령을 사용하십시오.

```
setmqweb properties -k traceSpec -v level
```

여기서 *level*은 553 페이지의 표 37에 나열된 값 중 하나입니다. 표에서는 세부사항 수준을 높인 로깅 레벨을 간략하게 설명합니다. 로깅 레벨을 사용할 경우 바로 앞의 로깅 레벨도 사용 가능하게 설정해야 합니다. 예를 들어, **`*=warning`** 로깅 레벨을 사용할 경우 **`*=severe`** 및 **`*=fatal`** 로깅 레벨도 사용 가능하게 설정해야 합니다.

기본값은 **`*=info`**입니다. IBM Service가 요청할 경우 이 값을 변경하십시오.

표 *37.* 올바른 로깅 레벨

가치	**적용되는 로깅 레벨**
*=off	로깅이 꺼져 있습니다.
*=fatal	태스크를 계속할 수 없고, 컴포넌트, 애플리케이션 및 서버가 작동하지 않습니다.
*=severe	태스크를 계속할 수 없지만, 컴포넌트, 애플리케이션 및 서버는 계속 작동합니다. 또한 이 레벨은 복구할 수 없는 임박한 오류를 나타낼 수 있습니다.
* =경고	잠재적인 오류 또는 임박한 오류입니다. 또한 이 레벨은 점진적인 실패(예: 자원의 잠재적인 누출)를 나타낼 수도 있습니다.
*=audit	서버 상태 또는 자원에 영향을 미치는 심각한 이벤트입니다.
*=info	전체적인 태스크 진행 상태를 요약하는 일반 정보입니다.
*=config	구성 변경 또는 상태입니다.
*=detail	하위 태스크 진행 상태를 설명하는 일반 정보입니다.
*=fine	추적 정보 - 일반 추적 + 메소드 입력, 종료 및 리턴 값
*=finer	추정 정보 - 상세 추적
*=finest	추적 정보 - 문제점을 디버그하는 데 필요한 모든 세부사항을 포함하는 더 상세한 추적
* =all	모든 이벤트가 로깅됩니다.

- IBM MQ 9.0.3 및 이전 버전의 경우 `mqwebuser.xml` 파일을 편집하십시오.
  1. `mqwebuser.xml` 파일을 여십시오.

     `mqwebuser.xml`은 다음 디렉토리 중 하나에 있습니다.

     - ULW UNIX, Linux, and Windows의 경우: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
     - z/OS z/OS: `WLP_user_directory/servers/mqweb`

여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.

2. 로깅 구성:

   - 최대 로그 파일 크기를 설정하려면 mqwebuser.xml 파일의 <server> 태그 내에 다음 행을 추가하거나 편집하십시오.

     ```
     <variable name="maxTraceFileSize" value="size" />
     ```

     여기서 *size*는 각 로그 파일이 도달할 수 있는 크기(MB)를 지정합니다. 기본값은 20입니다.

   - 로깅에 사용할 최대 파일 수를 설정하려면 mqwebuser.xml 파일의 <server> 태그 내에서 다음 행을 추가하거나 편집하십시오.

     ```
     <variable name="maxTraceFiles" value="max"/>
     ```

     여기서 *max*는 최대 파일 수를 지정합니다. 기본값은 2입니다.

   - 사용되는 로깅 레벨을 구성하려면 mqwebuser.xml 파일의 <server> 태그 내에서 다음 행을 추가하거나 편집하십시오.

     ```
     <variable name="traceSpec" value="level"/>
     ```

     여기서 *level*은 553 페이지의 표 37에 나열된 값 중 하나입니다.

     표에서는 세부사항 수준을 높인 로깅 레벨을 간략하게 설명합니다. 로깅 레벨을 사용할 경우 바로 앞의 로깅 레벨도 사용 가능하게 설정해야 합니다. 예를 들어, ***=warning** 로깅 레벨을 사용할 경우 ***=severe** 및 ***=fatal** 로깅 레벨도 사용 가능하게 설정해야 합니다.

     기본값은 ***=info**입니다. IBM Service가 요청할 경우 이 값을 변경하십시오.

LTPA 토큰 만기 간격 구성

LTPA 토큰을 사용하면 사용자가 요청 시마다 사용자 이름과 비밀번호 신임 정보를 WebSphere Application Server Liberty에 제공할 필요가 없습니다. LTPA 인증 토큰의 만기 간격을 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

V 9.0.4 IBM MQ 9.0.4에서는 **dspmqweb properties** 명령을 -a 플래그와 함께 사용하여 토큰 만기의 현재 구성을 볼 수 있습니다. 자세한 정보는 dspmqweb을 참조하십시오. **setmqweb properties** 명령을 -k 및 -d 플래그와 함께 사용하여 토큰 만기의 값을 재설정할 수 있습니다. 자세한 정보는 setmqweb을 참조하십시오.

V 9.0.2

참고: REST API에 대해 토큰 인증과 IBM MQ Console을 모두 사용하는 경우 만기 간격이 공유됩니다.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

이 태스크 정보

사용자가 IBM MQ Console에 로그인하면 LTPA 토큰이 생성됩니다. REST API에 대해 토큰 기반 인증을 사용하는 경우, HTTP POST 메소드에서 /login REST API 자원을 사용하여 사용자가 로그인할 때 LTPA 토큰이 생성됩니다. 이 토큰은 토큰이 만료될 때까지 사용자가 사용자 ID 및 비밀번호를 사용하여 다시 로그인할 필요 없이 사용자를 인증하는 데 사용됩니다. 기본 만기 간격은 120분입니다. V 9.0.4 IBM MQ 9.0.4부터는 **setmqweb** 명령을 사용하여 토큰 만료 시기를 구성할 수 있습니다. IBM MQ 9.0.3및 이전 버전에서는 mqwebuser.xml 파일을 편집하여 토큰이 만기되는 시기를 구성할 수 있습니다.

프로시저

- V 9.0.4

  다음 방법 중 하나를 사용하여 토큰 만기를 구성하십시오.

  - IBM MQ 9.0.4에서 **setmqweb properties** 명령을 사용하십시오.

    ```
    setmqweb properties -k ltpaExpiration -v time
    ```

    여기서 *time*은 LTPA 토큰이 만기되고 사용자가 로그아웃하기 전의 시간(분)을 지정합니다. 기본값은 120분입니다.
  - IBM MQ 9.0.3 및 이전 버전의 경우 mqwebuser.xml 파일을 편집하십시오.

    1. mqwebuser.xml 파일을 여십시오.

       mqwebuser.xml은 다음 디렉토리 중 하나에 있습니다.

       - ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/servers/mqweb
       - z/OS z/OS: *WLP_user_directory*/servers/mqweb

       여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.
    2. mqwebuser.xml 파일의 <server> 태그 내에서 다음 행을 추가 또는 편집하여 LTPA 토큰 만료 간격을 구성하십시오.

       ```
       <variable name="ltpaExpiration" value="time" />
       ```

       여기서 *time*은 LTPA 토큰이 만기되고 사용자가 로그아웃하기 전의 시간(분)을 지정합니다. 기본값은 120분입니다.

V 9.0.4 administrative REST API 게이트웨이 구성

기본적으로 administrative REST API 게이트웨이가 사용됩니다. administrative REST API 게이트웨이가 사용되면 게이트웨이 큐 관리자를 사용하여 REST API로 원격 관리를 수행할 수 있습니다. 기본 게이트웨이 큐 관리자로 사용할 큐 관리자를 구성하거나, **setmqweb properties** 명령으로 administrative REST API 게이트웨이를 사용 안함으로 설정하여 원격 관리를 금지할 수 있습니다.

이 태스크 정보

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

administrative REST API 게이트웨이의 현재 구성을 보려면 다음 명령을 사용하십시오.

```
dspmqweb properties -a
```

mqRestGatewayEnabled 필드에는 게이트웨이가 사용으로 설정되었는지 여부가 표시되고, mqRestGatewayQmgr 필드에는 기본 게이트웨이 큐 관리자의 이름이 표시됩니다. 자세한 정보는 dspmqweb을 참조하십시오.

기본 게이트웨이 큐 관리자는 다음 두 가지 사항을 모두 충족할 때 사용됩니다.

- 큐 관리자가 REST 요청의 `ibm-mq-rest-gateway-qmgr` 헤더에 지정되어 있지 않습니다.
- REST API 자원 URL에 지정된 큐 관리자가 로컬 큐 관리자가 아닙니다.

REST API를 사용하는 원격 관리에 대한 자세한 정보는 REST API를 사용한 원격 관리를 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 `crtmqweb.sh`로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

프로시저

- 다음 명령을 사용하여 administrative REST API 게이트웨이를 사용할지를 구성하십시오.

  `setmqweb properties -k mqRestGatewayEnabled -v enabled`

  여기서 administrative REST API 게이트웨이를 사용으로 설정하려면 *enabled*가 **true** 값이고 그렇지 않은 경우 **false**입니다.
- 기본 게이트웨이 큐 관리자로 사용할 큐 관리자를 구성하십시오.
  - 기본 게이트웨이 큐 관리자를 설정하려면 다음 명령을 사용하십시오.

    `setmqweb properties -k mqRestGatewayQmgr -v qmgrName`

    여기서 *qmgrName*은 mqweb 서버와 동일한 설치에 있는 큐 관리자의 이름입니다.
  - 기본 게이트웨이 큐 관리자를 설정 해제하려면 다음 명령을 사용하십시오.

    `setmqweb properties -k mqRestGatewayQmgr -d`

V 9.0.4 messaging REST API 구성

기본적으로 IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에서는 messaging REST API가 사용으로 설정되어 있습니다. **setmqweb properties** 명령을 사용하여 메시징을 사용 또는 사용 안함으로 설정할지 여부를 구성할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

messaging REST API의 현재 구성을 보려면 다음 명령을 사용하십시오.

```
dspmqweb properties -a
```

`mqRestMessagingEnabled` 필드에는 messaging REST API가 사용 또는 사용 안함으로 설정되었는지 여부가 표시됩니다. 자세한 정보는 dspmqweb을 참조하십시오.

messaging REST API를 사용하려면 호출자는 mqweb 서버에서 인증을 받아야 하며 `MQWebUser` 역할의 구성원이어야 합니다. `MQWebAdmin` 및 `MQWebAdminRO` 역할은 messaging REST API에 적용할 수 없습니다. 호출자는 또한 OAM/RACF를 통해 인증을 받아야 합니다. REST API에 대한 자세한 정보는 IBM MQ 콘솔 및 REST API 보안을 참조하십시오.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

프로시저

- V 9.0.4
  messaging REST API를 구성하려면 다음 방법을 사용하십시오.
  - **setmqweb properties** 명령을 사용하십시오.
    - 다음 명령을 사용하여 messaging REST API를 사용할지를 구성하십시오.

      setmqweb properties -k mqRestMessagingEnabled -v *enabled*

      여기서 messaging REST API를 사용으로 설정하려면 *enabled*가 true 값이고 그렇지 않은 경우 false입니다.

V 9.0.5 MFT용 REST API 구성

기본적으로 IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에서는 MFT REST API가 사용 안함으로 설정됩니다. **setmqweb properties** 명령을 사용하여 MFT의 REST API를 사용 또는 사용 안함으로 설정하고, 조정 큐 관리자를 설정하고 MFT 다시 연결 제한시간을 지정할 수 있습니다.

시작하기 전에

이 프로시저를 완료하려면 권한이 있는 사용자여야 합니다.

다음 명령을 사용하여 MFT의 REST API에 대한 현재 구성을 볼 수 있습니다.

```
dspmqweb properties -a
```

mqRestMftEnabled 필드는 MFT의 REST API가 사용으로 설정되었는지 또는 사용 안함으로 설정되었는지 여부를 표시합니다. mqRestMftCoordinationQmgr 필드는 조정 큐 관리자의 이름을 표시하고, mqRestMftReconnectTimeoutInMinutes 필드는 MFT 요청의 제한시간 값을 표시합니다. 자세한 정보는 dspmqweb을 참조하십시오.

MFT에 REST API 를 사용하려면 호출자가 mqweb 서버에 대해 인증되어야 하며 하나 이상의 MFTWebAdmin또는 MFTWebAdminRO 역할의 구성원이어야 합니다.

주의: V 9.0.4 z/OS

z/OS에서 **setmqweb** 또는 **dspmqweb** 명령 중 하나를 발행하기 전에 WLP_USER_DIR 환경 변수를 설정해야 하므로 변수는 mqweb 서버 구성을 가리킵니다.

이를 수행하려면 다음 명령을 실행하십시오.

```
export WLP_USER_DIR=WLP_user_directory
```

여기서 *WLP_user_directory*는 crtmqweb.sh로 전달된 디렉토리의 이름입니다. 예를 들면, 다음과 같습니다.

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

자세한 정보는 Liberty 서버 정의 작성을 참조하십시오.

이 태스크 정보

MFT의 REST API를 구성하는 경우 세 가지 특성을 구성할 수 있습니다.

- MFT의 REST API가 사용 가능한지 여부. 기본적으로 이는 사용 안함으로 설정됩니다.
- MFT REST API 자원을 사용하는 경우 정보를 검색하는 조정 큐 관리자의 이름. 이 큐 관리자는 mqweb 서버와 동일한 시스템에 있는 큐 관리자여야 합니다. MFT의 REST API는 mqweb 서버가 시작되면 이 큐 관리자에 대한 바인딩 연결을 설정합니다.

  기본적으로 이 큐 관리자 이름은 공백입니다. 값이 설정되지 않고 MFT REST API가 호출되면 HTTP 400이 리턴됩니다.
- MFT의 REST API가 조정 큐 관리자에 대한 연결 시도를 중지한 후의 제한시간(분)입니다. 연결을 재설정하기 위한 첫 번째 시도는 조정 큐 관리자와의 연결이 단절된 후 바로 수행됩니다. 이에 실패하면 모든 재연결 시도 사이에 5분의 간격이 있게 됩니다.

  다시 연결 제한시간이 초과되면, `/transfer` 또는 `/agent` REST API 자원을 호출할 때 다시 연결을 시도합니다. 이 재연결 시도가 실패하면 재연결 제한시간이 경과할 때까지 MFT에서 5분마다 재연결을 다시 시도합니다.

  기본적으로 제한시간 값은 30분입니다. 조정 큐 관리자가 시작되지 않은 상태에서 MFT REST API가 호출되면 HTTP 503이 리턴됩니다.

프로시저

1. MFT의 REST API 구성을 조정하십시오.
   - 다음 명령을 사용하여 MFT의 REST API가 사용으로 설정되었는지 여부를 구성하십시오.

     `setmqweb properties -k mqRestMftEnabled -v` *`value`*

     여기서 *value* 는 MFT 의 경우 REST API 이 사용 가능하도록 하려는 경우 `true` 이고 그렇지 않은 경우 `false` 입니다.
   - 다음 명령을 사용하여 전송 세부사항을 검색하는 조정 큐 관리자를 구성하십시오.

     `setmqweb properties -k mqRestMftCoordinationQmgr -v` *`qmgrName`*

     여기서 *qmgrName*은 조정 큐 관리자의 이름입니다. 조정 큐 관리자는 mqweb 서버가 실행 중인 시스템에 있어야 합니다.
   - 다음 명령을 사용하여 MFT의 REST API가 조정 큐 관리자에 대한 연결 시도를 중지하는 제한시간(분)을 구성하십시오.

     `setmqweb properties -k mqRestMftReconnectTimeoutInMinutes -v` *`time`*

     여기서 *time*은 제한시간이 초과되기 전까지의 시간(분)을 지정합니다.

     - `0-5` 사이의 값은 MFT 의 REST API 가 조정 큐 관리자에 한 번만 재연결을 시도하도록 지정합니다. 연결이 실패한 경우 REST API가 호출될 때까지 연결을 재설정하는 시도가 없습니다.
     - `-1` 값은 MFT 의 REST API 가 연결이 성공할 때까지 재연결을 시도하도록 지정합니다.
2. 다음 명령을 입력하여 mqweb 서버를 재시작하십시오.

```
endmqweb
strmqweb
```

V 9.0.2 mqweb 서버 JVM 성능 조정

기본적으로 mqweb 서버 JVM(Java Virtual Machine)은 힙의 최소 및 최대 크기에 대해 플랫폼별 기본값을 사용합니다. 기본값을 변경해야 할 수 있습니다. 예를 들어, mqweb 서버에서 `java.lang.OutOfMemoryError`가 발생하는 경우 힙의 최대 크기를 늘려야 합니다. `jvm.options` 파일에서 기본값을 변경할 수 있습니다.

프로시저

1. `jvm.options` 파일을 여십시오.

   `jvm.options` 파일은 다음 디렉토리 중 하나에서 찾을 수 있습니다.

   - ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/servers/mqweb
   - z/OS z/OS: *WLP_user_directory*/servers/mqweb

   여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.
2. 옵션: 다음 행을 파일에 추가하여 최대 힙 크기를 설정하십시오.

   ```
   -XmxMaxSizem
   ```

   여기서 *MaxSize*는 힙의 최대 크기(MB)를 지정합니다.
   예를 들어, 다음 행은 최대 힙 크기를 1GB로 설정합니다.

   ```
   -Xmx1024m
   ```
3. 옵션: 다음 행을 파일에 추가하여 최소 힙 크기를 설정하십시오.

   ```
   -XmsMinSizem
   ```

   여기서 *MinSize*는 힙의 최소 크기(MB)를 지정합니다.
   예를 들어, 다음 행은 최소 힙 크기를 512MB로 설정합니다.

   ```
   -Xms512m
   ```
4. 명령행에서 다음 명령을 입력하여 mqweb 서버를 재시작하십시오.

   ```
   endmqweb
   strmqweb
   ```

IBM MQ Console 및 REST API 설치 컴포넌트의 파일 구조

IBM MQ Console 및 REST API 설치 컴포넌트와 연관된 두 가지 세트의 디렉토리 구조가 있습니다. 하나의 디렉토리 구조에는 편집할 수 있는 파일이 포함됩니다. 다른 디렉토리 구조에는 편집할 수 없는 파일이 포함됩니다.

편집 가능한 파일

IBM MQ Console 및 REST API 설치 컴포넌트의 최초 설치 중에 사용자가 편집할 수 있는 파일이 규정됩니다. 이 파일은 편집할 수 있기 때문에 유지보수가 적용될 때 변경되지 않습니다.

사용자가 편집할 수 있는 파일의 위치는 운영 체제에 따라 다릅니다.

- ULW UNIX, Linux, and Windows의 경우: *MQ_DATA_DIRECTORY*/web/installations/*installationName*/
- z/OS z/OS: *WLP_user_directory*

  여기서 *WLP_user_directory*는 mqweb 서버 정의를 작성하기 위해 **crtmqweb.sh** 스크립트를 실행할 때 지정된 디렉토리입니다.

이 최상위 레벨 디렉토리 아래에 다음 디렉토리 및 파일이 있습니다.

디렉토리 및 파일	설명
angular.persistence/	IBM MQ Console 대시보드 구성이 저장되는 디렉토리입니다.

디렉토리 및 파일	설명
servers/	WebSphere Liberty Profile 서버 디렉토리입니다.
servers/mqweb	mqweb 서버 디렉토리 구조를 포함하는 디렉토리입니다.
servers/mqweb/logs	mqweb 서버에 대한 로그를 포함하는 디렉토리입니다.
servers/mqweb/logs/console.log	기본 서버 상태 및 조작 메시지의 로그입니다.
servers/mqweb/logs/ffdc	FFDC(First Failure Data Capture) 출력 디렉토리입니다.
servers/mqweb/logs/messages.log	mqweb 서버의 런타임 메시지 로그입니다(IBM MQ Console 및 REST API 포함). 이전 메시지는 messages_*timestamp*.log라는 파일에 저장됩니다.
servers/mqweb/logs/trace.log	mqweb 서버의 추적 로그입니다(IBM MQ Console 및 REST API 포함). 이전 추적은 trace_*timestamp*.log라고 하는 파일에 저장됩니다. 추적을 사용하는 경우에만 이 파일이 있습니다.
servers/mqweb/logs/state	서버 관련 상태입니다.
servers/mqweb/server.xml	기본 서버 구성 파일입니다. 이 파일은 읽기 전용입니다. 기본 구성을 대체하려면 mqwebuser.xml 파일을 편집하십시오.
servers/mqweb/mqwebuser.xml	IBM MQ Console 및 REST API에 대한 구성 파일입니다. 이 파일에 구성된 설정이 기본 구성을 대체합니다. 이 파일을 편집하려면 권한이 있는 사용자여야 합니다.
servers/mqweb/resources	키 저장소 같은 다양한 서버 자원을 포함하는 디렉토리입니다.
servers/mqweb/workarea	서버가 조작할 때 작성한 디렉토리입니다. 서버가 처음 실행된 후 이 디렉토리가 작성됩니다.

편집 불가능한 파일

IBM MQ Console 및 REST API 설치 컴포넌트의 최초 설치 중에 편집 불가능한 파일이 규정됩니다. 유지보수가 적용될 때 이 파일이 업데이트됩니다.

사용자가 편집할 수 있는 파일의 위치는 운영 체제에 따라 다릅니다.

- ULW UNIX, Linux, and Windows의 경우: *MQ_INSTALLATION_PATH*/web
- z/OS z/OS: *installation_directory*/web/

  여기서 *installation_directory*는 IBM MQ UNIX System Services 컴포넌트 설치 경로입니다.

이 위치에는 다음 디렉토리 구조 및 파일이 있습니다.

디렉토리 및 파일	설명
bin/	Liberty 명령을 포함하는 디렉토리입니다. 이 디렉토리에서 스크립트를 실행하려면 권한이 있는 사용자여야 합니다.

디렉토리 및 파일	설명
mq/	다양한 IBM MQ 자원을 포함하는 디렉토리 구조입니다.
mq/apps/	IBM MQ Console 및 REST API 애플리케이션을 포함하는 디렉토리입니다.
mq/etc/	
mq/etc/mqweb.xml	mqweb 서버에 대한 읽기 전용 구성 파일입니다. 구성을 변경하려면 mqwebuser.xml 파일을 편집하십시오.
mq/libs	IBM MQ Console 및 REST API에서 사용할 공유 라이브러리를 포함하는 디렉토리입니다.
mq/samp	샘플을 포함하는 디렉토리입니다.
mq/samp/configuration	mqwebuser.xml 파일에 복사할 수 있는 샘플 구성 파일을 포함하는 디렉토리입니다.

Docker을(를) 사용하여 IBM MQ 구성

Docker를 사용하여 IBM MQ 를 구성하려면 이 정보를 사용하십시오.

이 태스크 정보

Docker를 사용하면 IBM MQ 큐 관리자 또는 IBM MQ 클라이언트 애플리케이션을 그의 모든 종속 항목과 함께 소프트웨어 개발을 위한 표준화된 단위로 패키징할 수 있습니다.

애플리케이션에 대한 변경을 테스트 및 스테이징 시스템으로 빨리 그리고 쉽게 배치할 수 있습니다. 이 기능은 엔터프라이즈에서 연속적인 전달에 대한 중요한 이점일 수 있습니다.

프로시저

- Docker를 사용하여 IBM MQ 를 구성하는 방법에 대한 정보는 다음 하위 주제를 참조하십시오.
  - Linux 561 페이지의 『Linux 시스템에서 Docker 지원』
  - 562 페이지의 『Docker 를 사용하여 고유한 IBM MQ 큐 관리자 이미지 계획』
  - 562 페이지의 『Docker 를 사용하여 샘플 IBM MQ 큐 관리자 이미지 빌드』
  - 566 페이지의 『별도의 컨테이너에서 로컬 바인딩 애플리케이션 실행』

Linux Linux 시스템에서 Docker 지원

Linux 시스템에서 Docker를 사용하는 경우에 고려해야 할 정보입니다.

- Docker 이미지에 사용되는 기본 이미지는 지원되는 Linux 운영 체제를 사용해야 합니다.
- IBM MQ 설치 프로그램을 사용하여 Docker 이미지 내에 제품을 설치해야 합니다.
- 지원되는 패키지 목록은 Linux 시스템에 대한 IBM MQ rpm 컴포넌트의 내용을 참조하십시오.
- V 9.0.4 다음 패키지는 지원되지 않습니다.
  - MQSeriesBCBridge
  - MQSeriesRDQM
- 큐 관리자 데이터 디렉토리(기본적으로 /var/mqm)는 지속 상태를 유지하는 Docker 볼륨에 저장해야 합니다.

  중요사항: 유니온 파일 시스템을 사용할 수 없습니다.

데이터 볼륨으로 호스트 디렉토리를 마운트하거나 데이터 볼륨 컨테이너를 사용해야 합니다. 자세한 정보는 컨테이너에서 데이터 관리를 참조하십시오.

- 컨테이너 내에서 IBM MQ 제어 명령을 실행할 수 있어야 합니다(예: **endmqm**).
- 진단 목적으로 컨테이너 내에 있는 파일과 디렉토리를 가져올 수 있어야 합니다.
- V 9.0.3 로컬에서 애플리케이션을 별도의 컨테이너에서 실행 중인 큐 관리자에 바인드할 수 있도록 네임스페이싱을 사용하여 큐 관리자에 대한 컨테이너의 네임스페이스를 다른 컨테이너와 공유할 수 있습니다. 자세한 정보는 566 페이지의 『별도의 컨테이너에서 로컬 바인딩 애플리케이션 실행』의 내용을 참조하십시오.

Docker 를 사용하여 고유한 IBM MQ 큐 관리자 이미지 계획

Docker를 사용하여 IBM MQ 를 구성하려면 이 정보를 사용하십시오. Docker에서 IBM MQ 큐 관리자를 실행할 때 고려해야 할 몇 가지 요구사항이 있습니다. 샘플 Docker 이미지로 이러한 요구사항을 처리할 수 있지만, 고유한 이미지를 사용하려는 경우에는 이러한 요구사항을 처리하는 방법을 고려해야 합니다.

프로세스 감시

Docker 컨테이너를 실행하면 기본적으로 단일 프로세스(컨테이너 내의 PID 1)가 실행됩니다. 이 프로세스는 나중에 하위 프로세스를 파생시킬 수 있습니다.

기본 프로세스가 종료되면 Docker가 컨테이너를 중지합니다. IBM MQ 큐 관리자는 여러 프로세스가 백그라운드에서 실행 중이어야 합니다.

이 때문에, 큐 관리자가 실행하고 있는 동안에는 기본 프로세스가 활성 상태로 유지하도록 보장해야 합니다. 큐 관리자가 이 프로세스로부터 활성화 상태인지 확인하는 것이 좋습니다(예: 관리 쿼리 수행).

`/var/mqm` 채우기

Docker 컨테이너는 `/var/mqm`을 Docker 볼륨으로 사용하여 구성해야 합니다.

이를 수행할 때, 볼륨의 디렉토리는 컨테이너가 처음 시작할 때 비어 있습니다. 이 디렉토리는 보통 설치 시에 채워지지만, Docker를 사용할 경우 설치와 런타임은 별개의 환경입니다.

V 9.0.3 이를 해결하기 위해 컨테이너가 시작될 때 **crtmqdir** 명령을 사용하여 처음 실행될 때 `/var/mqm` 를 채울 수 있습니다.

Docker 를 사용하여 샘플 IBM MQ 큐 관리자 이미지 빌드

이 정보를 사용하여 Docker 컨테이너에서 IBM MQ 큐 관리자 실행을 위한 샘플 Docker 이미지를 빌드할 수 있습니다.

이 태스크 정보

첫 번째로, Ubuntu Linux 파일 시스템과 IBM MQ의 클린 설치를 포함하는 기본 이미지를 빌드하십시오.

두 번째로, 기본 이미지를 토대로 다른 Docker 이미지 계층을 빌드하십시오. 이 계층에는 기본 사용자 ID 및 비밀번호 보안을 허용하도록 몇 가지 IBM MQ 구성이 추가됩니다.

마지막으로, 이 이미지를 파일 시스템으로 사용하여 Docker 컨테이너를 실행하되, Docker의 호스트 파일 시스템에 있는 컨테이너 특정 Docker 볼륨이 제공하는 `/var/mqm`의 컨텐츠도 함께 실행하십시오.

프로시저

- Docker 컨테이너에서 IBM MQ 큐 관리자 실행을 위한 샘플 Docker 이미지를 빌드하는 방법은 다음 하위 주제를 참조하십시오.
  - 563 페이지의 『샘플 기본 IBM MQ 큐 관리자 이미지 빌드』
  - 564 페이지의 『샘플 구성된 IBM MQ 큐 관리자 이미지 빌드』

샘플 기본 IBM MQ 큐 관리자 이미지 빌드

Docker에서 IBM MQ 를 사용하려면 먼저 깨끗한 IBM MQ 설치를 사용하여 기본 이미지를 빌드해야 합니다. 다음 단계는 GitHub에 호스팅되는 코드를 사용하여 샘플 기본 이미지를 빌드하는 방법을 보여줍니다.

이 태스크 정보

Make를 사용하여 Docker 이미지 빌드
: mq-container GitHub 저장소에 제공된 make 파일을 사용하여 프로덕션 Docker 이미지를 빌드하려면 GitHub에서 Docker 이미지 빌드의 지시사항을 따르십시오.

Docker를 사용하여 수동으로 Docker 이미지 빌드
: Docker를 사용하여 수동으로 이미지를 빌드하려면 다음 단계를 완료하십시오.

프로시저

1. 필수 소프트웨어 패키지를 설치하십시오.

   이들 지시사항은 사용자가 설치해야 하는 일부 Linux 패키지를 사용합니다.

   - Ubuntu에서:

     ```
     sudo apt-get install python git
     ```

   - Red Hat Enterprise Linux에서:

     ```
     sudo yum install python git
     ```

2. `mkdir downloads`명령을 실행하여 `downloads` 디렉토리를 작성하십시오.
3. Passport Advantage®를 사용하여 Linux 이미지에 대해 IBM MQ 서버를 다운로드하십시오.

   자세한 내용은 Electronic Software Download를 사용한 설치를 참조하십시오.

   예를 들어, `WS_MQ_V9.0.5.0_LINUX_ON_X86_64_IM.tar.gz` 파일을 선택한 후 작성한 `downloads` 디렉토리에 이 파일을 배치하십시오.

   참고: CD Ubuntu를 기본 이미지로 사용하려면 Debian 설치를 다운로드해야 합니다.
4. HTTP 또는 FTP 서버에서 사용 가능한 Linux 이미지 (`tar.gz`) 파일에 대해 IBM MQ 서버를 작성하십시오.

   그러면 Docker 이미지 계층의 공간이 절약됩니다. Docker 파일의 각 명령어는 새 이미지 계층을 작성합니다.

   ADD 또는 **COPY** 명령어를 사용한 후 설치하기 위해 **RUN** 명령어를 사용하는 경우, 추가 또는 복사되는 파일이 새 이미지 계층에 커미트됩니다.

   후속 계층의 파일을 삭제하는 경우에도 해당 파일은 이전 계층에 여전히 존재합니다. 이 때문에 단일 **RUN** 명령 안에서 다운로드 및 설치하는 것이 좋은 방법입니다. 이것은 파일이 네트워크에서 사용 가능해야 함을 의미합니다.

   예를 들어, Python을 사용하여 HTTP 서버를 실행함으로써 현재 디렉토리의 모든 파일을 서비스할 수 있습니다.

```
pushd downloads
nohup python -m SimpleHTTPServer 8000 &
popd
```

5. GitHub에서 지원되는 Docker 이미지 빌드를 위한 샘플 파일을 추출하십시오.

   - LTS 다음 명령을 실행하십시오.

```
git clone -b mq-9-lts https://github.com/ibm-messaging/mq-docker mq-docker
```

   - CD 다음 명령을 실행하십시오.

```
git clone https://github.com/ibm-messaging/mq-container mq-container
```

6. 로컬 IP 주소를 식별하십시오.

   주소는 사용자의 로컬 환경에 특정하지만, 다음 명령을 실행하는 경우 사용 가능해야 합니다.

   ```
   ip addr show
   ```

   `localhost`는 작동하지 않음을 주의하십시오.
7. 다음 명령을 실행하여 MQ_URL에 있는 방금 식별한 값에 대한 IP 주소 및 파일 이름을 대체하여 기본 IBM MQ 이미지를 빌드하십시오.

   예를 들면, 다음과 같습니다.

   - LTS

```
sudo docker build --tag mq --build-arg MQ_URL=http://10.0.2.15:8000/
WS_MQ_V9.0.0.0_LINUX_ON_X86_64_IM.tar.gz mq-docker
```

   - CD

```
sudo docker build --tag mq --build-arg MQ_URL=http://10.0.2.15:8000/
WS_MQ_V9.0.0.0_LINUX_ON_X86_64_IM.tar.gz mq-container/Dockerfile-server mq-container
```

결과

이제 IBM MQ 가 설치된 기본 Docker 이미지가 있습니다.

샘플 구성된 IBM MQ 큐 관리자 이미지 빌드

일반적인 기본 IBM MQ Docker 이미지를 빌드한 후에는 보안 액세스를 허용하도록 사용자 고유의 구성을 적용해야 합니다. 이를 위해서는 일반 이미지를 상위로 사용하여 사용자 고유의 Docker 이미지를 작성하십시오. 다음 단계는 최소 보안 구성을 갖고 샘플 이미지를 빌드하는 방법을 보여줍니다.

프로시저

1. 새 디렉토리를 작성하고, 다음 컨텐츠를 갖는 `config.mqsc`라는 파일을 추가하십시오.

```
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)
```

   이전 예제는 단순 사용자 ID 및 비밀번호 인증을 사용한다는 사실에 유의하십시오. 그러나 엔터프라이즈가 요구하는 모든 보안 구성을 적용할 수 있습니다.
2. 다음 컨텐츠를 갖는 `Dockerfile`이라는 파일을 작성하십시오.

```
FROM mq
RUN useradd johndoe -G mqm && \
    echo johndoe:passw0rd | chpasswd
COPY config.mqsc /etc/mqm/
```

   설명:

   - `johndoe`는 추가하려는 사용자 ID입니다.
   - `passw0rd`는 원래 비밀번호입니다.
3. 다음 명령을 사용하여 사용자 정의 Docker 이미지를 빌드하십시오.

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

   여기서 "`.`"은 작성한 두 개의 파일이 포함된 디렉토리입니다.

그런 다음, Docker는 해당 이미지를 사용하여 임시 컨테이너를 작성하고 남아 있는 명령을 실행합니다.

RUN 명령은 비밀번호가 `passw0rd`인 `johndoe`라는 사용자를 추가하고, **COPY** 명령은 상위 이미지에 알려진 특정 위치에 `config.mqsc` 파일을 추가합니다.

4. 새로 사용자 정의된 이미지를 실행하여 방금 작성한 디스크 이미지를 갖는 새 컨테이너를 작성하십시오.

새 이미지 계층은 실행할 특정 명령을 지정하지 않으므로, 상위 이미지에서 상속됩니다. 상위의 시작점(코드는 GitHub에서 사용 가능):

- 큐 관리자 작성
- 큐 관리자 시작
- 기본 리스너 작성
- 그런 다음 `/etc/mqm/config.mqsc.`에서 임의의 MQSC 명령 실행

다음 명령을 실행하여 새로 사용자 정의된 이미지를 실행하십시오.

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

여기서,

첫 번째 env 매개변수
환경 변수를 컨테이너로 전달하는데, 이것은 IBM IBM WebSphere MQ에 대한 라이센스의 허용을 수신 확인합니다. 또한 해당 라이센스를 보기 위해 LICENSE 변수를 view로 설정할 수도 있습니다.

IBM MQ 라이센스에 대한 자세한 정보는 IBM MQ 라이센스 정보 를 참조하십시오.

두 번째 env 매개변수
사용 중인 큐 관리자 이름을 설정합니다.

Volume 매개변수
컨테이너에게 MQ가 `/var/mqm`에 쓰는 것은 무엇이든 실제로는 호스트의 `/var/example`에 기록되도록 지시합니다.

이 옵션은 나중에 컨테이너를 쉽게 삭제하고 모든 지속 데이터를 계속 보존할 수 있다는 것을 의미합니다. 이 옵션은 또한 로그 파일을 보기가 더 쉽게 합니다.

Publish 매개변수
호스트 시스템의 포트를 컨테이너의 포트에 맵핑합니다. 컨테이너는 기본적으로 자체 내부 IP 주소를 갖고 실행하며, 이것은 노출시키려는 임의의 포트를 특정하게 맵핑해야 함을 의미합니다.

이 예에서는 호스트의 포트 1414를 컨테이너의 포트 1414에 맵핑함을 의미합니다.

Detach 매개변수
컨테이너를 백그라운드에서 실행합니다.

결과

구성된 Docker 이미지를 빌드했으므로 Docker **ps** 명령을 사용하여 실행 중인 컨테이너를 볼 수 있습니다. docker **top** 명령을 사용하여 컨테이너에서 실행 중인 IBM MQ 프로세스를 볼 수 있습니다.

주의: Docker **ps** 명령을 사용할 때 컨테이너가 표시되지 않는 경우 컨테이너가 실패했을 수 있습니다. 명령 Docker **ps -a**를 사용하여 실패한 컨테이너를 볼 수 있습니다.

컨테이너 ID는 docker **ps -a** 명령을 사용하여 표시되며 docker **run** 명령을 실행할 때도 인쇄되었습니다.

docker **logs ${CONTAINER_ID}** 명령을 사용하여 컨테이너의 로그를 볼 수 있습니다.

일반적인 문제점은 **mqconfig** 이 Docker 호스트의 특정 커널 설정이 올바르지 않음을 표시하는 것입니다. 커널 설정은 Docker 호스트와 컨테이너 사이에서 공유되며, 올바로 설정되어야 합니다(UNIX and Linux 시스템의 하드웨어 및 소프트웨어 요구사항 참조).

예를 들어, **`sysctl fs.file-max=524288`**명령을 사용하여 최대 열린 파일 수를 설정할 수 있습니다.

V 9.0.3 별도의 컨테이너에서 로컬 바인딩 애플리케이션 실행

Docker의 컨테이너 간에 공유되는 프로세스 네임스페이스가 추가되어 IBM MQ에 대한 로컬 바인딩 연결이 필요한 애플리케이션을 이제 별도의 컨테이너에 있는 IBM MQ 큐 관리자에서 실행할 수 있습니다. 이 기능은 IBM MQ 9.0.3 이상 큐 관리자에서 지원됩니다.

이 태스크 정보

다음 제한사항을 준수해야 합니다.

- Docker 버전 1.12 이상을 사용해야 합니다.
- `--pid` 인수를 사용하여 컨테이너 PID 네임스페이스를 공유해야 합니다.
- `--ipc` 인수를 사용하여 컨테이너 IPC 네임스페이스를 공유해야 합니다.
- 다음 중 하나를 수행해야 합니다.
  1. `--uts` 인수를 사용하여 컨테이너 UTS 네임스페이스를 호스트와 공유하십시오.
  2. `-h` 또는 `--hostname` 인수를 사용하여 컨테이너가 동일한 호스트 이름을 사용하는지 확인하십시오.
- `/var/mqm` 디렉토리 아래의 모든 컨테이너에서 사용할 수 있는 볼륨에 IBM MQ 데이터 디렉토리를 마운트해야 합니다.

Docker 1.12 이상이 이미 설치되어 있는 Linux 시스템에서 다음 단계를 완료하여 이 기능을 시도할 수 있습니다.

다음 예에서는 샘플 IBM MQ Docker 컨테이너 이미지를 사용합니다. Github에서 이 이미지에 대한 세부사항을 찾을 수 있습니다.

프로시저

1. 다음 명령을 발행하여 볼륨으로 사용할 임시 디렉토리를 작성하십시오.

```
mkdir /tmp/dockerVolume
```

2. 다음 명령을 발행하여 `sharedNamespace`라는 이름으로 컨테이너에 큐 관리자(`QM1`)를 작성하십시오.

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

3. `secondaryContainer`라는 두 번째 컨테이너를 시작하고, `ibmcom/mq`를 기반으로 하지만 다음 명령을 실행하여 큐 관리자를 작성하지 마십시오.

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

4. 두 번째 컨테이너에서 **dspmq** 명령을 다음과 같이 실행하여 두 큐 관리자의 상태를 확인하십시오.

```
docker exec secondaryContainer dspmq
```

5. 다음 명령을 실행하여 다른 컨테이너에서 실행되는 큐 관리자에 대해 MQSC 명령을 처리하십시오.

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

결과

이제 별도의 컨테이너에서 로컬 애플리케이션이 실행 중이며 이제 두 번째 컨테이너에서 **dspmq**, **amqsput**, **amqsget** 및 **runmqsc**와 같은 명령을 QM1 큐 관리자에 대한 로컬 바인딩으로 성공적으로 실행할 수 있습니다.

예상한 결과를 얻지 못한 경우 자세한 정보는 567 페이지의 『네임스페이스 애플리케이션 문제점 해결』의 내용을 참조하십시오.

V 9.0.3 네임스페이스 애플리케이션 문제점 해결

공유 네임스페이스를 사용할 경우 모든 네임스페이스(IPC, PID, UTS/호스트 이름)와 마운트된 볼륨을 공유해야 합니다. 그렇지 않으면 애플리케이션이 작동하지 않습니다.

준수해야 하는 제한사항 목록은 566 페이지의 『별도의 컨테이너에서 로컬 바인딩 애플리케이션 실행』의 내용을 참조하십시오.

애플리케이션이 나열된 제한사항을 모두 충족하지 못할 경우, 컨테이너는 시작되지만 예상 기능이 작동하지 않는 문제점이 발견될 수 있습니다.

다음 목록에는 몇 가지 공통 원인 및 제한사항 중 하나가 충족되지 않았는지 확인할 수 있는 동작이 설명되어 있습니다.

- 네임스페이스(UTS/PID/IPC)를 공유하거나 컨테이너의 호스트 이름을 동일하게 설정하지 못했지만, 볼륨은 마운트한 경우 컨테이너가 큐 관리자를 표시할 수는 있지만 큐 관리자와 상호작용할 수는 없습니다.
  - **dspmq** 명령의 경우 다음이 표시됩니다.

```
docker exec container dspmq

QMNAME(QM1)                              STATUS(Status not available)
```

  - **runmqsc** 명령 또는 큐 관리자에 연결하려고 시도하는 다른 명령의 경우 AMQ8146 오류 메시지가 수신될 수 있습니다.

```
docker exec -it container runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2023.
Starting MQSC for queue manager QM1.
AMQ8146: IBM MQ queue manager not available
```

- 필요한 네임스페이스를 모두 공유했지만 공유 볼륨을 `/var/mqm` 디렉토리에 마운트하지 않았으며 올바른 IBM MQ 데이터 경로를 사용하는 경우 AMQ8146 오류 메시지도 수신됩니다.

  그러나 **dspmq**는 큐 관리자를 표시할 수 없어 빈 응답을 대신 리턴합니다.

```
docker exec container dspmq
```

- 필요한 네임스페이스를 모두 공유했지만 공유 볼륨을 `/var/mqm` 디렉토리에 마운트하지 않았으며 올바른 IBM MQ 데이터 경로를 사용하지 않거나 IBM MQ 데이터 경로가 없는 경우, 데이터 경로가 IBM MQ 설치의 핵심 컴포넌트이므로 다양한 오류가 표시됩니다. 데이터 경로가 없으면 IBM MQ를 작동할 수 없습니다.

  다음 명령을 실행한 후 이 예에 표시된 것과 유사한 응답이 표시될 경우, 디렉토리를 마운트했거나 IBM MQ 데이터 디렉토리를 작성했는지 확인해야 합니다.

```
docker exec container dspmq
'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 dspmqver
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 dltmqm QM1
AMQ7002: An error occurred manipulating a file.
```

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

V 9.0.2 Windows Linux IBM Cloud 에서 IBM Cloud Product Insights 서비스와 함께 사용하도록 IBM MQ 구성

IBM Cloud® Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

V 9.0.2 Windows Linux IBM Cloud (formerly Bluemix)에서 IBM Cloud Product Insights 서비스 인스턴스 작성

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

V 9.0.2 Windows Linux IBM Cloud (formerly Bluemix)에서 IBM Cloud Product Insights 서비스 인스턴스와 함께 사용하도록 큐 관리자 구성

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

V 9.0.4 HTTP 프록시를 통해 IBM Cloud에서 IBM Cloud Product Insights에 연결

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

V 9.0.4 Product Insights에 대한 연결 문제점 해결

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

V 9.0.2 Linux Salesforce 푸시 토픽 및 플랫폼 이벤트와 함께 사용하도록 IBM MQ 구성

IBM MQ Bridge to Salesforce를 구성한 후 실행하여 Salesforce 및 IBM MQ 네트워크에 대한 연결 및 보안을 설정하려면 이 정보를 사용하십시오.

시작하기 전에

- IBM MQ Bridge to Salesforce는 Linux Linux for System x(64비트)에서 사용할 수 있습니다. IBM WebSphere MQ 6.0 이하에서 실행 중인 큐 관리자에 대한 연결에는 브릿지가 지원되지 않습니다.
- **MQSeriesSFBridge** 패키지를 설치하십시오. 자세한 정보는 Linux에 IBM MQ 서버 설치를 참조하십시오.

이 태스크 정보

Salesforce는 클라우드 기반 고객 관계 관리 플랫폼입니다. If you are using Salesforce to manage customer data and interactions, at IBM MQ 9.0.2 you can use the IBM MQ Bridge to Salesforce to subscribe to Salesforce push topics and platform events that can then be published to your IBM MQ queue manager. 해당 큐 관리자에 연결되는 애플리케이션은 푸시 토픽 및 플랫폼 이벤트 데이터를 유용하게 이용할 수 있습니다.

V 9.0.4 IBM MQ 9.0.4에서는 브릿지를 사용하여 플랫폼 이벤트에 대한 이벤트 메시지를 Salesforce에 작성할 수도 있습니다.

IBM MQ Bridge to Salesforce에 대한 개요는 그림 1의 다이어그램을 참조하십시오.

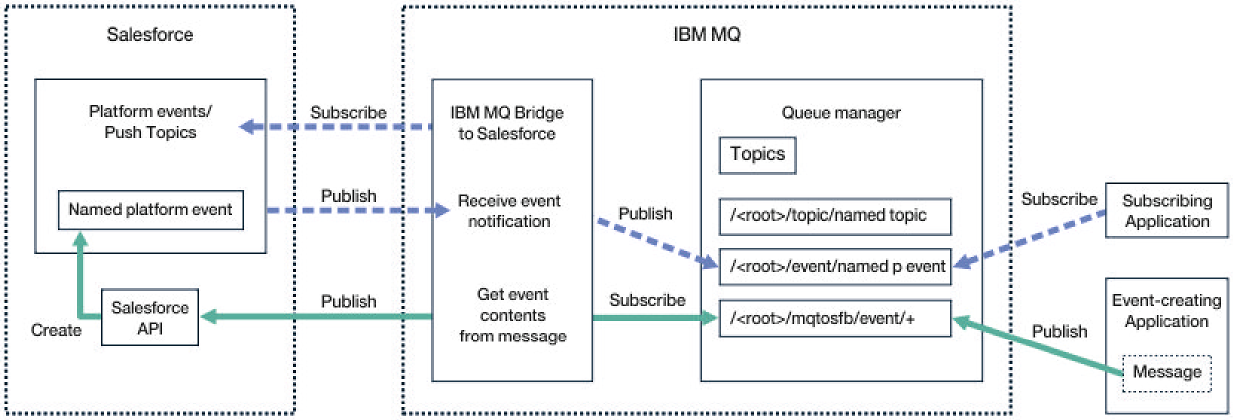


그림 *97. IBM MQ Bridge to Salesforce*

푸시 토픽은 `Force.com` Streaming API를 사용하여 Salesforce의 레코드 변경 사항에 대한 알림을 수신하도록 정의하는 조회입니다. 푸시 토픽 구성 및 스트리밍 API 사용 방법에 대한 자세한 정보는 스트리밍 API 소개 및 PushTopics에 대한 작업을 참조하십시오.

플랫폼 이벤트는 `Force.com` 플랫폼이 생성하거나 이용하는 이벤트 데이터를 판별하기 위해 정의할 수 있는 사용자 정의 가능 이벤트 메시지입니다. Salesforce 이벤트 간 차이와 플랫폼 이벤트에 대한 자세한 정보는 엔터프라이즈 메시징 플랫폼 이벤트 및 Salesforce 이벤트 간 차이를 참조하십시오.

- 푸시 토픽 및 플랫폼 이벤트를 구독하는 구성을 작성하려면 570 페이지의 『IBM MQ Bridge to Salesforce』의 내용을 참조하십시오.
- V 9.0.4 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성하는 구성을 작성하려면 574 페이지의 『Salesforce 플랫폼 이벤트에 대한 이벤트 메시지 작성』의 내용을 참조하십시오.

IBM MQ Console 를 통해 브릿지에서 **amqsrua** 명령과 함께 **-p** 매개변수를 사용하여 데이터를 모니터할 수 있습니다. 전체 브릿지 상태에 대해 하나의 데이터 세트가 발행됩니다.

- 하나의 간격에서 처리되는 총 푸시 토픽 메시지 수(`STATUS/PUSHTOPIC` 트리 아래에 있음)
- 이 간격에 표시되는 푸시 토픽의 수
- 하나의 간격에서 처리되는 총 플랫폼 이벤트 수(`STATUS/PLATFORM` 트리 아래에 있음)
- 이 간격에 표시되는 플랫폼 이벤트의 수
- V 9.0.4 하나의 간격에서 처리되는 IBM MQ 작성 총 플랫폼 이벤트 수(`STATUS/MQPE` 트리 아래에 있음)
- V 9.0.4 이 간격에 표시되는 고유한 IBM MQ 작성 플랫폼 이벤트 수
- V 9.0.4 이 간격에 표시되는 IBM MQ 작성 플랫폼 이벤트의 발행 실패 횟수

구성된 각 Salesforce 토픽에 대해 추가적인 메시지가 발행됩니다. IBM MQ 주제에서는 오브젝트 이름에 전체 Salesforce 토픽 이름과 `/event` 또는 `/topic`을(를) 사용합니다.

- 하나의 간격에서 처리되는 메시지의 수

브릿지 데이터를 모니터하도록 IBM MQ Console을 구성하려면 다음 태스크인 IBM MQ Bridge to Salesforce 구성의 9단계와 10단계를 참조하십시오. **amqsrua** 명령 사용에 대한 정보는 IBM MQ Bridge to Salesforce모니터링을 참조하십시오.

이 태스크에 있는 단계를 수행하여 IBM MQ Bridge to Salesforce를 구성하고 실행하십시오.

프로시저

1. IBM MQ Bridge to Salesforce를 구성하십시오.

2. V 9.0.4
   Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성하십시오.
3. IBM MQ Bridge to Salesforce를 실행하십시오.

관련 정보

runmqsfb(IBM MQ Bridge to Salesforce 실행)

IBM MQ Bridge to Salesforce 추적

V 9.0.2 Linux IBM MQ Bridge to Salesforce

IBM MQ를 구성하고 IBM MQ Bridge to Salesforce 매개변수를 입력하여 구성 파일을 작성하고 Salesforce 푸시 토픽 및 플랫폼 이벤트를 IBM MQ 큐 관리자에 연결할 수 있습니다.

시작하기 전에

- x86-64 Linux 플랫폼의 IBM MQ 설치에 **MQSeriesSFBridge** 패키지를 설치했습니다.

이 태스크 정보

이 태스크는 Salesforce 푸시 토픽 및 플랫폼 이벤트를 구독할 수 있도록 IBM MQ Bridge to Salesforce 구성 파일을 작성하고 Salesforce 및 IBM MQ에 성공적으로 연결하는 데 필요한 최소 설정을 안내합니다. 모든 매개변수에 대한 의미 및 옵션과 관련된 자세한 정보는 runmqsfb 명령을 참조하십시오. 자체 보안 요구사항을 고려하고 배치에 적합하게 매개변수를 사용자 정의해야 합니다.

V 9.0.4 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성하는 구성을 작성하려면 574 페이지의 『Salesforce 플랫폼 이벤트에 대한 이벤트 메시지 작성』의 내용을 참조하십시오.

Salesforce 푸시 토픽 및 플랫폼 이벤트 구독

IBM MQ Bridge to Salesforce는 Salesforce와 IBM MQ 모두에 대한 연결을 설정하는 경우 Salesforce 푸시 토픽 및 플랫폼 이벤트에 대한 구독을 작성합니다. 브릿지가 구독하려는 푸시 토픽 또는 플랫폼 이벤트 이름은 연결이 작성되기 전에 명령행에서 추가되거나 구성 파일에 포함되어야 합니다.

구성 속성 중 하나는 IBM MQ 토픽 트리의 루트이며 이벤트는 이 루트 아래에 발행됩니다. 브릿지는 이 루트에 액세스하고 전체 Salesforce 토픽 이름을 추가합니다(예: `/MQ/SF/ROOT/topic/EscalatedCases`). IBM MQ에 연결되는 모니터링 토픽 및 애플리케이션은 `/topic/EscalatedCases` 아래에서 푸시 토픽을 검색하고 `/event/NewCustomer__e` 아래에서 플랫폼 이벤트를 검색합니다.

발행된 메시지에는 요청된 데이터 필드가 포함된 데이터 구조 및 제어 정보가 포함되어 있습니다. 푸시 토픽의 경우 데이터 구조는 **sobject**이고 플랫폼 이벤트의 경우 구조는 **payload**입니다. 브릿지는 토픽 또는 이벤트가 Salesforce에 정의되어 있지 않은 경우 토픽 또는 이벤트를 구독할 수 없습니다. 브릿지가 토픽 구독을 시도할 때 오류가 발생하면 브릿지가 중지됩니다.

토픽 오브젝트는 IBM MQ에서 정의되지 않아도 되지만 트리에서 가장 가까운 상위 요소를 기반으로 적합한 권한이 존재해야 합니다. 재발행된 메시지에는 기본적으로 원래 메시지의 관련 데이터 구조만 포함되어 있습니다. 제어 정보는 제거됩니다. 플랫폼 이벤트의 경우 발행은 페이로드(payload) 구조를 가지고 있습니다. 구성 매개변수의 **작동 동작** 세트에 있는 **`Publish control data with the payload`** 구성 옵션을 사용하면 제어 데이터를 포함하여 전체 메시지를 다시 공개할 수 있습니다. 자세한 정보는 구성 매개변수를 참조하십시오.

각각의 푸시 토픽 및 플랫폼 이벤트에는 Salesforce로부터의 발행에 대해 연관된 *ReplayID*가 있습니다. *ReplayID*를 사용하여 서버에 대한 연결이 작성될 때 발행의 시작점을 요청할 수 있습니다. 최근 푸시 토픽 및 플랫폼 이벤트가 생성될 때 시작된 경우에도 Salesforce는 최대 24시간 동안 히스토리를 유지하고 브릿지가 최근 푸시 토픽 및 플랫폼 이벤트를 누락하지 않도록 허용합니다. 브릿지는 두 가지 서비스 품질(QoS) 모드를 지원합니다.

최대 한 번
: 브릿지는 재시작을 위해 *ReplayId*를 사용하지 않습니다. 브릿지 재시작 후에는 새로 생성된 푸시 토픽 및 플랫폼 이벤트만 처리됩니다. 애플리케이션은 누락된 발행물을 처리할 준비가 되어 있어야 합니다. 브릿지에서 아직 *ReplayId*를 추적하여 큐에 기록하므로 다른 서비스 품질(QoS)로 브릿지를 다시 시작하여 현재 상태를 알아볼 수 있습니다.

최소 한 번
: 브릿지에서 *ReplayId*를 추적하고 큐에 기록합니다. 브릿지 재시작 시 지속된 *ReplayId*를 사용하여 서버로부터의 발행물의 시작점을 요청합니다. 차이가 24시간 이하인 경우에는 이전 발행물이 전송됩니다. 토픽에 대한 *ReplayId*가 모든 메시지에 기록되지는 않습니다. 이는 일정한 간격으로 또는 브릿지 종료 시 지속 메시지에 기록됩니다. 애플리케이션은 중복 발행물을 볼 준비가 되어 있어야 합니다.

*ReplayId*는 새로 정의된 큐에 메시지로 기록됩니다. 브릿지가 시작되기 전에 이 큐 **SYSTEM.SALESFORCE.SYNCQ**를 정의해야 합니다. **SYSTEM.SALESFORCE.SYNCQ**가 존재하지 않는 경우 브릿지는 서비스 품질(QoS) 모드에 관계없이 계속되지 않습니다. 관련 속성을 사용하여 큐를 작성하기 위해 MQSC 스크립트가 제공됩니다. 브릿지 프로그램의 한 인스턴스만 **SYSTEM.SALESFORCE.SYNCQ** 큐를 업데이트할 수 있도록 하기 위해 DEFSOPT(EXCL) NOSHARE 옵션을 사용하여 큐를 구성해야 합니다.

V 9.0.4 플랫폼 이벤트에 대한 이벤트 메시지를 작성하는 구성을 작성하려면 574 페이지의 『Salesforce 플랫폼 이벤트에 대한 이벤트 메시지 작성』의 내용을 참조하십시오.

프로시저

1. 큐 관리자를 작성하고 시작하십시오.
   a) 큐 관리자를 작성하십시오(예: SQM1).
   ```
   crtmqm SQM1
   ```
   b) 큐 관리자를 시작하십시오.
   ```
   strmqm SQM1
   ```
2. **참고:** 기존 로그인 및 보안 Salesforce 신임 정보 및 자체 서명 인증서를 사용하려면 571 페이지의 『3』 단계로 건너뛰십시오.

   옵션: Salesforce 계정에 대한 보안 토큰을 작성하십시오.
   a) Salesforce 계정에 로그인하십시오.
   b) 도움말 기사 Salesforce 도움말: 보안 토큰 재설정에 있는 단계를 수행하여 보안 토큰을 작성하거나 재설정하십시오.
3. Salesforce에 CA 서명 보안 인증서를 작성하십시오.
   a) **Force.com 홈**의 **관리** 메뉴에서 **보안 제어**를 선택한 후 **인증서 및 키 관리**를 선택하십시오.
      인증서 및 키 관리 페이지가 열립니다.
   b) **CA 서명 인증서 작성**을 누르십시오.
      인증서 페이지가 열립니다.
   c) **레이블** 필드에 인증서의 이름을 입력하고 Tab을 누른 후 **저장**을 클릭하십시오.
      인증서 및 키 세부사항 정보가 표시됩니다.
   d) **목록으로 돌아가기: 인증서 및 키**를 클릭하십시오.
   e) **키 저장소에 내보내기**를 클릭하십시오.
   f) 키 저장소의 비밀번호를 입력한 후 **내보내기**를 클릭하십시오.
   g) 내보낸 키 저장소를 로컬 파일 시스템에 저장하십시오.
4. IBM Key Management GUI를 사용하여 Salesforce에서 내보낸 키 저장소를 열고 서명자 인증서를 채우십시오.
   a) **strmqikm** 명령을 실행하여 IBM 키 관리 GUI를 여십시오.
      자세한 정보는 디지털 인증서를 관리하기 위해 runmqckm, runmqakm 및 strmqikm 사용을 참조하십시오.
   b) **키 데이터베이스 파일 열기**를 클릭한 후 Salesforce 키 저장소의 위치로 이동하십시오.
   c) **열기**를 클릭하고 **키 데이터베이스 유형** 옵션에서 **JKS**를 선택했는지 확인한 후 **확인**을 클릭하십시오.
   d) 3f단계에서 키 저장소에 대해 작성한 비밀번호를 입력한 후 **확인**을 클릭하십시오.
   e) **키 데이터베이스 컨텐츠** 옵션에서 **서명자 인증서**를 선택하십시오.

f) **채우기**를 클릭하십시오.

g) **CA 인증서 추가** 목록에서 **Verisign Inc.** 선택란을 선택한 후 **확인**을 클릭하십시오.

5. 옵션: Salesforce 계정에서 IBM MQ Bridge to Salesforce에 대한 앱 연결을 작성하여 OAuth 컨슈머 키 및 시크릿을 생성하십시오.

   프로덕션 환경에서 IBM MQ Bridge to Salesforce를 사용하는 경우 **컨슈머 키** 및 **컨슈머 시크릿** 코드가 필요합니다.

   a) **작성**을 선택한 후 **Force.com 홈** 페이지의 **빌드** 메뉴에서 **앱**을 선택하십시오.

      앱 페이지가 열립니다.

   b) **연결된 앱** 섹션에서 **새로 작성**을 클릭하십시오.

      새 연결된 앱 페이지가 열립니다.

   c) **연결된 앱 이름**에 IBM MQ Bridge to Salesforce의 이름(예: **MQBridgeToSalesforce**)을 입력하십시오.

   d) **API 이름**을 입력하십시오.

      다음 필드로 탭하여 이동하는 경우 **연결된 앱 이름**이 **API 이름** 이름 필드에 복사됩니다.

   e) **담당자 이메일**을 입력하십시오.

   f) **API(OAuth 설정 사용)** 섹션에서 **OAuth 설정 사용** 옵션을 선택하십시오.

      해당 섹션의 추가적인 옵션이 제공됩니다.

   g) **콜백 URL**(예: `https://www.ibm.com`)을 추가하십시오.

   h) **선택된 OAuth 범위** 하위 섹션의 **사용 가능한 OAuth 범위** 목록에서 **전체 액세스(전체)** 옵션을 선택한 후 **추가**를 클릭하여 **선택된 OAuth 범위** 목록에 대한 전체 액세스를 추가하십시오.

   i) **저장**을 클릭하십시오.

   j) **계속**을 클릭하십시오.

   k) **컨슈머 키** 및 **컨슈머 시크릿** 코드를 기록해 두십시오.

6. 큐 관리자에서 필요한 동기화 큐를 작성하십시오.

```
cat /opt/mqm/mqsf/samp/mqsfbSyncQ.mqsc | runmqsc SQM1
```

   동기화 큐는 애플리케이션 또는 큐 관리자 재시작에서 이벤트 상태를 유지보수합니다. 큐에는 단일 메시지만 있을 것으로 예상되므로 큐 용량은 적을 수 있습니다. 이 큐에 대해 브릿지의 한 인스턴스만 한 번에 실행될 수 있으므로 독점 액세스를 위해 기본 옵션이 설정됩니다.

7. IBM MQ, Salesforce 및 IBM MQ Bridge to Salesforce 작동에 대해 연결 및 보안 매개변수를 사용하여 구성 파일을 작성하십시오.

```
runmqsfb -o new_config.cfg
```

   기존 값이 대괄호 안에 표시됩니다. `Enter`를 눌러 기존 값을 승인하고 `Space`를 누른 후 `Enter`를 눌러 값 및 유형을 지운 다음 `Enter`를 눌러 새 값을 추가하십시오.

   a) 큐 관리자 SQM1에 대한 연결을 위한 값을 입력하십시오.

      연결에 필요한 최소값은 큐 관리자 이름, IBM MQ 기본 토픽 루트 및 채널 이름입니다.

```
Connection to Queue Manager
---------------------------
Queue Manager or JNDI CF    : []SQM1
MQ Base Topic               : []/sf
MQ Channel                  : []A channel you have defined or for example
SYSTEM.DEF.SVRCONN
MQ Conname                  : []
V 9.0.4 MQ Publication Error Queue : [SYSTEM.SALESFORCE.ERRORQ]
MQ CCDT URL                 : []
JNDI implementation class   : [com.sun.jndi.fscontext.RefFSContextFactory]
JNDI provider URL           : []
MQ Userid                   : []
MQ Password                 : []
```

      참고: 로컬로 연결하는 경우에는 채널 이름이 필요하지 않습니다. 큐 관리자 이름 및 기본 토픽은 나중에 브릿지를 실행할 때 명령행에 포함될 수 있으므로 구성 파일에 제공하지 않아도 됩니다.

b) Salesforce에 대한 연결을 위한 값을 입력하십시오.

연결에 필요한 최소값은 Salesforce 사용자 ID, 비밀번호, 보안 토큰 및 로그인 엔드 포인트입니다. 프로덕션 환경에서는 OAuth 보안에 대한 컨슈머 키 및 시크릿을 추가할 수 있습니다.

```
Connection to Salesforce
------------------------
Salesforce Userid (reqd)   : []salesforce_login_email
Salesforce Password (reqd) : []salesforce_login_password
Security Token (reqd)      : []Security_Token
Login Endpoint             : [https://login.salesforce.com]
Consumer ID                : []
Consumer Secret Key        : []
```

c) TLS 연결의 인증서 저장소를 위한 값을 입력하십시오.

TLS 연결에 필요한 최소값은 TLS 인증서 및 키 저장소 비밀번호의 경로입니다. 신뢰할 수 있는 저장소 경로 또는 비밀번호가 제공되지 않는 경우에는 키 저장소 및 비밀번호 매개변수가 신뢰할 수 있는 저장소 및 비밀번호에 사용됩니다. IBM MQ 큐 관리자 연결에 TLS를 사용하는 경우 동일한 키 저장소를 사용할 수 있습니다.

```
Certificate stores for TLS connections
--------------------------------------
Personal keystore for TLS certificates : []path_to_keystore, for example: /var/mqm/qmgrs/
SQM1/ssl/key.jks
Keystore password            : []keystore_password
Trusted store for signer certificates : []
Trusted store password       : []
Use TLS for MQ connection    : [N]
```

d) 값을 입력하여 IBM MQ Bridge to Salesforce의 작동을 구성하십시오.

이 값은 변경하거나 제공하지 않아도 되지만 푸시 토픽 또는 플랫폼 이벤트 이름을 알고 있는 경우 이 이름을 여기에 추가하십시오. 이 이름은 나중에 브릿지를 실행할 준비가 되면 명령행에서 추가할 수도 있습니다. 구성 파일 또는 명령행에서 로그 파일을 지정해야 합니다.

```
Behaviour of bridge program
---------------------------
PushTopic Names          : []
Platform Event Names     : []
MQ Monitoring Frequency  : [30]
At-least-once delivery? (Y/N) : [Y]
V 9.0.4 Subscribe to MQ publications for platform events? (Y/N) : [N]
Publish control data with the payload? (Y/N) : [N]
Delay before starting to process events : [0]
Runtime logfile for copy of stdout/stderr : []
```

8. 옵션: IBM MQ 서비스를 작성하여 프로그램 실행을 제어하십시오. 새로 작성된 구성 파일을 가리키도록 샘플 `mqsfbService.mqsc` 파일을 편집하고 명령 매개변수에 대한 기타 변경사항을 작성하십시오.

```
cat modified mqsfbService.mqsc | runmqsc SQM1
```

9. V 9.0.1

옵션: IBM MQ Console 시작하기에 있는 지시사항을 수행하여 IBM MQ Console을 설정하십시오.

10. **참고:** MQ Console에서 브릿지에 대한 데이터를 보려면 먼저 브릿지가 시작될 때 Salesforce 및 IBM MQ에 연결되도록 브릿지를 적어도 한 번 실행해야 합니다. 브릿지 시작 시 브릿지에 대한 메타 토픽이 발행됩니다.

옵션: IBM MQ Console 인스턴스에서 위젯을 추가하고 구성하여 Salesforce 데이터를 보십시오.

a) **위젯 추가**를 클릭하십시오.

새 위젯이 열립니다.

b) **차트**를 선택하십시오.

c) 새 위젯의 제목 표시줄에서 **위젯 구성** 아이콘을 클릭하십시오.

d) 옵션: **위젯 제목**을 입력하십시오.

e) **모니터할 자원, 소스** 드롭 다운 메뉴에서 **Salesforce 브릿지**를 선택하십시오.

f) **저장**을 클릭하십시오.

결과

IBM MQ Bridge to Salesforce가 Salesforce 푸시 토픽 및 플랫폼 이벤트를 구독하고 IBM MQ 네트워크에 공개하는 데 사용하는 구성 파일을 작성했습니다.

다음에 수행할 작업

579 페이지의 『IBM MQ Bridge to Salesforce 실행』에 대한 단계를 수행하십시오.

관련 정보

runmqsfb(IBM MQ Bridge to Salesforce 실행)

IBM MQ Bridge to Salesforce 추적

IBM MQ Bridge to Salesforce 모니터링

V 9.0.4 Linux Salesforce 플랫폼 이벤트에 대한 이벤트 메시지 작성

IBM MQ를 구성하고, `IBM MQ Bridge to Salesforce` 매개변수를 입력하여 구성 파일을 작성하며, 브릿지를 사용하여 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성할 수 있습니다.

시작하기 전에

- x86-64 Linux 플랫폼의 IBM MQ 설치에 **MQSeriesSFBridge** 패키지를 설치했습니다.

이 태스크 정보

이 태스크는 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성할 수 있도록 IBM MQ Bridge to Salesforce 구성 파일을 작성하고 Salesforce 및 IBM MQ에 성공적으로 연결하는 데 필요한 최소 설정을 안내합니다. 모든 매개변수에 대한 의미 및 옵션과 관련된 자세한 정보는 runmqsfb 명령을 참조하십시오. 자체 보안 요구사항을 고려하고 배치에 적합하게 매개변수를 사용자 정의해야 합니다.

푸시 토픽 및 플랫폼 이벤트를 구독하는 구성을 작성하려면 570 페이지의 『IBM MQ Bridge to Salesforce』의 내용을 참조하십시오.

Salesforce 플랫폼 이벤트에 대한 이벤트 메시지 작성

IBM MQ 9.0.4에서는 IBM MQ 애플리케이션을 사용하여 큐 관리자 토픽/*root*/mqtosfb/event/+에 넣을 메시지를 작성할 수 있습니다. 브릿지는 토픽을 구독하고, 메시지에서 컨텐츠를 가져오며, 이를 사용하여 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 발행합니다. 플랫폼 이벤트에 대한 자세한 정보는 Salesforce 개발자 문서에서 플랫폼 이벤트와 함께 사용자 정의 알림 제공을 참조하십시오.

브릿지가 이벤트 메시지를 작성할 수 있도록 하려면 푸시 토픽 및 플랫폼 이벤트를 구독하는 데 사용된 IBM MQ 9.0.2의 속성 이외에도 두 개의 속성을 추가로 제공해야 합니다.

- **큐 관리자에 대한 연결**의 브릿지 구성 속성에 **MQ Publication Error Queue**의 이름을 작성하고 추가하십시오.
- **브리지 프로그램의 작동**을 정의하기 위한 브릿지 구성 속성에서 **Subscribe to MQ publications for platform events** 옵션을 *Y*로 설정하십시오.

브릿지를 사용하여 플랫폼 이벤트에 대한 이벤트 메시지를 작성하기 전에 Salesforce에 플랫폼 이벤트를 작성하고 컨텐츠 필드를 정의해야 합니다. 플랫폼 이벤트 이름과 컨텐츠에 따라 브릿지에서 처리하는 IBM MQ 메시지의 형식화 방법이 결정됩니다. 예를 들어, Salesforce 플랫폼 이벤트 **Object name**이(가) *MQPlatformEvent1*이고 두 개의 사용자 정의 필드가 **API name** *MyText__c* 및 *Name__c*가 있는 텍스트 필드인 경우 /*root*/mqtosfb/event/MQPlatformEvent1__e 주제에 게시되는 IBM MQ 메시지는 다음과 같이 올바른 형식의 JSON이어야 합니다.

```
{ "MyText__c" : "Some text here", "Name__c" : "Bob Smith" }
```

IBM MQ Bridge to Salesforce에서 MQFMT_STRING 형식화된 메시지 본문으로 인식할 수 있도록 메시지를 형식화해야 합니다.

576 페이지의 『7』 단계를 참조하여 Salesforce에 플랫폼 이벤트를 작성하거나, 이벤트 메시지를 작성할 플랫폼 이벤트가 이미 있는 경우에는 이 단계를 건너뛰십시오. Salesforce 플랫폼 이벤트에 설정된 필드와 일치하도

록 IBM MQ 메시지를 형식화해야 합니다. Salesforce 플랫폼 이벤트 내의 필드는 선택사항 또는 필수로 지정할 수 있습니다. 자세한 정보는 Salesforce 개발자 문서에서 플랫폼 이벤트 필드를 참조하십시오.

브릿지가 실행 중인 경우 지정된 IBM MQ 토픽을 구독합니다.

- 브릿지 구성에 **At-most-once** 서비스 품질(QoS)을 지정한 경우 브릿지가 작성하는 구독은 지속 불가능 구독입니다. 브릿지가 실행되지 않는 동안 IBM MQ 애플리케이션에서 발행한 항목이 처리되지 않습니다.
- 브릿지 구성에 **At-least-once** 서비스 품질(QoS)을 지정한 경우 브릿지가 작성하는 구독은 지속 가능 구독입니다. 즉, 브릿지가 실행되지 않는 동안 IBM MQ 애플리케이션에서 발행한 항목을 브릿지가 처리할 수 있습니다. 지속 가능 구독에는 알려진 구독 및 클라이언트 ID가 필요합니다. 브릿지는 구독 이름으로 *D_SUB_RUNMQSFB*를 사용하고, 클라이언트 ID로 *runmqsfb_1*을 사용합니다.

브릿지가 이벤트 메시지를 작성하는 데 사용되지 않고 Salesforce 푸시 토픽 및 플랫폼 이벤트를 구독하는 데 사용된 경우, 구성이 변경되면 지속 가능 구독을 삭제하려고 시도하므로 구독이 Orphan 상태가 됩니다.

브릿지가 작성하는 지속 가능 구독은 다음과 같이 제거할 수 있습니다.

IBM MQ Explorer를 사용합니다.
: 브릿지가 사용 중인 큐 관리자의 **구독 폴더**를 열고 *:D_SUB_RUNMQSFB*로 끝나는 구독 이름을 찾으십시오. 여기서 토픽 문자열은 `/sf/mqtosfb/event+`입니다. 구독 이름을 마우스 오른쪽 단추로 클릭하고 삭제를 클릭하십시오. 구독이 사용 중임을 나타내는 오류가 표시된다면 브릿지가 계속 실행 중인 것일 수 있습니다. 브릿지를 중지한 후 구독을 다시 삭제해 보십시오.

`runmqsc`를 사용하여 구독을 찾아서 삭제합니다.
: **`runmqsc`** 인터페이스를 시작하고 `DISPLAY SUB (*)`을(를) 실행하십시오. *:D_SUB_RUNMQSFB*로 끝나는 등록 이름 **SUB**를 찾으십시오. delete sub 명령을 발행하되 삭제할 구독의 **SUBID**를 포함시키십시오. 예를 들어, 다음과 같습니다.`DELETE SUB SUBID(414D5120514D312020202020202020205C589459987E8620)`

`At-most-once` 서비스 품질(QoS)로 브릿지를 중지한 후 시작합니다.
: **`At-least-once`** 서비스 품질 `At-least-once delivery? (Y/N) :[Y]`(으)로 브릿지를 시작한 경우 작성되는 등록은 지속 가능합니다. 등록을 삭제하려면 구성 파일에서 서비스 품질을 `At-least-once delivery? (Y/N) :[N]`(으)로 변경하고 브릿지를 재시작하십시오. 지속 가능 구독이 삭제되고 지속 불가능 구독이 작성됩니다.

프로시저

1. 큐 관리자를 작성하고 시작하십시오.
   a) 큐 관리자(예: PEQM1)를 작성하십시오.
   ```
   crtmqm PEQM1
   ```
   b) 큐 관리자를 시작하십시오.
   ```
   strmqm PEQM1
   ```
2. **참고:** 기존 로그인 및 보안 Salesforce 신임 정보와 자체 서명 인증서를 사용하려면 4단계로 건너뛰십시오.

   옵션: Salesforce 계정에 대한 보안 토큰을 작성하십시오.
   a) Salesforce 계정에 로그인하십시오.
   b) 도움말 기사 Salesforce 도움말: 보안 토큰 재설정에 있는 단계를 수행하여 보안 토큰을 작성하거나 재설정하십시오.
3. Salesforce에서 자체 서명 보안 인증서를 작성하십시오.
   a) **Force.com 홈**의 **관리** 메뉴에서 **보안 제어**를 선택한 후 **인증서 및 키 관리**를 선택하십시오.

      인증서 및 키 관리 페이지가 열립니다.
   b) **자체 서명 인증서 작성**을 클릭하십시오.

      인증서 페이지가 열립니다.
   c) **레이블** 필드에 인증서의 이름을 입력하고 Tab을 누른 후 **저장**을 클릭하십시오.

      인증서 및 키 세부사항 정보가 표시됩니다.

d) **목록으로 돌아가기: 인증서 및 키**를 클릭하십시오.
e) **키 저장소에 내보내기**를 클릭하십시오.
f) 키 저장소의 비밀번호를 입력한 후 **내보내기**를 클릭하십시오.
g) 내보낸 키 저장소를 로컬 파일 시스템에 저장하십시오.

4. IBM Key Management GUI를 사용하여 Salesforce에서 내보낸 키 저장소를 열고 서명자 인증서를 채우십시오.

a) **strmqikm** 명령을 실행하여 IBM Key Management GUI를 여십시오. 자세한 정보는 디지털 인증서를 관리하기 위해 runmqckm, runmqakm 및 strmqikm 사용을 참조하십시오.
b) **키 데이터베이스 파일 열기**를 클릭한 후 Salesforce 키 저장소의 위치로 이동하십시오.
c) **열기**를 클릭하고 **키 데이터베이스 유형** 옵션에서 **JKS**를 선택했는지 확인한 후 **확인**을 클릭하십시오.
d) 3f단계에서 키 저장소에 대해 작성한 비밀번호를 입력한 후 **확인**을 클릭하십시오.
e) **키 데이터베이스 컨텐츠** 옵션에서 **서명자 인증서**를 선택하십시오.
f) **채우기**를 클릭하십시오.
g) **CA 인증서 추가** 목록에서 **Verisign Inc.** 선택란을 선택한 후 **확인**을 클릭하십시오.

5. 옵션: Salesforce 계정에서 IBM MQ Bridge to Salesforce에 대한 앱 연결을 작성하여 OAuth 컨슈머 키 및 시크릿을 생성하십시오.

프로덕션 환경에서 IBM MQ Bridge to Salesforce를 사용하는 경우 **컨슈머 키** 및 **컨슈머 시크릿** 코드가 필요합니다.

a) **작성**을 선택한 후 **Force.com 홈** 페이지의 **빌드** 메뉴에서 **앱**을 선택하십시오.
앱 페이지가 열립니다.
b) **연결된 앱** 섹션에서 **새로 작성**을 클릭하십시오.
새 연결된 앱 페이지가 열립니다.
c) **연결된 앱 이름**에 IBM MQ Bridge to Salesforce의 이름(예: **MQBridgeToSalesforce**)을 입력하십시오.
d) **API 이름**을 입력하십시오.
다음 필드로 탭하여 이동하는 경우 **연결된 앱 이름**이 **API 이름** 이름 필드에 복사됩니다.
e) **담당자 이메일**을 입력하십시오.
f) **API(OAuth 설정 사용)** 섹션에서 **OAuth 설정 사용** 옵션을 선택하십시오.
해당 섹션의 추가적인 옵션이 제공됩니다.
g) **콜백 URL**(예: `https://www.ibm.com`)을 추가하십시오.
h) **선택된 OAuth 범위** 하위 섹션의 **사용 가능한 OAuth 범위** 목록에서 **전체 액세스(전체)** 옵션을 선택한 후 **추가**를 클릭하여 **선택된 OAuth 범위** 목록에 대한 전체 액세스를 추가하십시오.
i) **저장**을 클릭하십시오.
j) **계속**을 클릭하십시오.
k) **컨슈머 키** 및 **컨슈머 시크릿** 코드를 기록해 두십시오.

6. 필요한 동기화 및 오류 큐를 큐 관리자에 작성하십시오.

```
cat /opt/mqm/mqsf/samp/mqsfbSyncQ.mqsc | runmqsc PEQM1
```

동기화 큐는 애플리케이션 또는 큐 관리자 재시작에서 이벤트 상태를 유지보수합니다. 큐에는 단일 메시지만 있을 것으로 예상되므로 큐 용량은 적을 수 있습니다. 이 큐에 대해 브릿지의 한 인스턴스만 한 번에 실행될 수 있으므로 독점 액세스를 위해 기본 옵션이 설정됩니다. 먼저 오류 큐를 작성해야 브릿지를 사용하여 플랫폼 이벤트에 대한 이벤트 메시지를 작성할 수 있습니다. 오류 큐는 Salesforce에서 성공적으로 처리할 수 없는 메시지에 사용됩니다. 577 페이지의 『8.a』 단계에 표시된 대로 브릿지 구성 매개변수 섹션 **`Connection to Queue Manager`**에 오류 큐 이름을 추가해야 합니다.

7. 옵션: Salesforce 계정에 플랫폼 이벤트 오브젝트를 작성하십시오.

a) **Force.com 홈** 페이지의 **개발** 메뉴에서 **플랫폼 이벤트**를 선택한 후 **새 플랫폼 이벤트**를 클릭하십시오.
새 플랫폼 이벤트 페이지가 열립니다.

b) **레이블** 및 **복수 레이블** 필드를 완성하십시오.

c) **저장**을 클릭하십시오.

플랫폼 이벤트 정의 세부사항 페이지가 열립니다.

d) **사용자 정의 필드 및 관계**를 정의하십시오.

예를 들어 *MyText* 및 *Name*이라는 레이블을 사용하는 두 개의 텍스트 필드를 추가하고 **데이터 유형** 필드 길이를 각각 *Text(64)* 및 *Text(32)*로 설정하십시오.

플랫폼 이벤트를 작성하고 해당 이벤트에 대해 **`Custom Fields and Relationships`**을(를) 정의했습니다. 플랫폼 이벤트 *Platform Object name* 또는 *API name*을 브릿지가 처리할 메시지를 놓을 IBM MQ 토픽으로 사용하십시오. 예를 들어 **AMQSPUBA** 샘플을 사용하여 다음과 같은 JSON 형식의 메시지를 `/sf/mqtosfb/event/`*`Salesforce Platform Object Name`*`/`*`API name`* 토픽에 추가하십시오.

```
{ "MyText__c" : "Some text here", "Name__c" : "Bob Smith" }
```

AMQSPUBA 샘플을 사용하여 브릿지가 시작된 후 메시지를 작성할 수 있습니다. *`MQ installation location`*`/samp/bin` 디렉토리에서 다음 명령을 발행하십시오.

```
./amqspub /sf/mqtosfb/event/Salesforce Platform Object Name/API name PEQM1
```

프롬프트에 JSON 형식의 메시지를 입력하십시오.

8. IBM MQ, Salesforce 및 IBM MQ Bridge to Salesforce 작동에 대해 연결 및 보안 매개변수를 사용하여 구성 파일을 작성하십시오.

```
runmqsfb -o new_config.cfg
```

기존 값이 대괄호 안에 표시됩니다. `Enter`를 눌러 기존 값을 승인하고 `Space`를 누른 후 `Enter`를 눌러 값 및 유형을 지운 다음 `Enter`를 눌러 새 값을 추가하십시오.

a) 큐 관리자 PEQM1에 대한 연결 값을 입력하십시오.

연결에 필요한 최소값은 큐 관리자 이름, IBM MQ 기본 토픽 루트, 오류 큐 이름 및 채널 이름입니다.

```
Connection to Queue Manager
---------------------------
Queue Manager or JNDI CF   : []PEQM1
MQ Base Topic              : []/sf
MQ Channel                 : []A channel you have defined or for example
SYSTEM.DEF.SVRCONN
MQ Conname                 : []
MQ Publication Error Queue : [SYSTEM.SALESFORCE.ERRORQ]
MQ CCDT URL                : []
JNDI implementation class  : [com.sun.jndi.fscontext.RefFSContextFactory]
JNDI provider URL          : []
MQ Userid                  : []
MQ Password                : []
```

참고: 로컬에서 연결하는 경우에는 채널 이름이 필요하지 않습니다. 큐 관리자 이름 및 기본 토픽은 나중에 브릿지를 실행할 때 명령행에 포함될 수 있으므로 구성 파일에 제공하지 않아도 됩니다.

b) Salesforce에 대한 연결을 위한 값을 입력하십시오.

연결에 필요한 최소값은 Salesforce 사용자 ID, 비밀번호, 보안 토큰 및 로그인 엔드 포인트입니다. 프로덕션 환경에서는 OAuth 보안에 대한 컨슈머 키 및 시크릿을 추가할 수 있습니다.

```
Connection to Salesforce
------------------------
Salesforce Userid (reqd)   : []salesforce_login_email
Salesforce Password (reqd) : []salesforce_login_password
Security Token (reqd)      : []Security_Token
Login Endpoint             : [https://login.salesforce.com]
Consumer ID                : []
Consumer Secret Key        : []
```

c) TLS 연결의 인증서 저장소를 위한 값을 입력하십시오.

TLS 연결에 필요한 최소값은 TLS 인증서 및 키 저장소 비밀번호의 경로입니다. 신뢰할 수 있는 저장소 경로 또는 비밀번호가 제공되지 않는 경우에는 키 저장소 및 비밀번호 매개변수가 신뢰할 수 있는 저장소

및 비밀번호에 사용됩니다. IBM MQ 큐 관리자 연결에 TLS를 사용하는 경우 동일한 키 저장소를 사용할 수 있습니다.

```
Certificate stores for TLS connections
--------------------------------------
Personal keystore for TLS certificates : []path_to_keystore, for example: /var/mqm/qmgrs/
PEQM1/ssl/key.jks
Keystore password           : []keystore_password
Trusted store for signer certificates : []
Trusted store password      : []
Use TLS for MQ connection   : [N]
```

d) 값을 입력하여 IBM MQ Bridge to Salesforce의 작동을 구성하십시오.

브릿지를 사용하여 이벤트 메시지를 작성하려면 **Subscribe to MQ publications for platform events** 옵션을 기본 *N*에서 *Y*로 변경해야 합니다. 또한 로그 파일도 구성 파일 또는 명령행에 지정해야 합니다.

```
Behaviour of bridge program
---------------------------
PushTopic Names             : []
Platform Event Names        : []
MQ Monitoring Frequency     : [30]
At-least-once delivery? (Y/N) : [Y]
Subscribe to MQ publications for platform events? (Y/N) : [Y]
Publish control data with the payload? (Y/N) : [N]
Delay before starting to process events : [0]
Runtime logfile for copy of stdout/stderr : []
```

9. 옵션: IBM MQ 서비스를 작성하여 프로그램 실행을 제어하십시오. 새로 작성된 구성 파일을 가리키도록 샘플 mqsfbService.mqsc 파일을 편집하고 명령 매개변수에 대한 기타 변경사항을 작성하십시오.

```
cat modified mqsfbService.mqsc | runmqsc PEQM1
```

10. V 9.0.1
옵션: IBM MQ Console 시작하기에 있는 지시사항을 수행하여 IBM MQ Console을 설정하십시오.

11. 옵션: IBM MQ Console 인스턴스에서 위젯을 추가하고 구성하여 Salesforce 데이터를 보십시오.

a) **위젯 추가**를 클릭하십시오.

새 위젯이 열립니다.

b) **차트**를 선택하십시오.

c) 새 위젯의 제목 표시줄에서 **위젯 구성** 아이콘을 클릭하십시오.

d) 옵션: **위젯 제목**을 입력하십시오.

e) **모니터할 자원, 소스** 드롭 다운 메뉴에서 **Salesforce 브릿지**를 선택하십시오.

f) **자원 클래스** 드롭 다운 메뉴에서 **브릿지 상태**를 선택하십시오.

g) **자원 유형** 드롭 다운 메뉴에서 **MQ 작성 플랫폼 이벤트**를 선택하십시오.

h) **자원 요소** 드롭 다운 메뉴에서 **MQ 작성 플랫폼 이벤트 합계**를 선택하십시오.

i) **저장**을 클릭하십시오.

IBM MQ에서 작성한 총 플랫폼 이벤트 수를 표시하도록 IBM MQ Console을 구성했습니다. 브릿지가 실행 중이고 메시지를 /sf/mqtosfb/event/*Salesforce Platform Object Name*/*API name* 토픽에 넣기 시작하면 브릿지가 작성한 총 메시지 이벤트 수가 위젯에 표시됩니다.

V 9.0.4 IBM MQ Bridge to Salesforce에 대한 메시지 형식 및 오류 메시지

IBM MQ Bridge to Salesforce에서 처리하는 메시지의 형식에 대한 정보입니다.

애플리케이션은 메시지를 특정 큐 관리자 토픽(예: /*root*/mqtosfb/event/MQPlatformEvent1__e)에 넣습니다. 브릿지는 토픽을 구독하고, 메시지에서 컨텐츠를 가져오며, 이를 사용하여 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 발행합니다.

브릿지를 사용하여 플랫폼 이벤트에 대한 이벤트 메시지를 작성하기 전에 Salesforce에 플랫폼 이벤트를 작성하고 컨텐츠 필드를 정의해야 합니다. 플랫폼 이벤트 이름과 컨텐츠에 따라 브릿지에서 처리하는 IBM MQ 메시지

의 형식화 방법이 결정됩니다. 예를 들어, Salesforce 플랫폼 이벤트 **Object name**이(가) *MQPlatformEvent1*이고 두 개의 사용자 정의 필드가 **API name** *MyText__c* 및 *Name__c*가 있는 텍스트 필드인 경우 `/root/mqtosfb/event/MQPlatformEvent1__e` 주제에 게시되는 IBM MQ 메시지는 다음과 같이 올바른 형식의 JSON이어야 합니다.

```
{ "MyText__c" : "Some text here", "Name__c" : "Bob Smith" }
```

브릿지에서 이용하고 생성하는 메시지는 JSON 형식의 텍스트(MQSTR) 메시지입니다. 입력 메시지는 단순 JSON이므로 프로그램에서는 문자열 연결을 사용하여 이를 생성할 수 있습니다.

오류 메시지

메시지가 텍스트 형식이 아니거나 Salesforce에서 플랫폼 이벤트 이름이 없을 경우 브릿지에서 오류를 감지할 수 있습니다. 입력 메시지 처리 중에 오류가 발생할 경우, 메시지는 오류를 설명하는 특성과 함께 브릿지 오류 큐로 이동됩니다. 오류는 브릿지의 *stderr* 스트림에도 기록됩니다.

Salesforce에서 생성되는 오류는 JSON입니다. 올바르지 않은 형식의 메시지로 인해 발생하는 몇 가지 오류는 다음과 같습니다.

잘못된 플랫폼 이벤트 컨텐츠, 상태 400 텍스트

```
[{"message":"No such column 'Name__c' on sobject of type
MQPlatformEvent2__e","errorCode":"INVALID_FIELD"}
```

올바르지 않은 플랫폼 이벤트 이름, 상태 404 텍스트

```
{"errorCode":"NOT_FOUND","message":"The requested resource does not exist"}
```

잘못된 JSON, 상태 400 텍스트

```
{"errorCode":"NOT_FOUND","message":"The requested resource does not exist"}
```

메시지가 JSON이 아님, 상태 400 텍스트

```
[{"message":
  "Unexpected character ('h' (code 104)): expected a valid value (number, String, array,
object, 'true', 'false' or 'null') at [line:1, column:2]",
  "errorCode":"JSON_PARSER_ERROR"}
```

텍스트 메시지가 아님(Salesforce로 전송되지 않음)

```
Error: Publication on topic ' /sf/mqtosfb/event/MQPlatformEvent1' does not contain a text
formatted message
```

V 9.0.2 Linux IBM MQ Bridge to Salesforce 실행

IBM MQ Bridge to Salesforce을(를) 실행하여 Salesforce 및 IBM MQ에 연결하십시오. 연결되면 브릿지는 Salesforce 토픽에 대한 구독을 작성하고 메시지를 IBM MQ 토픽에 재발행할 수 있습니다. V 9.0.4 IBM MQ 9.0.4에서 브릿지는 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성할 수도 있습니다.

시작하기 전에

다음 태스크에서 구성 단계를 완료했습니다.

- 570 페이지의 『IBM MQ Bridge to Salesforce』
- V 9.0.4 574 페이지의 『Salesforce 플랫폼 이벤트에 대한 이벤트 메시지 작성』

.

이 태스크 정보

이전의 태스크에서 작성한 구성 파일을 사용하여 IBM MQ Bridge to Salesforce을 실행하십시오. 구성 파일에 필요한 모든 매개변수를 포함하지 않은 경우에는 명령행에서 모두 포함해야 합니다.

프로시저

1. Salesforce에서 구독할 푸시 토픽이나 플랫폼 이벤트 V 9.0.4 또는 이벤트 메시지를 작성할 플랫폼 이벤트를 정의하십시오.
2. IBM MQ Bridge to Salesforce를 시작하여 Salesforce 및 큐 관리자]에 연결하십시오. 브릿지를 실행하여 Salesforce 이벤트를 구독하는 경우 1단계에서 정의한 푸시 토픽 또는 플랫폼 이벤트의 이름을 포함시키십시오.

   ```
   runmqsfb -f new_config.cfg -r logFile -p PushtopicName -e eventName
   ```

   브릿지가 연결되면 다음과 같은 메시지가 리턴됩니다.

   IBM MQ 9.0.2

   ```
   Successful connection to queue manager QM1
   Successful login to Salesforce at https://eu11.salesforce.com
   Ready to process events.
   ```

   V 9.0.4 IBM MQ 9.0.4

   - 브릿지를 사용하여 Salesforce 푸시 토픽 및 플랫폼 이벤트를 구독하는 경우

     ```
     Successful connection to queue manager QM1
     Warning: Subscribing to MQ-created platform events is not enabled.
     Successful login to Salesforce at https://eu11.salesforce.com
     Ready to process events.
     ```

   - 브릿지를 사용하여 Salesforce 플랫폼 이벤트에 대한 이벤트 메시지를 작성하는 경우

     ```
     Successful connection to queue manager QM1
     Successful login to Salesforce at https://eu11.salesforce.com
     Successful subscription to '/sf/mqtosfb/event/+' for MQ-created platform events
     Ready to process events.
     ```

3. 옵션: 브릿지를 실행한 후 리턴되는 메시지가 연결에 실패했음을 나타내는 경우 큐 관리자 및 Salesforce에 대한 연결의 문제점을 해결하십시오.

   a) 디버그 옵션 1을 사용하여 디버그 모드에서 명령을 실행하십시오.

   ```
   runmqsfb -f new_config.cfg -r logFile -p PushtopicName -e eventName -d 1
   ```

   브릿지는 연결 설정을 스텝 스루하고 간결 모드로 처리 메시지를 표시합니다.

   b) 디버그 옵션 2를 사용하여 디버그 모드에서 명령을 실행하십시오.

   ```
   runmqsfb -f new_config.cfg -r logFile -p PushtopicName -e eventName -d 2
   ```

   브릿지는 연결 설정을 스텝 스루하고 상세 모드로 처리 메시지를 표시합니다. 전체 출력이 로그 파일에 기록됩니다.

4. Salesforce 인터페이스를 사용하여 데이터베이스에서 레코드를 수정하여 이벤트를 생성하십시오.
5. IBM MQ Console로 이동하여 푸시 토픽에 대한 변경사항이 이전의 태스크에서 구성된 위젯에 표시되는지 확인하십시오.

다음에 수행할 작업

MQSFB_EXTRA_JAVA_OPTIONS 변수를 사용하여 JVM 특성을 전달하십시오(예를 들어, IBM MQ 추적을 사용으로 설정하기 위해). 자세한 정보는 IBM MQ Bridge to Salesforce 추적을 참조하십시오.

관련 정보
runmqsfb(IBM MQ Bridge to Salesforce 실행)
IBM MQ Bridge to Salesforce 모니터링

V 9.0.4 MQ Adv. Linux 블록체인과 함께 사용하도록 IBM MQ 구성

Set up and run the IBM MQ Bridge to blockchain to securely connect an IBM MQ Advanced queue manager and IBM Blockchain. 브릿지를 사용하면 IBM MQ Advanced 큐 관리자에 연결되는 메시징 애플리케이션을 통해 블록체인의 자원에 비동기적으로 연결하고 자원의 상태를 조회 및 업데이트할 수 있습니다.

시작하기 전에

- IBM MQ Bridge to blockchain은 IBM MQ Advanced 큐 관리자에 연결하는 데만 사용할 수 있습니다.
- 큐 관리자는 브릿지와 동일한 명령 레벨(예: IBM MQ 9.0.4)이어야 합니다.
- IBM MQ Bridge to blockchain은 Hyperledger Fabric 1.0 architecture 기반의 블록체인 네트워크에서 사용할 수 있도록 지원됩니다.

이 태스크 정보

블록체인은 네트워크에서 피어 사이의 트랜잭션에 동의하는 블록 체인으로 구성되는 공유 및 분배식 디지털 원장입니다. 체인 내의 각 블록은 이전 블록에 링크되며 이런 방식으로 계속되어 첫 번째 트랜잭션에 링크됩니다.

IBM Blockchain은 Hyperledger Fabric을 기반으로 빌드되며, Docker를 사용하거나 IBM Cloud (formerly Bluemix®)의 컨테이너 클러스터에서 로컬로 개발할 수 있습니다. IBM Blockchain 네트워크를 프로덕션 중에 활성화하고 사용하여 높은 수준의 보안, 개인정보 보호 및 성능으로 비즈니스 네트워크를 빌드하고 관리할 수도 있습니다. 자세한 정보는 IBM Blockchain Platform을 참조하십시오.

Hyperledger Fabric is an open source, enterprise blockchain framework that is developed collaboratively by members of the Hyperledger Project, including IBM as the initial code contributor. Hyperledger Project, 또는 Hyperledger는 교차 산업 블록체인 기술을 발전시키기 위한 Linux Foundation 오픈 소스, 글로벌, 협업 이니셔티브입니다. 자세한 정보는 IBM Blockchain, Hyperledger 프로젝트 및 Hyperledger Fabric을 참조하십시오.

IBM MQ Advanced 및 IBM Blockchain을 이미 사용 중인 경우, IBM MQ Bridge to blockchain을 사용하여 간단한 조회와 업데이트를 송신하고 블록체인 네트워크에서 응답을 수신할 수 있습니다. 이러한 방식으로 사내 구축형 IBM 소프트웨어를 클라우드 블록체인 서비스와 통합할 수 있습니다.

브릿지 운영 프로세스에 대한 간략한 개요는 그림 1에서 볼 수 있습니다. 사용자 애플리케이션은 IBM MQ Advanced 큐 관리자의 입력/요청 큐에 JSON 형식의 메시지를 넣습니다. 브릿지는 큐 관리자에 연결하여 입력/요청 큐에서 메시지를 가져오고 JSON이 올바르게 형식화되었는지 확인한 후 조회 또는 업데이트를 블록체인에 발행합니다. 블록체인에서 리턴하는 데이터는 원래 IBM MQ 요청 메시지에 정의된 것과 같이, 브릿지를 통해 구문 분석된 후 응답 큐에 배치됩니다. 사용자 애플리케이션은 큐 관리자에 연결하여 응답 큐의 응답 메시지를 가져오며 정보를 사용할 수 있습니다.

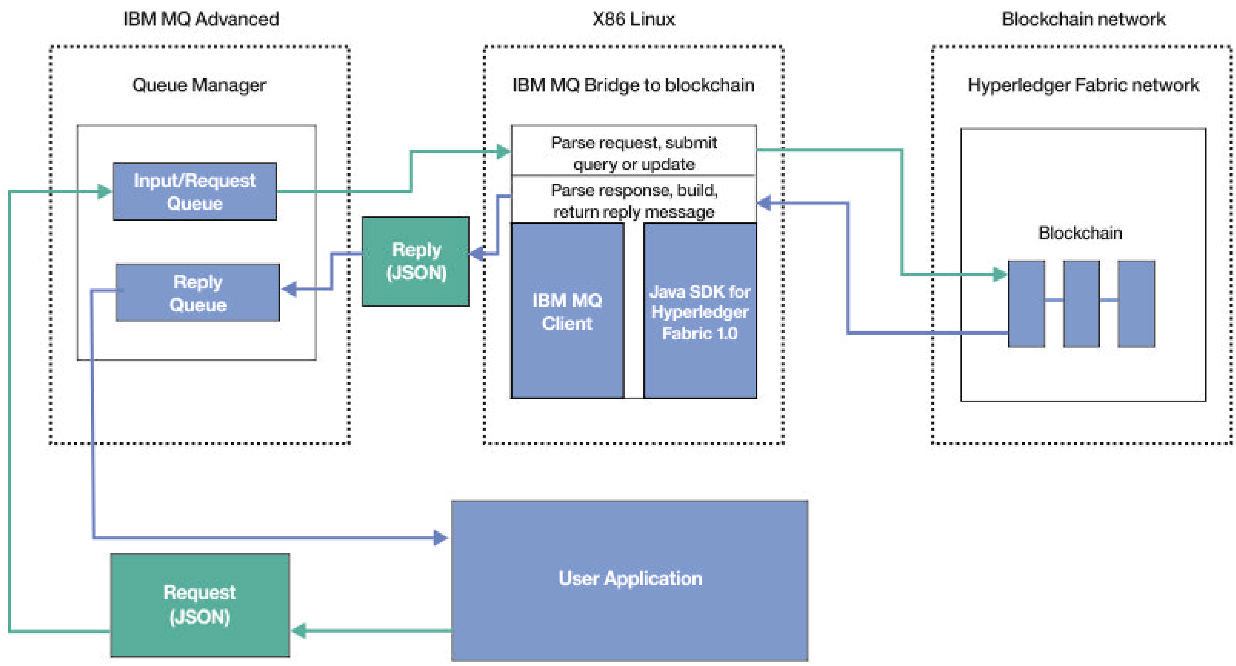


그림 98. *IBM MQ Bridge to blockchain*

IBM MQ Bridge to blockchain이 블록체인 네트워크에 참가자 또는 피어로 연결되도록 구성할 수 있습니다. 브릿지가 실행 중인 경우, 메시징 애플리케이션은 자원의 상태를 조회 또는 업데이트하여 그 결과를 메시징 애플리케이션에 응답으로 리턴하는 체인코드 루틴을 구동하도록 브릿지에 요청합니다.

프로시저

1. 큐 관리자를 작성하여 시작하거나, IBM MQ Bridge to blockchain과 함께 사용할 기존 큐 관리자를 시작하십시오.

   큐 관리자를 작성하십시오.

   ```
   crtmqm adv_qmgr_name
   ```

   큐 관리자를 시작하십시오.

   ```
   strmqm adv_qmgr_name
   ```

2. **DefineQ.mqsc** 스크립트에 정의된 브릿지용 큐를 작성하십시오.

   샘플 브릿지 큐 정의는 다음에 사용되는 이름 지정된 기본 큐에 제공됩니다.

   - 사용자 신임 정보(예: SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE)
   - 브릿지에 대한 메시지 입력 (예: APPL1.BLOCKCHAIN.INPUT.QUEUE)
   - 블록체인에서의 응답(예: APPL1.BLOCKCHAIN.REPLY.QUEUE)

   /opt/mqm/mqbc/samp 디렉토리에서 다음 명령을 발행하십시오.

   ```
   runmqsc adv_qmgr_name < ./DefineQ.mqsc
   ```

   서로 다른 애플리케이션이 같은 입력 큐를 사용할 수 있지만, 응답 큐는 애플리케이션당 한 개씩 여러 개 지정할 수 있습니다. 정의된 응답 큐를 사용할 필요는 없습니다. 응답에 동적 큐를 사용하려는 경우 보안 구성을 고려해야 합니다.

결과

IBM MQ 및 블록체인 네트워크에서 메시지를 처리하기 위해 브릿지에 필요한 큐를 작성했습니다.

다음에 수행할 작업

IBM MQ Advanced 큐 관리자 정보 및 블록체인 네트워크의 신임 정보를 사용하여 IBM MQ Bridge to blockchain용 구성 파일을 작성하십시오.

V 9.0.4 IBM MQ Bridge to blockchain용 구성 파일 작성

IBM MQ Bridge to blockchain용 구성 파일을 작성하여 IBM MQ 및 IBM Blockchain 네트워크에 연결하려면 큐 관리자와 블록체인 네트워크 매개변수를 입력하십시오.

시작하기 전에

- 블록체인 네트워크를 작성하고 구성했습니다.
- 블록체인 네트워크의 신임 정보 파일이 있습니다.
- IBM MQ Bridge to blockchain을 x86 Linux 환경에 설치했습니다.
- IBM MQ Advanced 큐 관리자를 시작했습니다.

이 태스크 정보

이 태스크는 IBM MQ Bridge to blockchain 구성 파일을 작성하고 IBM Blockchain 및 IBM MQ 네트워크에 성공적으로 연결하는 데 필요한 최소 설정을 안내합니다.

브릿지를 사용하여 Hyperledger Fabric 1.0 architecture를 기반으로 하는 블록체인 네트워크에 연결할 수 있습니다. 브릿지를 사용하려면 블록체인 네트워크의 구성 정보가 필요합니다. 이 태스크의 각 단계에서는 서로 다르게 구성된 다음 두 블록체인 네트워크를 기반으로 하는 구성 세부사항 예를 찾을 수 있습니다.

- Docker에서 실행되는 Hyperledger Fabric 네트워크. 자세한 정보는 Hyperledger Fabric 시작하기, 첫 번째 애플리케이션 작성 및 585 페이지의 『Hyperledger Fabric 네트워크 신임 정보 파일 예』의 내용을 참조하십시오.
- IBM Cloud (formerly Bluemix)의 Kubernetes 클러스터에서 실행되는 Hyperledger Fabric 네트워크. 자세한 정보는 IBM Blockchain Platform에서 클라우드 샌드박스로 개발 및 587 페이지의 『Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예』의 내용을 참조하십시오.

모든 IBM MQ Bridge to blockchain 매개변수의 의미와 옵션에 대한 자세한 정보는 runmqbcb 명령을 참조하십시오. 자체 보안 요구사항을 고려하고 배치에 적합하게 매개변수를 사용자 정의해야 합니다.

프로시저

1. 브릿지를 실행하여 구성 파일을 작성하십시오.

   블록체인 네트워크 신임 정보 파일 및 IBM MQ Advanced 큐 관리자에서 제공하는 매개변수가 필요합니다.

   ```
   runmqbcb -o config_file_name.cfg
   ```

   다음 예에 표시된 것과 같이, 기존 값은 대괄호 안에 표시됩니다. 기존 값을 승인하려면 `Enter`를 누르고, 값을 지우려면 `Space`를 누른 후 `Enter`를 누르며, 새 값을 추가하려면 대괄호 안에 값을 입력한 후 `Enter`를 누르십시오. 값 목록(예: 피어)은 쉼표로 구분하거나 각 값을 줄바꿈하여 구분할 수 있습니다. 빈 행이 있으면 목록이 종료됩니다.

   참고: 기존 값은 편집할 수 없습니다. 대신 기존 값을 유지, 바꾸기 또는 지울 수는 있습니다.

2. IBM MQ Advanced 큐 관리자에 대한 연결 값을 입력하십시오.

   연결에 필요한 최소값은 큐 관리자 이름, 브릿지 입력 이름 및 정의한 ID 큐입니다. 리모트 큐 관리자에 대한 연결의 경우, **`MQ Channel`** 및 **`MQ Conname`** (큐 관리자가 실행 중인 호스트 주소 및 포트) 도 필요합니다. To use TLS for connecting to IBM MQ in step 584 페이지의 『6』, you must use JNDI or CCDT and specify **`MQ CCDT URL`** or **`JNDI implementation class`** and **`JNDI provider URL`** accordingly.

   ```
   Connection to Queue Manager
   ---------------------------
   Queue Manager                   : [adv_qmgr_name]
   Bridge Input Queue              : [APPL1.BLOCKCHAIN.INPUT.QUEUE]
   ```

```
Bridge User Identity Queue           : [SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE]
MQ Channel                           : []
MQ Conname                           : []
MQ CCDT URL                          : []
JNDI implementation class            : []
JNDI provider URL                    : []
MQ Userid                            : []
MQ Password                          : []
```

3. 블록체인 네트워크의 인증 기관에 대한 로그인 세부사항을 입력하십시오.

   로컬 Hyperledger Fabric 및 Kubernetes 클러스터 예제의 기본값은 **Userid** 의 경우 *admin* 이고 **Enrollment Secret**의 경우 *adminpw* 입니다. 블록체인 네트워크의 이 값을 변경한 경우 올바른 값을 사용하여 브릿지를 구성했는지 확인하십시오.

```
Blockchain - User Identification
--------------------------------
Blockchain Userid                    : []admin
Enrollment Secret                    : []******
```

4. 블록체인 네트워크에 대한 멤버십 및 ID 규칙을 제어하는 멤버십 서비스 제공자 ID(**MSPid**)를 입력하십시오.

   신임 정보 파일에서 **Organisation Name** 및 **Organisation MSPId**에 대한 **msp_id** 매개변수를 제공하십시오. 585 페이지의 『Hyperledger Fabric 네트워크 신임 정보 파일 예』에서 파일의 피어 섹션에서 **CORE_PEER_LOCALMSPID** 값을 사용하십시오. 587 페이지의 『Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예』에서는 **mspID** 값을 사용하십시오.

```
Blockchain - Organisation Identification
----------------------------------------
Organisation Name                    : []Org1MSP
Organisation MSPId                   : []Org1MSP
```

5. 블록체인 네트워크 서버 위치 값을 입력하십시오.

   585 페이지의 『Hyperledger Fabric 네트워크 신임 정보 파일 예』에서 인증서 권한, 피어 및 orderer 요소의 이름 및 서버: 포트 위치를 제공하십시오.

```
Blockchain server locations
---------------------------
Certificate Authority servers        : [ca.example.com Docker_container_host:7054] (for
example ca.example.com localhost:7054)
Peer servers                         : [peer0 localhost:7051]
Orderer servers                      : [orderer0 localhost:7050]
Peer Event servers                   : [peer0 localhost:7053]
Location of PEM file for Blockchain certificate : []
```

   587 페이지의 『Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예』에서 인증서 권한, 피어 및 orderer 요소의 이름 및 서버: 포트 위치를 제공하십시오.

```
Blockchain server locations
---------------------------
Certificate Authority servers        : [CA1
your_blockchain_network_public_ip_address:30000] (for example CA1 123.456.789.10:30000)
Peer servers                         : [blockchain-org1peer1
your_blockchain_network_public_ip_address:30110]
Orderer servers                      : [blockchain-orderer
your_blockchain_network_public_ip_address:31010]
Peer Event servers                   : [blockchain-org1peer1
your_blockchain_network_public_ip_address:30111]
Location of PEM file for Blockchain certificate : []
```

6. TLS 연결에 대한 인증 저장소 값을 입력하십시오.

   브릿지는 큐 관리자에 연결되는 IBM MQ Java 클라이언트로 사용됩니다. 즉, TLS 보안을 사용하여 다른 IBM MQ Java 클라이언트와 동일한 방식으로 안전하게 연결하도록 구성할 수 있습니다. TLS 연결 구성 세부사항은 583 페이지의 『2』 단계에서 JNDI 또는 CCDT 정보를 지정한 경우에만 표시됩니다.

```
Certificate stores for TLS connections
--------------------------------------
```

```
Personal keystore                     : []
Keystore password                     : []
Trusted store for signer certs        : []
Trusted store password                : []
Use TLS for MQ connection             : [N]
Timeout for Blockchain operations     : [12]
```

7. IBM MQ Bridge to blockchain 로그 파일의 위치를 입력하십시오.
   로그 파일 이름과 위치는 구성 파일 또는 명령행에 지정해야 합니다.

```
Behavior of bridge program
--------------------------
Runtime logfile for copy of stdout/stderr : [/var/mqm/errors/runmqbcb.log]
Done.
```

결과

IBM MQ Bridge to blockchain에서 IBM Blockchain 네트워크 및 IBM MQ Advanced 큐 관리자에 연결할 때 사용할 구성 파일을 작성했습니다.

다음에 수행할 작업

587 페이지의 『IBM MQ Bridge to blockchain 실행』에 대한 단계를 수행하십시오.

Hyperledger Fabric 네트워크 신임 정보 파일 예

Docker에서 실행되는 로컬로 인스턴스화된 Hyperledger Fabric 블록체인 네트워크의 .yml 파일 콘텐츠로 IBM MQ Bridge to blockchain을(를) 구성하는 데 사용할 수 있습니다.

Hyperledger Fabric 시작하기 학습서를 통해 작업하고 이후에 어떻게 진행되는지 이해하고 Hyperledger Fabric 샘플 중 하나를 사용하여 네트워크를 실행한 후 /blockchain/fabric-samples/basic-network 폴더에 다음 구성 파일이 있어야 합니다.

블록체인 네트워크에 연결하려는 경우에는 583 페이지의 『IBM MQ Bridge to blockchain용 구성 파일 작성』을 수행할 때 이 파일의 구성 세부사항을 사용해야 합니다.

```
#
# Copyright IBM Corp All Rights Reserved
#
# SPDX-License-Identifier: Apache-2.0
#
version: '2'

networks:
  basic:

services:
  ca.example.com:
    image: hyperledger/fabric-ca
    environment:
      - FABRIC_CA_HOME=/etc/hyperledger/fabric-ca-server
      - FABRIC_CA_SERVER_CA_NAME=ca.example.com
    ports:
      - "7054:7054"
    command: sh -c 'fabric-ca-server start --ca.certfile /etc/hyperledger/fabric-ca-server-
config/ca.org1.example.com-cert.pem --ca.keyfile
/etc/hyperledger/fabric-ca-server-config/
f329434b83a06f32f17a300fefd841cfd16ff58f3185fb744aae047207b01a9e_sk -b admin:adminpw -d'
    volumes:
      - ./crypto-config/peerOrganizations/org1.example.com/ca/:/etc/hyperledger/fabric-ca-
server-config
    container_name: ca.example.com
    networks:
      - basic

  orderer.example.com:
    container_name: orderer.example.com
    image: hyperledger/fabric-orderer
    environment:
      - ORDERER_GENERAL_LOGLEVEL=debug
```

```
      - ORDERER_GENERAL_LISTENADDRESS=0.0.0.0
      - ORDERER_GENERAL_GENESISMETHOD=file
      - ORDERER_GENERAL_GENESISFILE=/etc/hyperledger/configtx/genesis.block
      - ORDERER_GENERAL_LOCALMSPID=OrdererMSP
      - ORDERER_GENERAL_LOCALMSPDIR=/etc/hyperledger/msp/orderer/msp
    working_dir: /opt/gopath/src/github.com/hyperledger/fabric/orderer
    command: orderer
    ports:
      - 7050:7050
    volumes:
        - ./config/:/etc/hyperledger/configtx
        - ./crypto-config/ordererOrganizations/example.com/orderers/orderer.example.com/:/etc/
hyperledger/msp/orderer
        - ./crypto-config/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/:/etc/
hyperledger/msp/peerOrg1
    networks:
      - basic

  peer0.org1.example.com:
    container_name: peer0.org1.example.com
    image: hyperledger/fabric-peer
    environment:
      - CORE_VM_ENDPOINT=unix:///host/var/run/docker.sock
      - CORE_PEER_ID=peer0.org1.example.com
      - CORE_LOGGING_PEER=debug
      - CORE_CHAINCODE_LOGGING_LEVEL=DEBUG
      - CORE_PEER_LOCALMSPID=Org1MSP
      - CORE_PEER_MSPCONFIGPATH=/etc/hyperledger/msp/peer/
      - CORE_PEER_ADDRESS=peer0.org1.example.com:7051
      # # the following setting starts chaincode containers on the same
      # # bridge network as the peers
      # # https://docs.docker.com/compose/networking/
      - CORE_VM_DOCKER_HOSTCONFIG_NETWORKMODE=${COMPOSE_PROJECT_NAME}_basic
      - CORE_LEDGER_STATE_STATEDATABASE=CouchDB
      - CORE_LEDGER_STATE_COUCHDBCONFIG_COUCHDBADDRESS=couchdb:5984
      # The CORE_LEDGER_STATE_COUCHDBCONFIG_USERNAME and
CORE_LEDGER_STATE_COUCHDBCONFIG_PASSWORD
      # provide the credentials for ledger to connect to CouchDB.  The username and password
must
      # match the username and password set for the associated CouchDB.
      - CORE_LEDGER_STATE_COUCHDBCONFIG_USERNAME=
      - CORE_LEDGER_STATE_COUCHDBCONFIG_PASSWORD=
    working_dir: /opt/gopath/src/github.com/hyperledger/fabric
    command: peer node start
    # command: peer node start --peer-chaincodedev=true
    ports:
      - 7051:7051
      - 7053:7053
    volumes:
        - /var/run/:/host/var/run/
        - ./crypto-config/peerOrganizations/org1.example.com/peers/peer0.org1.example.com/
msp:/etc/hyperledger/msp/peer
        - ./crypto-config/peerOrganizations/org1.example.com/users:/etc/hyperledger/msp/users
        - ./config:/etc/hyperledger/configtx
    depends_on:
      - orderer.example.com
      - couchdb
    networks:
      - basic

  couchdb:
    container_name: couchdb
    image: hyperledger/fabric-couchdb
    # Populate the COUCHDB_USER and COUCHDB_PASSWORD to set an admin user and password
    # for CouchDB.  This will prevent CouchDB from operating in an "Admin Party" mode.
    environment:
      - COUCHDB_USER=
      - COUCHDB_PASSWORD=
    ports:
      - 5984:5984
    networks:
      - basic

  cli:
    container_name: cli
    image: hyperledger/fabric-tools
    tty: true
    environment:
      - GOPATH=/opt/gopath
      - CORE_VM_ENDPOINT=unix:///host/var/run/docker.sock
      - CORE_LOGGING_LEVEL=DEBUG
      - CORE_PEER_ID=cli
```

```
      - CORE_PEER_ADDRESS=peer0.org1.example.com:7051
      - CORE_PEER_LOCALMSPID=Org1MSP
      -
CORE_PEER_MSPCONFIGPATH=/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/
peerOrganizations/org1.example.com/users/Admin@org1.example.com/msp
      - CORE_CHAINCODE_KEEPALIVE=10
    working_dir: /opt/gopath/src/github.com/hyperledger/fabric/peer
    command: /bin/bash
    volumes:
        - /var/run/:/host/var/run/
        - ./../chaincode/:/opt/gopath/src/github.com/
        - ./crypto-config:/opt/gopath/src/github.com/hyperledger/fabric/peer/crypto/
    networks:
        - basic
    #depends_on:
    #  - orderer.example.com
    #  - peer0.org1.example.com
    #  - couchdb
```

Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예

IBM MQ Bridge to blockchain을 구성하는 데 사용할 수 있으며, IBM Cloud (formerly Bluemix)의 Kubernetes 클러스터에서 실행되는 Hyperledger Fabric 블록체인 네트워크에서 제공하는 구성 파일의 컨텐츠입니다.

IBM Blockchain 준비 및 설정, 단순 설치 및 블록체인과의 상호작용 학습서를 통해 작업한 후에는 사용자 연결 프로파일 폴더에 JSON 파일이 있어야 합니다.

블록체인 네트워크에 연결하려는 경우에는 583 페이지의 『IBM MQ Bridge to blockchain용 구성 파일 작성』을 수행할 때 이 파일의 구성 세부사항을 사용해야 합니다.

```
{
     "name": "ibm-bc-org1",
     "description": "Connection profile for IBM Blockchain Platform",
     "type": "hlfv1",
     "orderers": [
         {
             "url": "grpc://INSERT_PUBLIC_IP:31010"
         }
     ],
     "ca": {
         "url": "http://INSERT_PUBLIC_IP:30000",
         "name": "CA1"
     },
     "peers": [
         {
             "requestURL": "grpc://INSERT_PUBLIC_IP:30110",
             "eventURL": "grpc://INSERT_PUBLIC_IP:30111"
         }
     ],
     "keyValStore": "INSERT_CREDENTIALS_PATH",
     "channel": "channel1",
     "mspID": "Org1MSP",
     "timeout": 300
 }
```

V 9.0.4 IBM MQ Bridge to blockchain 실행

IBM MQ Bridge to blockchain를 실행하여 IBM Blockchain 및 IBM MQ에 연결합니다. 연결되면 브릿지가 조회 및 업데이트 메시지를 처리하고, 이 메시지를 블록체인 네트워크로 송신하며, 응답을 수신하고 처리할 수 있습니다.

이 태스크 정보

이전의 태스크에서 작성한 구성 파일을 사용하여 IBM MQ Bridge to blockchain을 실행하십시오.

프로시저

1. 브릿지와 함께 사용할 IBM MQ Advanced 큐 관리자를 시작하십시오.

2. IBM MQ Bridge to blockchain을 시작하여 블록체인 네트워크와 IBM MQ Advanced 큐 관리자에 연결하십시오.

   브릿지 명령을 실행하십시오.

   ```
   runmqbcb -f /config_file_location/config_file_name.cfg -r /log_file_location/logFile.log
   ```

   브릿지가 연결되면 다음과 유사한 출력이 리턴됩니다.

   ```
   Fri Oct 06 06:32:11 PDT 2017 IBM MQ Bridge to Blockchain
   5724-H72 (C) Copyright IBM Corp. 2017, 2023.

   Fri Oct 06 06:32:17 PDT 2017 Ready to process input messages.
   ```

3. 옵션: 브릿지를 실행한 후 리턴되는 메시지가 연결에 실패했음을 나타내는 경우 IBM MQ Advanced 큐 관리자 및 블록체인 네트워크에 대한 연결 문제점을 해결하십시오.
   a) 디버그 옵션 1을 사용하여 디버그 모드에서 명령을 실행하십시오.

      ```
      runmqbcb -f /config_file_location/config_file_name.cfg -r /log_file_location/logFile.log
      -d 1
      ```

      브릿지는 연결 설정을 스텝 스루하고 간결 모드로 처리 메시지를 표시합니다.
   b) 디버그 옵션 2를 사용하여 디버그 모드에서 명령을 실행하십시오.

      ```
      runmqbcb -f /config_file_location/config_file_name.cfg -r /log_file_location/logFile.log
      -d 2
      ```

      브릿지는 연결 설정을 스텝 스루하고 상세 모드로 처리 메시지를 표시합니다. 전체 출력이 로그 파일에 기록됩니다.

결과

IBM MQ Bridge to blockchain을 시작하고 큐 관리자와 블록체인 네트워크에 연결했습니다.

다음에 수행할 작업

- 718 페이지의 『IBM MQ Bridge to blockchain 클라이언트 샘플 실행』에 설명된 단계에 따라 조회 또는 업데이트 메시지를 형식화한 후 블록체인 네트워크로 송신하십시오.
- *MQBCB_EXTRA_JAVA_OPTIONS* 변수로 JVM 특성을 전달하여 IBM MQ 추적을 사용으로 설정하십시오. 자세한 정보는 IBM MQ Bridge to blockchain 추적을 참조하십시오.

V 9.0.4 IBM MQ Bridge to blockchain에 대한 메시지 형식

IBM MQ Bridge to blockchain에서 송신하고 수신한 메시지의 형식에 대한 정보입니다.

애플리케이션은 IBM MQ Bridge to blockchain이 블록체인에 보유된 정보의 조회 또는 업데이트를 수행하도록 요청합니다. 애플리케이션은 이를 위해 요청 메시지를 브릿지 요청 큐에 배치합니다. 조회 또는 업데이트 결과는 브릿지를 통해 응답 메시지로 형식화됩니다. 브릿지는 요청 메시지의 MQMD에서 **ReplyToQ** 및 **ReplyToQMgr** 필드에 포함된 정보를 응답 메시지의 목적지로 사용합니다.

브릿지에서 이용하고 생성하는 메시지는 JSON 형식의 텍스트(MQSTR) 메시지입니다. 입력 메시지는 단순 JSON이므로 프로그램에서는 문자열 연결을 사용하여 이를 생성할 수 있습니다. **args**를 제외한 모든 필드는 필수이므로, 해당 필드에 대한 인수 목록을 사용하려면 저장된 체인코드의 함수를 알고 있어야 합니다.

요청 메시지 형식

입력 메시지 형식:

```
{ "function": functionName,
    "channel" : chainName,
    "chaincodeName" : codeName,
```

```
    "args" : [argument list]
}
```

Fabcar 샘플이 작동 중인 로컬 Hyperledger 네트워크 예

- 블록체인에 보유된 자동차 세부사항을 나타내는 JSON 오브젝트 목록을 리턴하는 Fabcar 체인코드에 `queryAllCars` 함수를 호출하는 조회 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

  ```
  { "function": "queryAllCars",
      "channel":"mychannel",
      "chaincodeName": "fabcar",
      "args":[]
  }
  ```

  응답 예:

  ```
  {
     "statusCode": 200,
     "statusType": "SUCCESS",
     "message": "OK",
     "data": [
    {"Record":{"owner":"Tomoko","colour":"blue","model":"Prius","make":"Toyota"},"Key":"CAR0"},
    {"Record":{"owner":"Brad","colour":"red","model":"Mustang","make":"Ford"},"Key":"CAR1"},
    {"Record":{"owner":"Jin
  Soo","colour":"green","model":"Tucson","make":"Hyundai"},"Key":"CAR2"},
    {"Record":
  {"owner":"Max","colour":"yellow","model":"Passat","make":"Volkswagen"},"Key":"CAR3"},
    {"Record":{"owner":"Adriana","colour":"black","model":"S","make":"Tesla"},"Key":"CAR4"},
    {"Record":
  {"owner":"Michel","colour":"purple","model":"205","make":"Peugeot"},"Key":"CAR5"},
    {"Record":{"owner":"Aarav","colour":"white","model":"S22L","make":"Chery"},"Key":"CAR6"},
    {"Record":{"owner":"Pari","colour":"violet","model":"Punto","make":"Fiat"},"Key":"CAR7"},
    {"Record":{"owner":"Valeria","colour":"indigo","model":"Nano","make":"Tata"},"Key":"CAR8"},
    {"Record":
  {"owner":"Shotaro","colour":"brown","model":"Barina","make":"Holden"},"Key":"CAR9"}
  ]}
  ```

  응답 메시지에는 블록체인에 현재 보유된 모든 자동차 레코드가 포함되어 있습니다.

- 블록체인 원장에 새 자동차 항목을 작성하는 fabcar 체인코드 예에서 `createCar` 함수를 작성하는 업데이트 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

  ```
  { "function":"createCar",
    "channel":"mychannel",
    "chaincodeName":"fabcar",
    "args":["CAR10", "Ford", "Mustang GT", "Blue", "Bob"]
  }
  ```

  응답 예:

  ```
  {
     "statusCode": 200,
     "statusType": "SUCCESS",
     "message": "OK",
     "data": ""
  }
  ```

  블록체인에 새 자동차 항목이 추가되었는지 확인하려면 모든 자동차를 리턴하는 초기 메시지를 다시 사용하면 됩니다.

example02 데모가 작동 중인 Kubernetes 클러스터 네트워크 예

- 블록체인 원장 내에 *"a"* 엔티티에 대한 값을 리턴하는 example02 체인코드에 `query` 함수를 호출하는 조회 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

  ```
  { "function":"query",
    "channel":"channel1",
    "chaincodeName":"example02",
    "args":["a"]
  }
  ```

  응답 예:

```
{
   "statusCode": 200,
   "statusType": "SUCCESS",
   "message": "OK",
   "data": "100"
}
```

- 세 번째 인수에 지정된 값만큼 첫 번째 인수에 지정된 엔티티는 감소하고 두 번째 인수에 지정된 엔티티는 증분하는 example02 체인코드에 invoke 함수를 호출하는 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

  ```
  { "function":"invoke",
    "channel":"channel1",
    "chaincodeName":"example02",
    "args":["a", "b", "10"]
  }
  ```

  값은 다음과 같습니다.

  - 이전: a=100, b=200
  - 이후: a=90, b=210

  응답 예:

  ```
  {
     "statusCode": 200,
     "statusType": "SUCCESS",
     "message": "OK",
     "data": ""
  }
  ```

  새 값을 확인하려면 **"a"** 및 **"b"**의 값을 찾는 새 메시지 조회 메시지를 제출하십시오.

응답 메시지 형식

응답 메시지의 상관 ID는 인바운드 메시지의 메시지 ID로 설정됩니다. 사용자 특성은 입력에서 출력 메시지로 복사됩니다. 응답의 사용자 ID는 set-identity 컨텍스트를 통해 발신자의 사용자 ID로 설정됩니다.

성공적인 처리 예:

```
{ "data": "500", "message": "OK", "statusCode": 200, "statusType": "SUCCESS" }
```

이 메시지의 응답 데이터는 체인코드 응답(바이트가 UTF-8 문자열로 변환됨)에서 생성됩니다.

모든 오류 응답은 브릿지, 블록체인에 대한 호출 또는 체인코드 호출을 통해 생성되었는지 여부에 관계없이 동일한 필드를 갖습니다. 예를 들면, 다음과 같습니다.

- 잘못된 채널 이름

  ```
  {
     "message": "Bad newest block expected status 200 got 404, Chain myUnknownChannel",
     "statusCode": 404,
     "statusType": "FAILURE"
  }
  ```

- 잘못된 JSON 입력 메시지

  ```
  {
     "message": "Error: Cannot parse message contents.",
     "statusCode": 2110,
     "statusType": "FAILURE"
  }
  ```

- 잘못된 체인코드 매개변수

  ```
  {
     "message": "Sending proposal to fabric-peer-1a failed because of gRPC
  ```

```
failure=Status{code=UNKNOWN, description={\"Error\":\"Nil amount for c\"}, cause=null}",
   "statusCode": 500,
   "statusType": "FAILURE"
}
```

애플리케이션은 **statusType** 문자열을 검토하거나 데이터 필드가 있는지 검토하여 요청이 성공 또는 실패했는지 여부를 알릴 수 있습니다. 입력 메시지를 처리할 때 오류가 있어 브릿지에서 블록체인으로 송신하지 못한 경우, 브릿지에서 리턴되는 값은 MQRC 값(보통 **MQRC_FORMAT_ERROR**)입니다.

V 9.0.4 IBM MQ Bridge to blockchain 클라이언트 샘플 실행

IBM MQ Bridge to blockchain에서 제공하는 JMS 클라이언트 샘플을 사용하여 블록체인 브릿지가 검사하는 메시지를 입력 큐에 넣고 수신되는 응답을 확인할 수 있습니다.

시작하기 전에

IBM MQ Bridge to blockchain이 실행 중이고 IBM MQ Advanced 큐 관리자 및 블록체인 네트워크에 연결되었으며 입력 메시지를 처리할 준비가 되었습니다.

이 태스크 정보

IBM MQ Bridge to blockchain의 samp 디렉토리에서 JMS 샘플 애플리케이션을 찾으십시오.

프로시저

1. 클라이언트 샘플 Java 소스 파일을 편집하십시오.

   샘플 지시사항에 따라 사용 중인 IBM MQ 환경 및 블록체인 네트워크와 일치하도록 샘플을 구성하십시오. 샘플의 다음 코드는 브릿지로 보낼 JSON 요청 메시지를 정의합니다.

```
       // Create the JSON request message.
       // Modify "query", "exampleBlockchainChannelName", and "exampleChaincodeName" to
       // match your deployed blockchain chaincode.
       // The "operation" field is optional, but recommended. It should be set to QUERY
       // or UPDATE to match what the chaincode is going to do.

      JSONObject inputMsg = new JSONObject();
       inputMsg.put("operation", "QUERY");

      inputMsg.put("function", "query");
       inputMsg.put("channel", "exampleBlockchainChannelName");
       inputMsg.put("chaincodeName","exampleChaincodeName");


      // Create the JSON arguments for the request message.
       // Modify "a" to match your deployed blockchain chaincode
       // requirements, and add further arguments as necessary

      JSONArray myArgs = new JSONArray();
       myArgs.add("a");
       inputMsg.put("args", myArgs);


       TextMessage message = session.createTextMessage(inputMsg.serialize());
       message.setJMSReplyTo(replyToQueue);
```

2. 샘플을 컴파일하십시오.

   브릿지 디렉토리에 제공된 IBM MQ 클라이언트 클래스 및 JSON4j.jar 파일을 지정하십시오.

```
javac -cp $MQ_JAVA_INSTALL_PATH/lib/*:../prereqs/JSON4J.jar SimpleBCBClient.java
```

3. 컴파일된 클래스를 실행하십시오.

```
java -cp $MQ_JAVA_INSTALL_PATH/lib/*:../prereqs/JSON4J.jar:. SimpleBCBClient
```

```

Starting Simple MQ Blockchain Bridge Client
```

```
Created the message. Starting the connection
Sent message:

 JMSMessage class: jms_text
  JMSType:          null
  JMSDeliveryMode:  2
  JMSDeliveryDelay: 0
  JMSDeliveryTime:  1508427559117
  JMSExpiration:    0
  JMSPriority:      4
  JMSMessageID:     ID:414d5120424342514d202020202020209063e859ea36aa24
  JMSTimestamp:     1508427559117
  JMSCorrelationID: null
  JMSDestination:   queue:///APPL1.BLOCKCHAIN.INPUT.QUEUE
  JMSReplyTo:       queue:///APPL1.BLOCKCHAIN.REPLY.QUEUE
  JMSRedelivered:   false
    JMSXAppID: java
   JMSXDeliveryCount: 0
    JMSXUserID: USER1
   JMS_IBM_PutApplType: 6
    JMS_IBM_PutDate: 20171019
    JMS_IBM_PutTime: 15391912
{"args":
["a"],"function":"query","channel":"exampleBlockchainChannelName","operation":"QUERY","chainc
odeName":"exampleChaincodeName"}
```

응답 메시지:

```

  JMSMessage class: jms_text
  JMSType:          null
  JMSDeliveryMode:  1
  JMSDeliveryDelay: 0
  JMSDeliveryTime:  0
  JMSExpiration:    0
  JMSPriority:      4
  JMSMessageID:     ID:c3e2d840e2e2f0f84040404040404040d2afa27229838af2
  JMSTimestamp:     1497439784000
  JMSCorrelationID: ID:414d5120424342514d202020202020209063e859ea36aa24  *(JMSMessageID of
the input message)
  JMSDestination:   null
  JMSReplyTo:       null
  JMSRedelivered:   false
    JMSXAppID: java
    JMSXDeliveryCount: 1
    JMSXUserID: USER1
    JMS_IBM_Character_Set: UTF-8
    JMS_IBM_Encoding: 273
    JMS_IBM_Format: MQSTR
    JMS_IBM_MsgType: 8
    JMS_IBM_PutApplType: 2
    JMS_IBM_PutDate: 20171019
    JMS_IBM_PutTime: 15392014
{
   "data": "20",
   "message": "OK",
   "statusCode": 200,
   "statusType": "SUCCESS"
}
Response text:
{
   "data": "20",
   "message": "OK",
   "statusCode": 200,
   "statusType": "SUCCESS"
}
SUCCESS
```

클라이언트가 응답을 기다리는 동안 제한시간 초과 오류를 수신할 경우 브릿지가 실행 중인지 확인하십시오.

z/OS z/OS에서 큐 관리자 구성

IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

시작하기 전에

IBM MQ을(를) 구성하기 전에 IBM MQ for z/OS 개념의 IBM MQ for z/OS 개념에 대해 읽으십시오.

z/OS z/OS의 IBM MQ 환경 계획에서 IBM MQ for z/OS 환경을 계획하는 방법에 대해 읽으십시오.

이 태스크 정보

IBM MQ를 설치한 후 여러 태스크를 수행해야 사용자들이 사용할 수 있습니다.

프로시저

- IBM MQ for z/OS에서 큐 관리자를 구성하는 방법은 다음 하위 주제를 참조하십시오.

관련 태스크

5 페이지의 『Multiplatforms에서 큐 관리자 작성 및 관리』
메시지 및 큐를 사용하려면 먼저 최소한 하나의 큐 관리자 및 연관된 오브젝트를 작성하고 시작해야 합니다. 큐 관리자는 연관된 자원, 특히 큐에서 소유하는 자원을 관리합니다. 큐 관리자는 IBM MQ 오브젝트 작성, 수정, 표시, 삭제를 위한 메시지 큐잉 인터페이스(MQI) 호출 및 명령에 대한 큐잉 서비스를 애플리케이션에 제공합니다.

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

14 페이지의 『서버와 클라이언트 간의 연결 구성』
IBM MQ MQI clients와 서버 간 통신을 구성하려면 통신 프로토콜을 결정하고 링크 양 끝에서 연결을 정의하고 리스너를 시작하고 채널을 정의하십시오.

관련 정보

IBM MQ for z/OS 개념
보안 설정
z/OS IBM MQ for z/OS 관리
계획 중
z/OS 명령 실행
z/OS IBM MQ for z/OS 유틸리티

z/OS z/OS에서 큐 관리자 사용자 정의 준비

설치 가능한 기능, 자국어(NL) 기능, 보안 테스트 및 설정 정보의 세부사항으로 큐 관리자를 사용자 정의할 때 이 주제를 사용하십시오.

사용자 정의 준비

프로그램 디렉토리는 IBM MQ 설치 테이프의 컨텐츠, IBM MQ에 대한 프로그램 및 서비스 레벨 정보를 나열하고, SMP/E(System Modification Program Extended)를 사용하여 IBM MQ for z/OS를 설치하는 방법을 설명합니다. IBM MQ for z/OS 의 프로그램 디렉토리는 IBM 서적 센터 에서 다운로드할 수 있습니다 (IBM MQ 9.0 PDF 문서참조).

IBM MQ를 설치했을 때, 사용자들에게 사용 가능하게 만들기 전에 많은 태스크를 수행해야 합니다. 이러한 태스크에 대한 설명은 다음 절을 참조하십시오.

- 597 페이지의 『IBM MQ for z/OS 설정』
- 649 페이지의 『z/OS에서 큐 관리자 테스트』
- z/OS에서 보안 설정

IBM MQ for z/OS의 이전 버전에서 마이그레이션하는 중인 경우, 대부분의 사용자 정의 태스크를 수행할 필요가 없습니다. 수행해야 하는 태스크에 관한 자세한 정보는 유지보수 및 마이그레이션을 참조하십시오.

IBM MQ for z/OS의 설치 가능 기능

IBM MQ for z/OS는 다음 기능으로 구성됩니다.

기본
: 필수입니다. 모든 기본 기능을 포함하며 다음과 같습니다.

- 관리 및 유틸리티
- CICS, IMS, IBM MQ Application Programming Interface 또는 C++을 사용한 배치 유형 애플리케이션에 대한 지원
- 분산 큐잉 기능(TCP/IP 및 APPC 통신 둘 다 지원)

자국어(NL) 기능
: 지원되는 모든 자국어(NL)로 된 오류 메시지 및 패널을 포함합니다. 각 언어에는 언어와 연관된 문자가 있습니다. 언어 및 문자는 다음과 같습니다.

C
: 중국어

E
: U.S. 영어(대소문자 혼용)

F
: 프랑스어

K
: 일본어

U
: U.S. 영어(대문자)

미국 영어(대소문자 혼용) 옵션을 설치해야 합니다. 또한 하나 이상의 다른 언어를 설치할 수 있습니다. 다른 언어의 설치 프로세스를 수행하려면 미국 영어(대소문자 혼용)를 사용하지 않는 경우에도 미국 영어(대소문자 혼용)을 설치해야 합니다.

IBM MQ for z/OS Unix 시스템 서비스 컴포넌트
: 이 기능은 선택사항입니다. Java Message Service (JMS) 를 사용하여 IBM MQ for z/OS 에 연결하는 Java 애플리케이션을 빌드하고 실행하거나 HTTP를 사용하여 IBM MQ for z/OS에 연결하는 HTTP 애플리케이션을 빌드하고 실행하려는 경우 이 기능을 선택하십시오.

V 9.0.1 **IBM MQ for z/OS Unix System Services 웹 컴포넌트**
: 이 기능은 선택사항입니다.

IBM MQ Console 또는 REST API를 사용하려면 이 기능을 선택하십시오.

이 기능을 설치하려면 IBM MQ for z/OS Unix System Services 컴포넌트 기능을 설치해야 합니다.

설치 후 존재하는 라이브러리

IBM MQ는 많은 개별 로드 라이브러리와 함께 제공됩니다. 594 페이지의 표 38은 IBM MQ를 설치한 후 존재할 수 있는 라이브러리를 표시합니다.

표 *38.* 설치 후 존재하는 *IBM MQ* 라이브러리

이름	설명
thlqual.SCSQANLC	IBM MQ의 중국어 버전에 대한 로드 모듈을 포함합니다.
thlqual.SCSQANLE	영어의 로드 모듈을 포함합니다. IBM MQ의 영어(대소문자 혼용) 버전입니다.
thlqual.SCSQANLF	IBM MQ의 프랑스어 버전에 대한 로드 모듈을 포함합니다.
thlqual.SCSQANLK	IBM MQ의 일본어 버전에 대한 로드 모듈을 포함합니다.
thlqual.SCSQANLU	영어의 로드 모듈을 포함합니다. IBM MQ의 영어(대문자) 버전입니다.
thlqual.SCSQASMS	어셈블러 샘플 프로그램에 대한 소스를 포함합니다.

표 38. 설치 후 존재하는 IBM MQ 라이브러리 (계속)	
이름	**설명**
thlqual.SCSQAUTH	모든 IBM MQ 제품 로드 모듈에 대한 기본 저장소입니다. 기본 매개변수 모듈인 CSQZPARM도 포함합니다. 이 라이브러리는 APF 인증되고 PDS-E 형식이어야 합니다.
thlqual.SCSQCICS	CICS DFHRPL 연결에 포함되어야 하는 추가 로드 모듈을 포함합니다. 이 라이브러리는 APF 인증되고 PDS-E 형식이어야 합니다.
thlqual.SCSQCLST	샘플 프로그램이 사용하는 CLIST를 포함합니다.
thlqual.SCSQCOBC	샘플 프로그램에 필요한 카피북을 포함한 COBOL 카피북을 포함합니다.
thlqual.SCSQCOBS	COBOL 샘플 프로그램의 소스를 포함합니다.
thlqual.SCSQCPPS	C++ 샘플 프로그램의 소스를 포함합니다.
thlqual.SCSQC37S	C 샘플 프로그램에 대한 소스를 포함합니다.
thlqual.SCSQC370	샘플 프로그램에 필요한 헤더를 포함하여 C 헤더를 포함합니다.
thlqual.SCSQDEFS	공유 큐잉에 대해 C++ 및 Db2 DBRM용 사이드 정의를 포함합니다.
thlqual.SCSQEXEC	IBM MQ 운영 및 제어판을 사용 중인 경우 SYSEXEC 또는 SYSPROC 연결에 포함될 REXX 실행 가능 파일을 포함합니다.
thlqual.SCSQHPPS	C++용 헤더 파일을 포함합니다.
thlqual.SCSQINST	설치 작업에 대한 JCL을 포함합니다.
thlqual.SCSQLINK	초기 코드 라이브러리입니다. 시스템 초기 프로그램 로드(IPL)에서 로드되는 로드 모듈을 포함합니다. 라이브러리는 APF 승인을 받아야 합니다.
thlqual.SCSQLOAD	로드 라이브러리입니다. 비APF 코드, 사용자 엑시트, 유틸리티, 샘플, 설치 확인 프로그램 및 어댑터 스텁에 대한 로드 모듈을 포함합니다. 라이브러리는 APF 권한이 부여되지 않아도 되며 링크 목록에 없어도 됩니다. 이 라이브러리는 PDS-E 형식이어야 합니다.
thlqual.SCSQMACS	샘플 매크로, 제품 매크로 및 시스템 매개변수 매크로를 포함하여 어셈블러 매크로를 포함합니다.
thlqual.SCSQMAPS	샘플 프로그램에서 사용하는 CICS 맵 세트를 포함합니다.
thlqual.SCSQMSGC	IBM MQ 운영 및 제어판에서 중국어 언어 기능을 사용하는 경우 ISPMLIB 연결에 포함할 ISPF 메시지를 포함합니다.
thlqual.SCSQMSGE	영어를 사용하는 경우 ISPMLIB 연결에 포함될 ISPF 메시지를 포함합니다. IBM MQ 조작 및 제어판에 대한 영어(대소문자 혼용) 언어 기능입니다.
thlqual.SCSQMSGF	IBM MQ 운영 및 제어판에서 프랑스어 언어 기능을 사용하는 경우 ISPMLIB 연결에 포함할 ISPF 메시지를 포함합니다.
thlqual.SCSQMSGK	IBM MQ 운영 및 제어판에서 일본어 언어 기능을 사용하는 경우 ISPMLIB 연결에 포함할 ISPF 메시지를 포함합니다.
thlqual.SCSQMSGU	영어를 사용하는 경우 ISPMLIB 연결에 포함될 ISPF 메시지를 포함합니다. IBM MQ 조작 및 제어판에 대한 영어(대문자) 언어 기능입니다.
thlqual.SCSQMVR1	분산 큐잉에 대한 로드 모듈을 포함합니다. 이 라이브러리는 APF 인증되고 PDS-E 형식이어야 합니다.
thlqual.SCSQPLIC	PL/I 포함 파일을 포함합니다.

표 38. 설치 후 존재하는 *IBM MQ* 라이브러리 (계속)	
이름	**설명**
thlqual.SCSQPLIS	PL/I 샘플 프로그램에 대한 소스를 포함합니다.
thlqual.SCSQPNLA	ISPPLIB 연결에 포함할 덤프 형식기에 대한 IPCS 패널을 포함합니다. 또한 IBM MQ 샘플 프로그램에 대한 패널도 포함합니다.
thlqual.SCSQPNLC	IBM MQ 운영 및 제어판에서 중국어 언어 기능을 사용하는 경우 ISPPLIB 연결에 포함할 ISPF 패널을 포함합니다.
thlqual.SCSQPNLE	영어를 사용하는 경우 ISPPLIB 연결에 포함시킬 ISPF 패널을 포함합니다. IBM MQ 조작 및 제어판에 대한 영어(대소문자 혼용) 언어 기능입니다.
thlqual.SCSQPNLF	IBM MQ 운영 및 제어판에 대한 프랑스어 언어 기능을 사용 중인 경우 ISPPLIB 연결에 포함될 ISPF 패널을 포함합니다.
thlqual.SCSQPNLK	IBM MQ 운영 및 제어판에서 일본어 언어 기능을 사용하는 경우 ISPPLIB 연결에 포함할 ISPF 패널을 포함합니다.
thlqual.SCSQPNLU	영어를 사용하는 경우 ISPPLIB 연결에 포함시킬 ISPF 패널을 포함합니다. IBM MQ 조작 및 제어판에 대한 영어(대문자) 언어 기능입니다.
thlqual.SCSQPROC	샘플 JCL 및 기본 시스템 초기화 데이터 세트를 포함합니다.
thlqual.SCSQSNLC	특수 용도 기능(예: 초기 코드)에 필요한 IBM MQ 모듈의 중국어 버전에 대한 로드 모듈을 포함합니다.
thlqual.SCSQSNLE	영어의 로드 모듈을 포함합니다. 특수 목적 기능(예: 초기 코드)에 필요한 IBM MQ 모듈의 영어(대소문자 혼용) 버전.
thlqual.SCSQSNLF	특수 용도 기능(예: 초기 코드)에 필요한 IBM MQ 모듈의 프랑스어 버전에 대한 로드 모듈을 포함합니다.
thlqual.SCSQSNLK	특수 용도 기능(예: 초기 코드)에 필요한 IBM MQ 모듈의 일본어 버전에 대한 로드 모듈을 포함합니다.
thlqual.SCSQSNLU	영어의 로드 모듈을 포함합니다. 특수 목적 기능(예: 초기 코드)에 필요한 IBM MQ 모듈의 영어(대문자) 버전.
thlqual.SCSQTBLC	IBM MQ 운영 및 제어판에서 중국어 언어 기능을 사용하는 경우 ISPTLIB 연결에 포함할 ISPF 테이블을 포함합니다.
thlqual.SCSQTBLE	영어를 사용하는 경우 ISPTLIB 연결에 포함시킬 ISPF 테이블을 포함합니다. IBM MQ 조작 및 제어판에 대한 영어(대소문자 혼용) 언어 기능입니다.
thlqual.SCSQTBLF	IBM MQ 운영 및 제어판에 대한 프랑스어 언어 기능을 사용 중인 경우 ISPTLIB 연결에 포함될 ISPF 패널을 포함합니다.
thlqual.SCSQTBLK	IBM MQ 운영 및 제어판에서 일본어 언어 기능을 사용하는 경우 ISPTLIB 연결에 포함할 ISPF 테이블을 포함합니다.
thlqual.SCSQTBLU	영어를 사용하는 경우 ISPTLIB 연결에 포함시킬 ISPF 테이블을 포함합니다. IBM MQ 조작 및 제어판에 대한 영어(대문자) 언어 기능입니다.

참고: 이러한 라이브러리를 수정하거나 사용자 정의하지 마십시오. 변경하려면 라이브러리를 복사하고 사본에서 변경을 수행하십시오.

관련 개념

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

683 페이지의 『IMS와 함께 IBM MQ 사용』
IBM MQ -IMS 어댑터 및 IBM MQ - IMS 브릿지는 IBM MQ가 IMS와 상호작용할 수 있도록 허용하는 두 컴포넌트입니다.

691 페이지의 『CICS와 함께 IBM MQ 사용』
CICS와 함께 IBM MQ를 사용하려면 IBM MQ CICS 어댑터 및 IBM MQ CICS bridge 컴포넌트를 구성해야 합니다.

693 페이지의 『IMS에서 OTMA 엑시트 사용』
IBM MQ for z/OS와 함께 IMS OTMA(Open Transaction Manager Access) 엑시트를 사용하려는 경우 이 토픽을 사용하십시오.

관련 참조

691 페이지의 『Language Environment 또는 z/OS 호출 가능 서비스 업그레이드 및 서비스 적용』
수행해야 하는 조치는 CALLLIBS 또는 LINK 중 사용하는 항목과 SMP/E의 버전에 따라 달라집니다.

관련 정보

IBM MQ for z/OS 개념
IBM MQ for z/OS 관리

z/OS IBM MQ for z/OS 설정

이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

큐 관리자를 구성하는 가장 좋은 방법은 표시된 순서대로 다음 단계를 수행하는 것입니다.

1. 기본 큐 관리자를 구성하십시오.
2. 큐 관리자 간 통신 및 원격 클라이언트 애플리케이션 통신을 수행하는 채널 시작기를 구성하십시오.
3. 메시지를 암호화하거나 보호하려면 Advanced Message Security for z/OS를 구성하십시오.
4. IBM MQ를 사용하여 파일을 전송하려면 Managed File Transfer for z/OS를 구성하십시오.
5. 관리 또는 메시징 REST API나 MQ Console을 사용하여 웹 브라우저에서 IBM MQ를 관리하려면 mqweb 서버를 구성하십시오.

이 주제에서는 IBM MQ를 설치한 후 설정하는 다양한 단계를 안내합니다. 설치 프로세스는 프로그램 디렉토리에 설명되어 있습니다. IBM MQ for z/OS 의 프로그램 디렉토리는 IBM 서적 센터 에서 다운로드할 수 있습니다 (IBM MQ 9.0 PDF 문서참조).

사용자 정의를 돕기 위해 IBM MQ와 함께 샘플이 제공됩니다. 샘플 데이터 세트 멤버는 이름이 CSQ4라는 네 자로 시작하며 이는 라이브러리 thlqual.SCSQPROC에 있습니다.

IBM MQ for z/OS의 성능 및 자원 요구사항에 영향을 주기 때문에, 이 주제에서 설명한 사용자 정의 태스크를 수행하기 전에 고려해야 할 여러 구성 옵션이 있습니다. 예를 들어 사용하려는 글로벌화 라이브러리를 결정해야 합니다.

사용자 정의 단계 중 일부를 자동화하려면 697 페이지의 『IBM z/OSMF를 사용하여 IBM MQ 자동화』의 내용을 참조하십시오.

구성 옵션

이러한 옵션에 대한 자세한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

이 절의 각 태스크에 대한 설명은 다음 여부를 표시합니다.

- 태스크는 IBM MQ를 설정하는 프로세스의 부분입니다. 즉, z/OS 시스템에서 IBM MQ를 사용자 정의할 때 태스크를 한 번 수행합니다. (병렬 sysplex에서는 sysplex의 각 z/OS 시스템에 대한 태스크를 수행하고 각 z/OS 시스템이 동일하게 설정되는지 확인해야 합니다.)
- 태스크는 큐 관리자 추가의 일부입니다. 즉, 해당 큐 관리자를 추가할 때 각 큐 관리자에 대해 태스크를 한 번 수행합니다.

명령을 사용하여 다양한 z/OS 시스템 매개변수를 변경하고 제안대로 608 페이지의 『SYS1.PARMLIB 멤버 업데이트』을 수행하는 경우, z/OS 시스템의 IPL을 수행해야 하는 태스크는 없습니다.

조작을 단순화하고 문제점 판별을 지원하려면, 큐 관리자가 시스템에서 긴급 상황으로 빠르게 작성할 수 있도록 SYSPLEX의 모든 z/OS 시스템이 동일하게 설정되었는지 확인하십시오.

유지보수 편의성을 위해 IBM MQ 라이브러리를 참조할 알리어스 정의를 고려하십시오. IBM MQ 라이브러리를 참조할 알리어스 사용을 참조하십시오.

관련 개념

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

683 페이지의 『IMS와 함께 IBM MQ 사용』
IBM MQ -IMS 어댑터 및 IBM MQ - IMS 브릿지는 IBM MQ가 IMS와 상호작용할 수 있도록 허용하는 두 컴포넌트입니다.

691 페이지의 『CICS와 함께 IBM MQ 사용』
CICS와 함께 IBM MQ를 사용하려면 IBM MQ CICS 어댑터 및 IBM MQ CICS bridge 컴포넌트를 구성해야 합니다.

693 페이지의 『IMS에서 OTMA 엑시트 사용』
IBM MQ for z/OS와 함께 IMS OTMA(Open Transaction Manager Access) 엑시트를 사용하려는 경우 이 토픽을 사용하십시오.

관련 참조

691 페이지의 『Language Environment 또는 z/OS 호출 가능 서비스 업그레이드 및 서비스 적용』
수행해야 하는 조치는 CALLLIBS 또는 LINK 중 사용하는 항목과 SMP/E의 버전에 따라 달라집니다.

관련 정보

IBM MQ for z/OS 개념
IBM MQ for z/OS 관리
IBM MQ for z/OS의 프로그램 디렉토리

z/OS IBM MQ에 대한 z/OS 시스템 구성

이러한 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용합니다.

z/OS *z/OS* 시스템 매개변수 식별

태스크의 일부는 z/OS 시스템 매개변수 업데이트를 포함합니다. 시스템 IPL이 수행될 때 지정된 것을 알아야 합니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 한 번씩 이 태스크를 수행해야 합니다.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.

SYS1.PARMLIB(IEASYSpp)에는 SYS1.PARMLIB(여기서 pp는 시스템의 IPL을 수행하는 데 사용된 z/OS 시스템 매개변수 목록을 나타냄)의 다른 멤버를 가리키는 매개변수의 목록이 들어 있습니다.

찾아야 하는 항목은 다음과 같습니다.

598 페이지의 『APF에서 IBM MQ 로드 라이브러리에 권한 부여』의 경우:
: PROG=xx 또는 APF=aa가 APF(Authorized Program Facility) 권한 부여된 라이브러리 목록을 가리킵니다(멤버 PROGxx 또는 IEFAPFaa).

600 페이지의 『z/OS 링크 목록 및 LPA 업데이트』의 경우:
: LNK=kk가 링크 목록을 가리키고(멤버 LNKLSTkk), LPA=mm은 LPA 목록을 가리킵니다(멤버 LPALSTmm).

601 페이지의 『z/OS 프로그램 특성 테이블 업데이트』의 경우:
: SCH=xx는 프로그램 특성 테이블(PPT)을 가리킵니다(멤버 SCHEDxx).

602 페이지의 『z/OS에 대해 IBM MQ 서브시스템 정의』의 경우:
: SSN=ss는 정의된 서브시스템 목록을 가리킵니다(멤버 IEFSSNss).

z/OS *APF*에서 *IBM MQ* 로드 라이브러리에 권한 부여

다양한 라이브러리를 APF 권한 부여합니다. 일부 로드 모듈은 이미 권한이 부여되었을 수 있습니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 한 번씩 이 태스크를 수행해야 합니다.

- 큐 공유 그룹을 사용 중인 경우 *IBM MQ* 설정이 *sysplex*에 있는 각 *z/OS* 시스템에서 동일한지 확인해야 합니다.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.
- *LLA(Library Look Aside)* 사용:
  - 일부 *IBM MQ* 사용으로 인해 라이브러리의 로드 모듈에 입출력*(I/O)*이 많아질 수 있습니다. 이 *IO*는 운영 체제의 *LLA* 기능을 사용하여 줄일 수 있습니다.
  - 다음과 같은 경우에 *IO*가 많아질 수 있습니다.
    - 높은 *MQCONN/MQDISC* 비율의 애플리케이션(예: *WLM* 스토어드 프로시저에서).
    - 채널 엑시트 로드. 채널을 자주 시작하고 중지하며 채널 엑시트를 사용하는 경우.
  - *SYS1.PARMLIB*의 *CSVLLAxx* 멤버는 *LLA* 설정을 지정합니다. *LIBRARIES* 문의 라이브러리 이름 포함은 항상 *VLF(Virtual Lookaside Facility)*에서 프로그램 사본을 가져오므로 많이 사용되는 경우 대개 *I/O*가 필요하지 않음을 의미합니다.

    FREEZE 문의 포함은 관련 *DD* 문 연결 디렉토리를 가져오는 *I/O*가 없음(종종 프로그램 로드 자체보다 *I/O*가 많을 수 있음)을 의미합니다.

    이러한 라이브러리 중 하나를 변경한 후에는 운영 체제 명령 “ *F LLA,REFRESH*”를 사용하십시오.

IBM MQ 로드 라이브러리 thlqual.SCSQAUTH 및 thlqual.SCSQLINK는 APF 권한이 있어야 합니다. 자국어(NL) 기능(thlqual.SCSQANLx 및 thlqual.SCSQSNLx)과 분산 큐 기능(thlqual.SCSQMVR1)에 대해서도 APF 권한을 부여해야 합니다. Advanced Message Security를 사용 중인 경우 thlqual.SDRQAUTH 라이브러리도 APF 권한 부여해야 합니다.

그러나 LPA의 모든 로드 모듈은 자동으로 APF 권한이 인증됩니다. 따라서 SYS1.PARMLIB 멤버 IEASYSpp에 다음 명령문이 포함된 경우 링크 목록의 모든 멤버가 됩니다.

```
LNKAUTH=LNKLST
```

LNKAUTH가 지정되지 않은 경우, LNKAUTH=LNKLST가 기본값입니다.

LPA 또는 링크 목록에 넣기 위해 선택하는 항목에 따라(600 페이지의 『z/OS 링크 목록 및 LPA 업데이트』 참조) 라이브러리를 APF 링크 목록에 넣을 필요가 없을 수 있습니다.

참고: IBM MQ STEPLIB에 포함하는 모든 라이브러리에 대해 APF 권한을 부여해야 합니다. STEPLIB에 APF 권한이 부여되지 않은 라이브러리를 넣으면, 전체 라이브러리 연결이 APF 인증을 잃게 됩니다.

APF 목록은 SYS1.PARMLIB 멤버 PROGxx 또는 IEAAPFaa에 있습니다. 목록에는 APF 권한이 부여된 z/OS 라이브러리의 이름이 들어 있습니다. 목록의 항목 순서는 중요하지 않습니다. APF 목록에 대한 정보는 *z/OS MVS Initialization and Tuning Reference* 매뉴얼을 참조하십시오.

시스템 성능 조정에 대한 자세한 정보는 SupportPac MP16을 참조하십시오.

PROGxx 멤버를 동적 형식으로 사용하는 경우, 변경사항을 적용하려면 z/OS 명령 `SETPROG APF,ADD,DSNAME=hlq.SCSQ` *XXXX* `,VOLUME=` *YYYYYY* 만 실행해야 합니다. 여기서 *XXXX* 는 라이브러리 이름에 따라 다릅니다. 여기서 *YYYYY* 는 볼륨입니다. 그렇지 않으면 정적 형식 또는 IEAAPFaa 멤버를 사용할 경우, 사용자 시스템에서 IPL을 수행해야 합니다.

APF 목록에서 실제 라이브러리 이름을 사용해야 합니다. 라이브러리의 데이터 세트 알리어스를 사용하려고 하면 권한 부여에 실패합니다.

관련 개념

600 페이지의 『z/OS 링크 목록 및 LPA 업데이트』
LPA 라이브러리를 초기 코드 라이브러리의 새 버전으로 업데이트하십시오. 기타 코드는 링크 목록이나 LPA로 이동할 수 있습니다.

593 페이지의 『z/OS에서 큐 관리자 사용자 정의 준비』
설치 가능한 기능, 자국어(NL) 기능, 보안 테스트 및 설정 정보의 세부사항으로 큐 관리자를 사용자 정의할 때 이 주제를 사용하십시오.

z/OS *z/OS* 링크 목록 및 *LPA* 업데이트

LPA 라이브러리를 초기 코드 라이브러리의 새 버전으로 업데이트하십시오. 기타 코드는 링크 목록이나 LPA로 이동할 수 있습니다.

- IBM MQ를 실행하려는 각 z/OS 시스템에 대해 이 태스크를 한 번 수행해야 합니다.
- 큐 공유 그룹을 사용하는 경우, 큐 관리자를 IBM MQ 9.0로 마이그레이션하기 전에 QSG의 각 큐 관리자에서 IBM MQ 9.0 레벨로 초기 코드를 새로 고쳐야 합니다.

  각 LPAR에서 최신 초기 코드를 설치한 다음 마이그레이션하기 전 일정 시점에 한 번에 하나의 큐 관리자를 새로 고치십시오. 동시에 모든 큐 관리자를 마이그레이션하지 않아도 됩니다.
- 이전 버전에서 마이그레이션 시 이 태스크를 수행해야 할 수도 있습니다. 자세한 내용은 프로그램 디렉토리 IBM MQ for z/OS 의 프로그램 디렉토리는 IBM 서적 센터 에서 다운로드할 수 있습니다 (IBM MQ 9.0 PDF 문서참조).를 참조하십시오.

참고: LPA용 데이터 세트는 버전에 따라 다릅니다. 시스템의 기존 LPA를 사용 중인 경우 시스템 관리자에게 문의하여 사용할 LPA를 결정하십시오.

초기 코드

IBM MQ가 서브시스템의 역할을 할 수 있도록 하려면 일부 IBM MQ 로드 모듈을 MVS에 추가해야 합니다. 이러한 모듈을 초기 코드라 하며 이는 큐 관리자가 활성 상태가 아니더라도 실행할 수 있습니다. 예를 들어 운영자 명령이 IBM MQ 명령 접두부를 갖고 콘솔에서 실행되는 경우, 이 초기 코드가 제어를 얻고 큐 관리자를 시작할지 아니면 실행 중인 큐 관리자로 요청을 전달해야 하는지 확인합니다. 이 코드는 LPA(Link Pack Area)에 로드됩니다. 초기 모듈은 한 세트로, 모든 큐 관리자에 사용되며 이들은 IBM MQ의 최상위 레벨에 있어야 합니다. 더 높은 버전의 IBM MQ의 초기 코드는 IBM MQ의 더 낮은 버전으로 큐 관리자를 사용하며 그 반대는 성립되지 않습니다.

초기 코드는 다음 로드 모듈을 포함합니다.

- thqual.SCSQLINK 라이브러리의 CSQ3INI 및 CSQ3EPX
- thqual.SCSQSNL *x*의 CSQ3ECMX. 여기서 *x*는 사용자의 언어 문자입니다.
  - thlqual.SCSQSNLE(미국 영어 혼합 대소문자)
  - thlqual.SCSQSNLU(미국 영어 대문자)
  - thlqual.SCSQSNLK(일본어)
  - thlqual.SCSQSNLF(프랑스어)
  - thlqual.SCSQSNLC(중국어)

IBM MQ에는 thqual.SCSQSNL *i* 라이브러리의 내용을 thqual.SCSQLINK로 이동시키고 SMP/E에 알리는 사용자 수정사항이 포함되어 있습니다. 이 사용자 수정은 CSQ8UERL 이라고 하며 IBM 서적 센터에서 다운로드할 수 있는 Long Term Support 또는 Continuous Delivery의 *Program Directory for IBM MQ for z/OS*에 설명되어 있습니다.

LPA 라이브러리에서 초기 코드를 업데이트한 경우, 다음 z/OS IPL(CLPA 옵션 포함)에서부터 SYS1.PARMLIB의 IEFSSNss 멤버의 정의로부터 IPL 동안 추가된 모든 큐 관리자 서브시스템에 이르기까지 사용 가능합니다.

다음과 같이 LPA에 추가하여 나중에 추가되는 모든 새 큐 관리자 서브시스템에 대해 IPL 없이 즉시 사용 가능하게 만들 수 있습니다(602 페이지의 『z/OS에 대해 IBM MQ 서브시스템 정의』에서 설명함).

- CSQ8UERL을 사용하지 않은 경우, 다음 z/OS 명령을 실행하십시오.

```
SETPROG LPA,ADD,MODNAME=(CSQ3INI,CSQ3EPX),DSNAME=thqual.SCSQLINK
SETPROG LPA,ADD,MODNAME=(CSQ3ECMX),DSNAME=thqual.SCSQSNL x
```

- CSQ8UERL을 사용한 경우, 다음 z/OS 명령을 사용하여 LPA로 초기 코드를 로드할 수 있습니다.

```
SETPROG LPA,ADD,MASK=*,DSNAME=thqual.SCSQLINK
```

- Advanced Message Security를 사용하는 경우 다음 z/OS 명령을 사용하여 LPA에 추가 모듈을 포함해야 합니다.

```
SETPROG LPA,ADD,MODNAME=(CSQ0DRTM),DSNAME=thqual.SCSQLINK
```

유지보수를 적용했거나 나중 버전 또는 IBM MQ릴리스로 큐 관리자를 재시작하려는 경우, 이미 정의된 큐 관리자 서브시스템에서 초기 코드를 사용할 수 있습니다. 사용할 수 있도록 하려면 다음 단계를 사용하십시오.

1. 이 토픽의 앞에서 설명한 대로 z/OS SETPROG 명령을 사용하여 LPA에 이를 추가하십시오.
2. IBM MQ 명령 STOP QMGR을 사용하여 큐 관리자를 중지하십시오.
3. qmgr.REFRESH.QMGR 보안 프로파일이 설정되었는지 확인하십시오. MQSC 명령, 프로파일 및 해당 액세스 레벨을 참조하십시오.
4. IBM MQ 명령 REFRESH QMGR TYPE(EARLY)을 사용하여 큐 관리자의 초기 코드를 새로 고치십시오.
5. IBM MQ 명령 START QMGR을 사용하여 큐 관리자를 재시작하십시오.

IBM MQ 명령 STOP QMGR, REFRESH QMGR, START QMGR은 MQSC 명령에 설명되어 있습니다.

기타 코드

다음 라이브러리에 있는 모든 IBM MQ 제공 로드 모듈은 재진입되어 LPA에 배치될 수 있습니다.

- SCSQAUTH
- SCSQANL*x*, 여기서 *x*는 사용자의 언어 문자임
- SCSQMVR1

중요사항: 그러나, 이 라이브러리를 LPA에 배치하는 경우 유지보수를 적용할 때마다 모든 변경된 모듈을 수동으로 LPA에 복사해야 합니다. 따라서, IBM MQ 로드 라이브러리를 링크 목록에 넣는 것이 바람직한데, 이는 z/OS 명령 REFRESH LLA를 실행하여 유지보수 후에 업데이트될 수 있습니다.

이는 특히 일부 STEPLIB에 이를 포함할 필요가 없어지기 때문에 SCSQAUTH에 권장됩니다. 단 하나의 언어 라이브러리인 SCSQANL*x*만 LPA나 링크 목록에 위치해야 합니다. 링크 목록 라이브러리는 SYS1.PARMLIB의 LNKLSTkk 멤버에서 지정됩니다.

분산 큐잉 기능과 CICS bridge(큐 관리자 자체는 아님)는 LE(Language Environment) 런타임 라이브러리 SCEERUN에 대한 액세스 권한이 필요합니다. 이러한 기능 중 하나를 사용할 경우, 링크 목록에 SCEERUN을 포함해야 합니다.

관련 개념

601 페이지의 『z/OS 프로그램 특성 테이블 업데이트』
일부 추가 PPT 항목이 IBM MQ 큐 관리자를 위해 필요합니다.

z/OS *z/OS* 프로그램 특성 테이블 업데이트

일부 추가 PPT 항목이 IBM MQ 큐 관리자를 위해 필요합니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 이 태스크를 한 번 수행해야 합니다.
- 큐 공유 그룹을 사용 중인 경우 *IBM MQ* 설정이 *sysplex*에 있는 각 *z/OS* 시스템에서 동일한지 확인해야 합니다.
- 이전 버전에서 마이그레이션할 때는 이 태스크를 수행할 필요가 없습니다.
- *Advanced Message Security*가 필요한 경우 이 태스크의 *CSQ0DSRV* 파트를 수행해야 합니다.

필요한 모든 PPT 항목을 포함하는 샘플은 thlqual.SCSQPROC(CSQ4SCHD)에 제공됩니다. 필수 항목을 SYS1.PARMLIB(SCHEDxx)에 있는 PPT에 추가하도록 하십시오.

z/OS 1.12 이상 버전에서, CSQYASCP가 이미 상세한 속성과 함께 운영 체제에 정의되었었으며 더 이상 PARMLIB의 SCHEDxx 멤버에 포함될 필요가 없습니다.

IBM MQ 큐 관리자는 스와핑 자체를 제어합니다. 그러나 IBM MQ 네트워크가 과도하게 로드되었으며 응답 시간이 중요한 경우 CSQXJST PPT 항목을 추가하여 IBM MQ 채널 시작기가 스와핑되지 않도록 하면 유리할 수 있습니다. 단, 나머지 z/OS 시스템의 성능에 영향을 미칠 위험이 있습니다.

Advanced Message Security가 필요한 경우 CSQ0DSRV PPT 항목을 추가하십시오.

이들 변경이 적용되도록 z/OS 명령 SET SCH=를 실행하십시오.

관련 개념

602 페이지의 『z/OS에 대해 IBM MQ 서브시스템 정의』
서브시스템 이름 테이블을 업데이트하고 명령 접두부 문자열에 대한 규칙을 결정하십시오.

z/OS 큐 관리자 및 채널 시작기 구성

이러한 주제를 큐 관리자 및 채널 시작기 구성을 위한 단계별 안내서로 사용합니다.

z/OS *z/OS*에 대해 *IBM MQ* 서브시스템 정의

서브시스템 이름 테이블을 업데이트하고 명령 접두부 문자열에 대한 규칙을 결정하십시오.

각 IBM MQ 큐 관리자에 대해 이 태스크를 반복하십시오. 이전 버전에서 마이그레이션할 때는 이 태스크를 수행할 필요가 없습니다.

관련 개념

605 페이지의 『IBM MQ 큐 관리자에 대한 프로시저 작성』
각 IBM MQ 서브시스템은 큐 관리자를 시작하기 위한 카탈로깅된 프로시저가 필요합니다. 사용자 자신의 프로시저를 작성하거나 IBM이 제공하는 프로시저 라이브러리를 사용할 수 있습니다.

z/OS 서브시스템 이름 테이블 업데이트

IBM MQ 서브시스템을 정의할 때 서브시스템 이름 테이블에 항목을 추가해야 합니다.

초기에 SYS1.PARMLIB 멤버 IEFSSNss로부터 가져오는 z/OS의 서브시스템 이름 테이블에는 이전에 정의된 z/OS 서브시스템의 정의가 들어 있습니다. 각 IBM MQ 서브시스템을 정의하려면 SYS1.PARMLIB의 IEFSSNss 멤버를 변경하거나 가능하면 z/OS 명령 SETSSI를 사용하여 이 테이블에 항목을 추가해야 합니다.

IBM MQ 서브시스템 초기화는 병렬 처리를 지원하므로, IBM MQ 서브시스템 정의 명령문을 z/OS V1.12 이상에서 사용 가능한 IEFSSNss 테이블의 BEGINPARALLEL 키워드 위아래 모두에 추가할 수 있습니다.

SETSSI 명령을 사용하는 경우, 변경은 즉시 적용되고 시스템 IPL을 수행할 필요가 없습니다. 후속 IPL 후에 변경이 지속하도록 608 페이지의 『SYS1.PARMLIB 멤버 업데이트』에서 설명한 대로 SYS1.PARMLIB를 업데이트해야 합니다.

IBM MQ 서브시스템을 동적으로 정의하기 위한 SETSSI 명령은 다음과 같습니다.

```
SETSSI ADD,S=ssid,I=CSQ3INI,P='CSQ3EPX,cpf,scope'
```

IEFSSNss의 대응하는 정보는 다음 두 가지 방법 중 하나로 지정할 수 있습니다.

- IEFSSNss에서 IBM MQ 서브시스템 정의의 키워드 매개변수 양식. 이것이 권장되는 방법입니다.

```
SUBSYS SUBNAME(ssid) INITRTN(CSQ3INI) INITPARM('CSQ3EPX,cpf,scope')
```

- IBM MQ 서브시스템 정의의 위치 매개변수 양식.

```
ssid,CSQ3INI,'CSQ3EPX,cpf,scope'
```

한 IEFSSNss 멤버에서 두 양식을 혼합하지 마십시오. 다른 양식이 필요한 경우 IEASYSpp SYS1.PARMLIB 멤버에 새 멤버의 SSN 피연산자를 추가하여 각 유형에 대해 별도의 IEFSSNss 멤버를 사용하십시오. 두 개 이상의 SSN을 지정하려면 IEASYSpp에서 SSN=(aa,bb,...)을 사용하십시오.

예에서,

ssid
: 서브시스템 ID입니다. 최고 4문자까지 가능합니다. 모든 문자는 영숫자(대문자 A - Z, 0 - 9)여야 하며, 영문자로 시작해야 합니다. 큐 관리자는 서브시스템과 동일한 이름을 가지므로, z/OS 서브시스템 이름과 IBM MQ 오브젝트 이름 모두에 허용되는 문자만 사용할 수 있습니다.

cpf
: 명령 접두부 문자열(CPF에 대한 정보는 603 페이지의 『명령 접두부 문자열(CPF) 정의』의 내용을 참조하십시오).

scope
: z/OS sysplex에서 실행 중인 경우에 사용되는 시스템 범위입니다. (시스템 범위에 대한 정보는 604 페이지의 『SYSPLEX 환경의 CPF』의 내용을 참조하십시오.)

603 페이지의 그림 99은 IEFSSNss 명령문의 여러 가지 예를 보여줍니다.

```
CSQ1,CSQ3INI,'CSQ3EPX,+mqs1cpf,S'
CSQ2,CSQ3INI,'CSQ3EPX,+mqs2cpf,S'
CSQ3,CSQ3INI,'CSQ3EPX,++,S'
```

그림 *99.* 서브시스템 정의를 위한 샘플 *IEFSSNss* 명령문

참고: 서브시스템에서 오브젝트를 작성했을 때, 서브시스템 이름을 변경하거나 한 서브시스템에서 다른 서브시스템의 페이지 세트를 사용할 수 없습니다. 이 중 하나를 수행하려면 한 서브시스템의 모든 오브젝트 및 메시지를 로드 해제하고 다른 서브시스템으로 재로드해야 합니다.

603 페이지의 표 39은 603 페이지의 그림 99의 명령문에 의해 정의된 대로 서브시스템 이름과 명령 접두부 문자열(CPF)의 연관을 보여주는 몇 가지 예를 제공합니다.

표 *39.* 서브시스템 이름 대 *CPF* 연관

IBM MQ 서브시스템 이름	CPF
CSQ1	+mqs1cpf
CSQ2	+mqs2cpf
CSQ3	++

참고: z/OS 명령 SETSSI의 ACTIVATE 및 DEACTIVATE 함수는 IBM MQ에서 지원되지 않습니다.

변경사항의 상태를 확인하려면 SDSF: `/D SSI,L`에서 다음 명령을 실행하십시오. ACTIVE 상태를 갖고 작성된 새 서브시스템이 표시됩니다.

z/OS 명령 접두부 문자열*(CPF)* 정의

IBM MQ의 각 서브시스템 인스턴스는 해당 서브시스템을 식별할 명령 접두부 문자열을 가질 수 있습니다.

충돌되지 않도록 모든 서브시스템의 CPF에 관한 시스템 전체의 규칙을 채택하십시오. 다음 지침을 지키십시오.

- 최대 8자의 문자열로 CPF를 정의하십시오.
- 다른 서브시스템이 이미 사용 중인 CPF를 사용하지 말고 시스템에 정의된 JES 백스페이스 문자를 문자열의 첫 번째 문자로 사용하지 마십시오.
- 604 페이지의 표 41에 나열된 올바른 문자 세트의 문자를 사용하여 CPF를 정의하십시오.
- 이미 정의된 프로세스의 약어이거나 명령 구문과 혼동될 수 있는 CPF는 사용하지 마십시오. 예를 들어 'D' 같은 CPF는 DISPLAY 같은 z/OS 명령과 충돌합니다. 이것이 발생하는 것을 피하기 위해 특수 문자(604 페이지의 표 41에 표시됨) 중 하나를 CPF 문자열에서 첫 번째 또는 유일한 문자로 사용하십시오.
- 기존 CPF의 서브세트 또는 수퍼세트인 CPF를 정의하지 마십시오. 예에 대해서는 604 페이지의 표 40의 내용을 참조하십시오.

표 *40. CPF* 서브세트 및 수퍼세트 규칙의 예		
서브시스템 이름	**정의된 CPF**	**라우팅된 명령**
MQA	!A	MQA
MQB	!B	MQB
MQC1	!C1	MQC1
MQC2	!C2	MQC2
MQB1	!B1	MQB

서브시스템 MQB1용 명령(CPF 사용!B1)이 서브시스템에 대한 CPF가 !이기 때문에 서브시스템 MQB로 라우트 됩니다.B, !의 서브세트B1. 예를 들어 다음 명령을 입력한 경우

```
!B1  START  QMGR
```

서브시스템 MQB는 다음 명령을 수신합니다.

```
1 START QMGR
```

(이 경우에는 처리할 수 없습니다.)

z/OS 명령 DISPLAY OPDATA를 실행하여 존재하는 접두부를 확인할 수 있습니다.

Sysplex에서 실행 중인 경우, z/OS는 CPF 등록 시에 이 유형의 모드 충돌을 진단합니다. (CPF 등록에 관한 정보는 604 페이지의 『SYSPLEX 환경의 CPF』의 내용을 참조하십시오.)

604 페이지의 표 41는 CPF 문자열을 정의할 때 사용할 수 있는 문자를 표시합니다.

표 *41. CPF* 문자열의 올바른 문자 세트	
문자 세트	**컨텐츠**
영문자	대문자 A - Z, 소문자 a - z
숫자	0 - 9
자국어(참고 참조)	@ $ # (16진 값으로 표시될 수 있는 문자)
기타	. ⊠ ( ) * & + - = ¢ < | ! ; % _ ? : >

참고:

시스템은 자국 문자의 다음 16진 표시를 인식합니다. @는 X'7C', $는 X'5B', #은 X'7B'로 표시됩니다. U.S이외의 국가에서는 U.S. 터미널 키보드에 표시되는 자국 문자는 다른 16진 표시를 생성하여 오류를 유발할 수 있습니다. 예를 들어, 일부 국가에서는 $ 문자가 X'4A'를 생성할 수 있습니다.

세미콜론(;)은 CPF로서 올바르지만, 대부분의 시스템에서 이 문자는 명령 구분 기호입니다.

z/OS *SYSPLEX* 환경의 *CPF*

SYSPLEX 범위 내에서 CPF 사용 방법을 이해하려면 이 토픽을 사용하십시오.

SYSPLEX 환경에서 사용되는 경우 IBM MQ는 SYSPLEX의 임의 콘솔에서 명령을 입력하고 해당 명령을 실행을 위해 적합한 시스템으로 라우트할 수 있도록 CPF를 등록합니다. 명령 응답은 생성하는 콘솔로 리턴됩니다.

SYSPLEX 조작에 대한 범위 정의

범위가 SYSPLEX 환경에서 IBM MQ를 실행 중일 때 IBM MQ 서브시스템에 의해 수행되는 CPF 등록의 유형을 판별하는 데 사용됩니다.

범위에 대해 가능한 값은 다음과 같습니다.

M
: 시스템 범위.

  CPF는 IBM MQ에 의해 시스템 IPL 시에 z/OS에 등록되고 z/OS 시스템이 활성인 전체 시간 동안 등록된 상태로 있습니다.

  IBM MQ 명령은 대상 서브시스템을 실행하는 z/OS 이미지에 연결된 콘솔에 입력되어야 하며, 그렇지 않으면 ROUTE 명령을 사용하여 해당 이미지로 명령을 전달해야 합니다.

  SYSPLEX에서 실행 중이 아닌 경우 이 옵션을 사용하십시오.

S
: Sysplex가 범위를 시작했습니다.

  IBM MQ 서브시스템이 시작될 때 CPF가 z/OS에 등록되고, IBM MQ 서브시스템이 종료할 때까지 활성 상태로 있습니다.

  ROUTE 명령을 사용하여 원래 START QMGR 명령을 대상 시스템으로 전달해야 하지만, 모든 추가 IBM MQ 명령은 SYSPLEX에 연결된 임의의 콘솔에서 입력할 수 있으며 자동으로 대상 시스템으로 라우트됩니다.

  IBM MQ 종료 후, ROUTE 명령을 사용하여 후속 START 명령을 대상 IBM MQ 서브시스템으로 전달해야 합니다.

X
: Sysplex IPL 범위.

  CPF는 IBM MQ에 의해 시스템 IPL 시에 z/OS에 등록되고 z/OS 시스템이 활성인 전체 시간 동안 등록된 상태로 있습니다.

  IBM MQ 명령은 SYSPLEX에 연결된 임의의 콘솔에서 입력될 수 있으며, 자동으로 대상 시스템을 실행 중인 이미지로 라우트됩니다.

범위 S인 CPF를 갖는 IBM MQ 서브시스템은 SYSPLEX 내의 하나 이상의 z/OS 이미지에서 정의될 수 있으므로, 이들 이미지는 단일 서브시스템 이름 테이블을 공유할 수 있습니다. 그러나 초기 START 명령이 IBM MQ 서브시스템이 실행하기 원하는 z/OS 이미지에서 실행(또는 이미지로 라우트)되도록 해야 합니다. 이 옵션을 사용하는 경우 서브시스템 이름 테이블을 변경하거나 z/OS 시스템의 IPL을 수행할 필요 없이 IBM MQ 서브시스템을 중지하고 SYSPLEX 내의 다른 z/OS 이미지에서 재시작할 수 있습니다.

범위 X인 CPF를 갖는 IBM MQ 서브시스템은 SYSPLEX 내의 한 z/OS 이미지에서만 정의될 수 있습니다. 이 옵션을 사용하는 경우 범위 X인 CPF를 갖는 IBM MQ 서브시스템이 필요한 각 z/OS 이미지에 대해 고유한 서브시스템 이름 테이블을 정의해야 합니다.

z/OS 자동 재시작 관리자(ARM)를 사용하여 자동으로 다른 z/OS 이미지에서 큐 관리자를 재시작하려는 경우, 해당 큐 관리자가 재시작될 수 있는 각 z/OS 이미지에서 모든 큐 관리자가 정의되어야 합니다. 모든 큐 관리자는 S의 CPF 범위를 갖고 시스템 전반의 고유한 4문자 서브시스템 이름을 갖고 정의되어야 합니다.

z/OS *IBM MQ* 큐 관리자에 대한 프로시저 작성

각 IBM MQ 서브시스템은 큐 관리자를 시작하기 위한 카탈로깅된 프로시저가 필요합니다. 사용자 자신의 프로시저를 작성하거나 IBM이 제공하는 프로시저 라이브러리를 사용할 수 있습니다.

- 각 IBM MQ 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 카탈로그화된 프로시저를 수정해야 할 수 있습니다.

서브시스템 이름 테이블에 정의된 각 IBM MQ 서브시스템에 대해, 큐 관리자 시작을 위한 프로시저 라이브러리에 카탈로깅된 프로시저를 작성하십시오. IBM 제공 프로시저 라이브러리는 SYS1.PROCLIB로 부르지만, 설치가 고유한 이름 지정 규칙을 사용할 수 있습니다.

태스크 프로시저를 시작한 큐 관리자의 이름은 서브시스템 이름을 문자 MSTR과 연결하여 형성됩니다. 예를 들어, CSQ1 서브시스템은 프로시저 이름 CSQ1MSTR을 갖습니다. 사용자가 정의하는 각 서브시스템에 하나의 프로시저가 필요합니다.

선택한 언어로 된 메시지가 들어 있는 라이브러리를 포함해야 합니다.

- thlqual.SCSQSNLE(미국 영어 혼합 대소문자)
- thlqual.SCSQSNLU(미국 영어 대문자)
- thlqual.SCSQSNLK(일본어)
- thlqual.SCSQSNLF(프랑스어)
- thlqual.SCSQSNLC(중국어)

이 제품 문서의 많은 예와 지시사항은 CSQ1이라는 서브시스템이 있다고 가정합니다. CSQ1이라는 서브시스템이 초기에 설치 확인 및 테스트 목적으로 작성되는 경우 이들 예가 사용하기에 더 쉬움을 발견할 수 있습니다.

두 개의 시작된 태스크 프로시저 샘플이 thlqual.SCSQPROC에 제공됩니다. CSQ4MSTR 멤버는 각 메시지 클래스에 대해 하나의 페이지 세트를 사용하고, CSQ4MSRR 멤버는 주요 메시지 클래스에 대해 다중 페이지 세트를 사용합니다. 이러한 프로시저 중 하나를 SYS1.PROCLIB, 또는 SYS1.PROCLIB를 사용하지 않는 경우 자체 프로시저 라이브러리의 멤버 xxxxMSTR(여기서 xxxx는 IBM MQ 서브시스템의 이름)에 복사하십시오. 사용자가 정의하는 각 IBM MQ 서브시스템에 대해 샘플 프로시저를 프로시저 라이브러리의 멤버에 복사하십시오.

멤버를 복사한 경우, 멤버의 지시사항을 사용하여 각 서브시스템의 요구사항에 맞게 이를 조정할 수 있습니다. 16MB 라인 아래, 16MB 라인 위 및 2GB 막대 위로 리전 크기 지정에 대한 정보는 제안되는 리전 크기를 참조하십시오. 프로시저가 시작될 때 수정할 수 있도록 허용하기 위해 JCL에서 기호 매개변수를 사용할 수도 있습니다. 여러 IBM MQ 서브시스템이 있는 경우, 프로시저의 공용 부분에 대해 JCL 포함 그룹을 사용하여 추후 유지보수를 간소화하는 것이 유리하다는 것을 알게 될 것입니다.

큐 공유 그룹을 사용 중인 경우, STEPLIB 연결이 Db2 런타임 대상 라이브러리 SDSNLOAD를 포함하고 APF 승인되어야 합니다. 이 라이브러리는 링크 목록 또는 LPA를 통해 액세스할 수 없는 경우 STEPLIB 연결에만 필요합니다.

Advanced Message Security를 사용 중인 경우 STEPLIB 연결은 *thlqual*.SDRQAUTH를 포함해야 하며 APF 승인되어야 합니다.

참고: JCL에서 사용하기 위해 BSDS(Bootstrap Data Set)의 이름, 로그 페이지 세트를 기록한 후 프로세스의 후속 단계에서 이러한 세트를 정의할 수 있습니다.

관련 개념

606 페이지의 『채널 시작기에 대한 프로시저 작성』
각 IBM MQ 서브시스템에 대해 CSQ4CHIN의 사본을 조정하십시오. 사용 중인 기타 제품에 따라 다른 데이터 세트에 대한 액세스를 허용해야 할 수 있습니다.

z/OS 채널 시작기에 대한 프로시저 작성

각 IBM MQ 서브시스템에 대해 CSQ4CHIN의 사본을 조정하십시오. 사용 중인 기타 제품에 따라 다른 데이터 세트에 대한 액세스를 허용해야 할 수 있습니다.

- 각 IBM MQ 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 카탈로그화된 프로시저를 수정해야 할 수 있습니다.

분산 큐잉을 사용할 각 IBM MQ 서브시스템에 대한 채널 시작기 시작된 태스크 프로시저를 작성해야 합니다.

이를 수행하려면:

1. 샘플 시작된 태스크 프로시저 thlqual.SCSQPROC(CSQ4CHIN)를 프로시저 라이브러리에 복사하십시오. 프로시저 이름을 *xxxx* CHIN으로 지정하십시오. 여기서, *xxxx* 은 IBM MQ 서브시스템의 이름입니다 (예: CSQ1CHIN 은 큐 관리자 CSQ1의 채널 시작기 시작 태스크 프로시저가 됨).
2. 사용하려는 각 IBM MQ 서브시스템에 대한 사본을 작성하십시오.
3. 샘플 프로시저 CSQ4CHIN에서 지시사항을 사용하여 사용자 요구사항에 맞게 프로시저를 조정하십시오. 프로시저가 시작될 때 수정할 수 있도록 허용하기 위해 JCL에서 기호 매개변수를 사용할 수도 있습니다. 이는 IBM MQ for z/OS 관리의 시작 옵션에서 설명됩니다.

   분산 큐잉 라이브러리 thlqual.SCSQMVR1을 연결하십시오.

   LE 런타임 라이브러리 SCEERUN에 대한 액세스가 필요합니다. 링크 목록에 없는 경우 (SYS1.PARMLIB(LNKLSTkk)), STEPLIB DD 명령문에서 이를 연결하십시오.
4. 외부 보안 관리자 아래에서 실행하도록 프로시저에 권한을 부여하십시오.

5. 선택한 언어로 된 메시지가 들어 있는 라이브러리를 포함해야 합니다.
   - thlqual.SCSQSNLE(미국 영어 혼합 대소문자)
   - thlqual.SCSQSNLU(미국 영어 대문자)
   - thlqual.SCSQSNLK(일본어)
   - thlqual.SCSQSNLF(프랑스어)
   - thlqual.SCSQSNLC(중국어)

채널 시작기는 장기 실행하는 주소 공간입니다. 제한된 양의 CPU가 소비된 후 종료되는 것을 방지하려면 다음 중 하나를 확인하십시오.

- z/OS 시스템의 시작된 태스크에 대한 기본값이 무제한 CPU입니다. TIME=(1440,00)을 갖는 JOBCLASS(STC)에 대한 JES2 구성 명령문이 이를 달성합니다. 또는
- CSQXJST에 대한 EXEC 명령문에 명시적으로 TIME=1440 또는 TIME=NOLIMIT 매개변수를 추가하십시오.

채널 엑시트를 사용하려는 경우 나중에 이 프로시저에 엑시트 라이브러리(CSQXLIB)를 추가할 수 있습니다. 이를 수행하려면 채널 시작기를 중지하고 재시작해야 합니다.

TLS를 사용하는 경우 시스템 TLS 런타임 라이브러리에 대한 액세스가 필요합니다. 이 라이브러리를 SIEALNKE라고 합니다. 라이브러리는 APF 승인되어야 합니다.

TCP/IP를 사용 중인 경우, 채널 시작기 주소 공간에서 TCP/IP 시스템 매개변수가 포함된 TCPIP.DATA 데이터 세트에 액세스할 수 있어야 합니다. 데이터 세트를 설정하는 방법은 사용 중인 인터페이스와 TCP/IP 제품에 달려 있습니다. 다음이 포함됩니다.

- 환경 변수 RESOLVER_CONFIG
- HFS 파일, /etc/resolv.conf
- //SYSTCPD DD문
- //SYSTCPDD DD문
- *jobname/userid*.TCPIP.DATA
- SYS1.TCPPARMS(TCPDATA)
- *zapname*.TCPIP.DATA

이 중 일부는 시작된 태스크 프로시저 JCL에 영향을 미칩니다. 자세한 정보는 z/OS 통신 서버: IP 구성 안내서의 내용을 참조하십시오.

관련 개념

607 페이지의 『z/OS WLM 서비스 클래스에 대해 IBM MQ 서브시스템 정의』
IBM MQ에 z/OS 시스템에서 적절한 성능 우선순위를 제공하려면 큐 관리자 및 채널 시작기 주소 공간을 적절한 z/OS 워크로드 관리(WLM) 서비스 클래스에 지정해야 합니다. 이것을 명시적으로 수행하지 않는 경우 부적절한 기본값이 적용될 수 있습니다.

z/OS *z/OS WLM* 서비스 클래스에 대해 *IBM MQ* 서브시스템 정의

IBM MQ에 z/OS 시스템에서 적절한 성능 우선순위를 제공하려면 큐 관리자 및 채널 시작기 주소 공간을 적절한 z/OS 워크로드 관리(WLM) 서비스 클래스에 지정해야 합니다. 이것을 명시적으로 수행하지 않는 경우 부적절한 기본값이 적용될 수 있습니다.

- 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때는 이 태스크를 수행할 필요가 없습니다.

다음 태스크를 수행하려면 WLM에서 제공되는 ISPF 대화 상자를 사용하십시오.

- WLM 결합 데이터 세트로부터 z/OS WLM 정책 정의를 추출하십시오.
- 큐 관리자 및 채널 시작기 시작된 태스크 프로시저 이름을 선택된 서비스 클래스에 추가하여 이 정책 정의를 업데이트하십시오.
- WLM 결합 데이터 세트에 변경된 정책을 설치하십시오.

그런 다음 z/OS 명령을 사용하여 이 정책을 활성화하십시오.

```
V WLM,POLICY=policyname,REFRESH
```

성능 옵션 설정에 대한 자세한 정보는 을 참조하십시오.

관련 개념

639 페이지의 『Db2 환경 설정』
큐 공유 그룹을 사용하는 경우 여러 샘플 작업을 사용자 정의 및 실행하여 필요한 Db2 오브젝트를 작성해야 합니다.

z/OS *ESM* 보안 제어 구현

큐 관리자 및 채널 시작기에 대한 보안 제어를 구현합니다.

- 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.

RACF®를 외부 보안 관리자로 사용하는 경우, z/OS에서 보안 설정을 참조하십시오. 여기에서는 이러한 보안 제어의 구현 방법에 대해 설명합니다.

채널 시작기를 사용 중인 경우 다음도 수행해야 합니다.

- 서브시스템이 활성인 연결 보안을 갖는 경우, 외부 보안 관리자에 연결 보안 프로파일 ssid.CHIN을 정의하십시오(이에 관한 정보는 채널 시작기에 대한 연결 보안 프로파일을 참조하십시오).
- TLS (Transport Layer Security) 또는 소켓 인터페이스를 사용하는 경우, 채널 시작기가 실행 중인 권한이 있는 사용자 ID가 *OS/390® UNIX* 시스템 서비스 계획 문서에 설명된 대로 UNIX 시스템 서비스를 사용하도록 구성되어 있는지 확인하십시오.
- TLS를 사용 중인 경우 채널 시작기가 그 권한 아래에서 실행 중인 사용자 ID가 ALTER QMGR 명령의 SSLKEYR 매개변수에 지정된 키 링에 액세스하도록 구성되어 있는지 확인하십시오.

큐 관리자를 시작하기 전에 다음을 통해 IBM MQ 데이터 세트 및 시스템 보안을 설정하십시오.

- 외부 보안 관리자에서 실행하도록 큐 관리자 시작된 태스크 프로시저 권한 부여.
- 큐 관리자 데이터 세트에 대한 액세스 권한 부여.

수행 방법에 대한 자세한 내용은 z/OS(r)의 보안 설치 태스크를 참조하십시오.

RACF를 사용 중인 경우, RACF STARTED 클래스를 사용한다면 시스템의 IPL을 수행할 필요가 없습니다(RACF 시작된 태스크 프로시저의 권한 부여 참조).

관련 개념

608 페이지의 『SYS1.PARMLIB 멤버 업데이트』
변경사항이 IPL 이후에도 유효한지 확인하려면, SYS1.PARMLIB의 일부 멤버를 업데이트해야 합니다.

642 페이지의 『큐 공유 그룹에 대한 ESM 보안 제어 구현』
큐 공유 그룹에서 모든 큐 관리자에 대한 보안 제어를 구현하여 Db2 및 커플링 기능 목록 구조에 액세스합니다.

z/OS *SYS1.PARMLIB* 멤버 업데이트

변경사항이 IPL 이후에도 유효한지 확인하려면, SYS1.PARMLIB의 일부 멤버를 업데이트해야 합니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 한 번씩 이 태스크를 수행해야 합니다.
- 큐 공유 그룹을 사용 중인 경우 *IBM MQ* 설정이 *sysplex*에 있는 각 *z/OS* 시스템에서 동일한지 확인해야 합니다.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.

다음과 같이 SYS1.PARMLIB 멤버를 업데이트하십시오.

1. 602 페이지의 『z/OS에 대해 IBM MQ 서브시스템 정의』에 설명된 대로 IEFSSNss 멤버를 업데이트하십시오.
2. IPL 수행 시 다음과 같은 멤버를 사용할 수 있도록 IEASYSpp를 변경하십시오.
    - 598 페이지의 『APF에서 IBM MQ 로드 라이브러리에 권한 부여』에서 사용된 PROGxx 또는 IEAAPFaa 멤버

- 600 페이지의 『z/OS 링크 목록 및 LPA 업데이트』에서 사용된 LNKLSTkk 및 LPALSTmm 멤버
- 601 페이지의 『z/OS 프로그램 특성 테이블 업데이트』에서 사용된 SCHEDxx 멤버
- 602 페이지의 『z/OS에 대해 IBM MQ 서브시스템 정의』에서 사용된 IEFSSNss 멤버

관련 개념

609 페이지의 『초기화 입력 데이터 세트 사용자 정의』
샘플 초기화 입력 데이터 세트의 작업 사본을 작성하고 시스템 요구사항에 맞게 조정하십시오.

z/OS 초기화 입력 데이터 세트 사용자 정의

샘플 초기화 입력 데이터 세트의 작업 사본을 작성하고 시스템 요구사항에 맞게 조정하십시오.

- 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 합니다.

각 IBM MQ 큐 관리자는 IBM MQ 초기화 입력 데이터 세트에 포함된 일련의 명령에서 해당 초기 정의를 가져옵니다. 이러한 데이터 세트는 큐 관리자 시작된 태스크 프로시저에 정의된 DDnames CSQINP1, CSQINP2 및 CSQINPT에 의해 참조됩니다.

이러한 명령에 대한 응답은 DDnames CSQOUT1, CSQOUT2 및 CSQOUTT에서 참조되는 초기화 출력 데이터 세트에 기록됩니다.

원본을 보존하려면 각 샘플의 작업 사본을 작성하십시오. 그런 다음, 시스템 요구사항에 맞도록 이러한 작업 사본에서 명령을 조정할 수 있습니다.

둘 이상의 IBM MQ 서브시스템을 사용하는 경우 초기화 입력 데이터 세트 이름의 상위 레벨 규정자에 서브시스템 이름을 포함하면 각 데이터 세트에 연관된 IBM MQ 서브시스템을 더 쉽게 식별할 수 있습니다.

샘플에 대한 자세한 정보는 다음 주제를 참조하십시오.

- 초기화 데이터 설정 형식
- CSQINP1 샘플 사용
- CSQINP2 샘플 사용
- CSQINPX 샘플 사용
- CSQINPT 샘플 사용

초기화 데이터 세트 형식

초기화 입력 데이터 세트는 파티션된 데이터 세트(PDS) 멤버 또는 순차 데이터 세트일 수 있습니다. 데이터 세트의 병합된 시리즈일 수 있습니다. 레코드 길이가 80바이트로 이들을 정의하십시오. 여기서,

- 1 - 72 사이의 열만 유의미합니다. 73 - 80 사이의 열은 무시됩니다.
- 열 1에 별표(*)가 있는 레코드는 주석으로 해석되어 무시됩니다.
- 공백 레코드는 무시됩니다.
- 각 명령은 새 레코드에서 시작되어야 합니다.
- 후미 문자 -는 다음 레코드의 열 1부터 계속됨을 의미합니다.
- 후미 문자 +는 다음 레코드의 공백이 아닌 첫 번째 열에서 계속됨을 나타냅니다.
- 명령에서 허용되는 최대 문자 수는 32,762입니다.

초기화 출력 데이터 세트는 레코드 길이가 125이고 레코드 형식이 VBA이며 블록 크기가 629인 순차 데이터 세트입니다.

CSQINP1 샘플 사용

데이터 세트 thlqual.SCSQPROC는 버퍼 풀, 버퍼 풀 연관에 대한 페이지 세트 및 ALTER SECURITY 명령의 정의를 포함하는 두 개 멤버를 보유합니다.

CSQ4INP1 멤버는 각 메시지 클래스에 하나의 페이지 세트를 사용합니다. CSQ4INPR 멤버는 주요 메시지 클래스에 다중 페이지 세트를 사용합니다.

큐 관리자 시작된 태스크 프로시저의 CSQINP1 연결에 적절한 샘플을 포함하십시오.

참고사항:

1. IBM MQ는 0 - 99 범위에서 최대 100개의 버퍼 풀을 지원합니다. DEFINE BUFFPOOL 명령은 CSQINP1 초기화 데이터 세트에서만 발행될 수 있습니다. 샘플의 정의는 네 개 버퍼 풀을 지정합니다.
2. 큐 관리자에서 사용하는 각 페이지 세트는 DEFINE PSID 명령을 사용하여 CSQINP1 초기화 데이터 세트에 정의되어야 합니다. 페이지 세트 정의는 버퍼 풀 ID를 페이지 세트를 연관시킵니다. 버퍼 풀이 지정되지 않은 경우 기본적으로 버퍼 풀 0이 사용됩니다.

   페이지 세트 0(00)을 정의해야 합니다. 여기에는 모든 오브젝트 정의가 포함됩니다. 각 큐 관리자에 대해 최대 100개의 페이지 세트를 정의할 수 있습니다.
3. ALTER SECURITY 명령은 보안 속성 TIMEOUT 및 INTERVAL을 대체하는 데 사용할 수 있습니다. CSQ4INP1에서 기본값은 TIMEOUT의 경우 54, INTERVAL의 경우 12로 정의됩니다.

버퍼 풀 및 페이지 세트 구성에 대한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

큐 관리자가 실행 중인 동안 버퍼 풀 및 페이지 세트 정의를 동적으로 변경하는 경우, CSQINP1 정의도 업데이트해야 합니다. 버퍼 풀 정의가 REPLACE 속성을 포함하지 않으면, IBM MQ의 콜드 스타트에 대해서만 변경사항이 유지됩니다.

CSQINP2 샘플 사용

이 표에서는 해당 기능의 설명과 함께 큐 관리자 시작된 태스크 프로시저의 CSQINP2 연결에 포함할 수 있는 thlqual.SCSQPROC의 멤버를 나열합니다. 이름 지정 규칙은 CSQ4INS*입니다. CSQ4INY*는 사용자 구성에 맞게 수정해야 합니다. 다음 릴리스로 마이그레이션할 때 모든 변경을 재적용해야 하기 때문에 CSQINS* 멤버 변경을 피해야 합니다. 대신, CSQ4INY* 멤버에 DEFINE 또는 ALTER 명령을 넣을 수 있습니다.

표 *42. thlqual.SCSQPROC*의 멤버

멤버 이름	설명
CSQ4INSG	시스템 오브젝트 정의.
CSQ4INSA	채널 인증에 사용되는 시스템 오브젝트 및 기본 규칙.
CSQ4INSX	시스템 오브젝트 정의.
CSQ4INSS	큐 공유 그룹을 사용하는 경우 이 멤버를 사용자 정의 및 포함하십시오.
CSQ4INSJ	JMS를 사용하는 발행/구독을 사용 중인 경우 이 멤버를 사용자 정의하고 포함하십시오.
CSQ4INSM	고급 메시지 보안을 위한 시스템 오브젝트 정의.
CSQ4INSR	IBM MQ V7 이상에서 큐된 발행/구독 디먼에서 지원하는 큐된 발행/구독 인터페이스 또는 WebSphere Application Server를 사용하는 경우 이 멤버를 사용자 정의 및 포함하십시오.
CSQ4DISP	오브젝트 정의를 표시하기 위한 CSQINP2 샘플.
CSQ4INYC	클러스터링 정의.
CSQ4INYD	분산 큐잉 정의.
CSQ4INYG	일반 정의.
CSQ4INYR	메시지의 주요 클래스에 대해 다중 페이지 세트를 사용하는 스토리지 클래스 정의.
CSQ4INYS	각 메시지 클래스에 대해 하나의 페이지 세트를 사용하는 스토리지 클래스 정의.

큐 관리자를 시작할 때마다가 아니라 오브젝트를 한 번만 정의해야 합니다. 따라서 항상 CSQINP2에서 이 정의를 포함하지 않아도 됩니다. 항상 포함하는 경우 이미 존재하는 오브젝트를 정의하려고 하며 다음과 유사한 메시지를 가져옵니다.

```
CSQM095I +CSQ1 CSQMAQLC QLOCAL(SYSTEM.DEFAULT.LOCAL.QUEUE) ALREADY EXISTS
CSQM090E +CSQ1 CSQMAQLC FAILURE REASON CODE X'00D44003'
CSQ9023E +CSQ1 CSQMAQLC ' DEFINE QLOCAL' ABNORMAL COMPLETION
```

오브젝트는 이 실패로 손상되지 않습니다. CSQINP2 연결에서 SYSTEM 정의 데이터 세트를 남겨두는 경우 각 오브젝트에 대해 REPLACE 속성을 지정하여 실패 메시지를 방지할 수 있습니다.

CSQINPX 샘플 사용

샘플 thlqual.SCSQPROC(CSQ4INPX)는 채널 시작기가 시작될 때마다 실행하려는 명령 세트를 포함합니다. 이는 일반적으로 입력 데이터 세트 CSQINP1 또는 CSQINP2에서 허용되지 않으며 큐 관리자를 시작할 때마다가 아니라 채널 시작기를 시작할 때마다 필요한 START LISTENER와 같은 채널 관련 명령입니다. 사용하기 전에 이 샘플을 사용자 정의해야 합니다. 그런 다음 채널 시작기에 대해 CSQINPX 데이터 세트에 이를 포함할 수 있습니다.

데이터 세트에 포함된 IBM MQ 명령은 채널 시작기 초기화의 끝에 실행되며 CSQOUTX DD 문에서 지정된 데이터 세트에 출력이 작성됩니다. 출력은 IBM MQ 유틸리티 프로그램(CSQUTIL)의 COMMAND 함수에서 생성된 것과 유사합니다. 자세한 내용은 CSQUTIL 유틸리티를 참조하십시오.

채널 명령뿐만 아니라 CSQUTIL에서 실행할 수 있는 IBM MQ 명령을 지정할 수 있습니다. CSQINPX가 처리되는 동안 다른 소스에서 명령을 입력할 수 있습니다. 모든 명령은 이전 명령의 성공에 상관없이 순서대로 실행됩니다.

명령 응답 시간을 지정하려면 데이터 세트에서 첫 번째 명령으로 의사 명령 COMMAND를 사용할 수 없습니다. 이것은 단일 선택적 키워드 RESPTIME(*nnn*)을 취하는데, 여기서 *nnn*은 각 명령에 대한 응답을 대기할 시간(초)입니다. 해당 범위는 5 - 999이며, 기본값은 30입니다.

IBM MQ가 네 개의 명령에 대한 응답이 너무 오래 걸림을 감지하면, CSQINPX의 처리가 중지되고 추가 명령이 실행되지 않습니다. 채널 시작기가 중지되지 않았지만 CSQU052E 메시지가 CSQOUTX 데이터 세트에 기록되고 CSQU013E 메시지가 콘솔로 전송됩니다.

IBM MQ이(가) CSQINPX의 처리를 완료하면 CSQU012I 메시지가 콘솔로 전송됩니다.

CSQINPT 샘플 사용

이 표에서는 해당 기능의 설명과 함께 큐 관리자 시작된 태스크 프로시저의 CSQINPT 연결에 포함할 수 있는 thlqual.SCSQPROC의 멤버를 나열합니다.

표 *43. thlqual.SCSQPROC*의 멤버

멤버 이름	설명
CSQ4INST	시스템 기본 구독 정의
CSQ4INYT	발행/구독 정의.

데이터 세트에 포함된 IBM MQ 명령은 발행/구독 초기화 완료 시 실행되며 CSQOUTT DD 문에서 지정된 데이터 세트에 출력이 작성됩니다. 출력은 IBM MQ 유틸리티 프로그램(CSQUTIL)의 COMMAND 함수에서 생성된 것과 유사합니다. 자세한 내용은 CSQUTIL 유틸리티를 참조하십시오.

관련 개념

611 페이지의 『부트스트랩 및 로그 데이터 세트 작성』
부트스트랩 데이터 세트(BSDS) 및 로그 데이터 세트를 준비하려면 제공되는 프로그램 CSQJU003을 사용하십시오.

z/OS 부트스트랩 및 로그 데이터 세트 작성

부트스트랩 데이터 세트(BSDS) 및 로그 데이터 세트를 준비하려면 제공되는 프로그램 CSQJU003을 사용하십시오.

- 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.

- 이전 버전에서 마이그레이션할 때는 이 태스크를 수행할 필요가 없습니다.

단일 또는 이중 로깅 환경을 작성하기 위해 CSQJU003을 실행하기 위한 샘플 JCL 및 액세스 방법 서비스(AMS) 제어 명령문은 thlqual.SCSQPROC(CSQ4BSDS)에 있습니다. BSDS 및 로그를 작성하고 로그를 사전 포맷하려면 이 작업을 사용자 정의하고 실행하십시오.

중요사항: RECORDS(850 60)을 사용하려면 CSQ4BSDS의 최신 버전을 사용하거나 JCL을 수동으로 업데이트해야 합니다.

605 페이지의 『IBM MQ 큐 관리자에 대한 프로시저 작성』에서 설명하는 시작된 태스크 프로시저 CSQ4MSTR은 다음 양식의 명령문에서 BSDS를 참조합니다.

```
//BSDS1    DD DSN=++HLQ++.BSDS01,DISP=SHR
//BSDS2    DD DSN=++HLQ++.BSDS02,DISP=SHR
```

로그 데이터 세트가 BSDS에 의해 참조됩니다.

참고:

1. BLKSIZE가 LOGDEF 단계의 SYSPRINT DD문에서 지정되어야 합니다. BLKSIZE는 629여야 합니다.
2. 부트스트랩 데이터 세트와 로그 데이터 세트를 다른 큐 관리자로부터 식별하는 데 도움을 받으려면 이들 데이터 세트의 상위 레벨 규정자에 서브시스템 이름을 포함하십시오.
3. 큐 공유 그룹을 사용 중인 경우 SHAREOPTIONS(2 3)으로 부트스트랩 및 로그 데이터 세트를 정의해야 합니다.

부트스트랩 및 로그 데이터 세트와 해당 크기 계획에 대한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

IBM MQ 8.0에서 8바이트 로그 RBA 향상은 더 큰 로그 상대 바이트 주소에 설명된 대로 큐 관리자의 가용성을 향상시킵니다. 큐 관리자가 먼저 시작되기 전에 큐 관리자에서 8바이트 로그 RBA를 사용으로 설정하려면 로깅 환경을 작성한 후 다음 단계를 수행하십시오.

1. **IDCAMS ALTER**를 사용하여 버전 1 형식 BSDS(CSQJU003 프로그램을 사용하여 작성)를 `++HLQ++.V1.BSDS01` 같은 것으로 이름을 바꾸십시오.

   참고: 데이터 및 색인 컴포넌트뿐 아니라 VSAM 클러스터도 이름을 바꿔야 합니다.
2. 이미 정의된 것과 동일한 속성을 갖는 새 BSDS를 할당하십시오. 이들은 시작될 때 큐 관리자가 사용할 버전 2 형식 BSDS가 됩니다.
3. BSDS 변환 유틸리티(CSQJUCNV)를 실행하여 버전 1 형식 BSDS를 버전 2 형식 BSDS로 변환하십시오.
4. 변환이 성공적으로 완료한 후 버전 1 형식 BSDS를 삭제하십시오.

참고: 큐 관리자가 큐 공유 그룹에 있는 경우, 8바이트 로그 RBA를 사용으로 설정하려면 먼저 큐 공유 그룹에 있는 모든 큐 관리자가 follows로 시작되었어야 합니다.

- 큐 관리자가 IBM MQ 8.0.0인 경우 **OPMODE**(*NEWFUNC,800*)로 시작되어야 합니다.
- 큐 관리자가 IBM MQ 9.0.0 LTS인 경우, **OPMODE**(NEWFUNC,900) 또는 **OPMODE**(*NEWFUNC,800*)로 시작되어야 합니다.
- 큐 관리자가 IBM MQ 9.0.x CD, IBM MQ 9.1.0 LTS이상에 있으면 해당 레벨에서 시작되어야 합니다.

관련 개념

612 페이지의 『페이지 세트 정의』
제공되는 샘플 중 하나를 사용하여 각 큐 관리자에 대한 페이지 세트를 정의하십시오.

z/OS 페이지 세트 정의

제공되는 샘플 중 하나를 사용하여 각 큐 관리자에 대한 페이지 세트를 정의하십시오.

- 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때는 이 태스크를 수행할 필요가 없습니다.

각 IBM MQ 큐 관리자에 대해 별도의 페이지 세트를 정의하십시오. thlqual.SCSQPROC(CSQ4PAGE) 및 thlqual.SCSQPROC(CSQ4PAGR)에 페이지 세트를 정의 및 형식화하기 위한 JCL 및 AMS 제어 명령문을 포함하

고 있습니다. CSQ4PAGE 멤버는 메시지의 각 클래스에 대해 하나의 페이지 세트를 사용하고, CSQ4PAGR 멤버는 메시지의 주 클래스에 대한 여러 페이지 세트를 사용합니다. JCL은 제공되는 유틸리티 프로그램 CSQUTIL을 실행합니다. 샘플을 검토하고 원하는 페이지 세트의 수와 사용할 크기에 대해 사용자 정의하십시오. 페이지 세트 및 적합한 크기를 계산하는 방법에 대한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

605 페이지의 『IBM MQ 큐 관리자에 대한 프로시저 작성』에서 설명하는 시작된 태스크 프로시저 CSQ4MSTR은 다음 양식의 명령문에서 페이지 세트를 참조합니다.

```
//CSQP00 nn DD DISP=OLD,DSN= xxxxxxxxxx
```

여기서 *nn*은 00 - 99 범위의 페이지 세트 번호이고, *xxxxxxxxxx*는 사용자가 정의하는 데이터 세트입니다.

참고:

1. 동적 페이지 세트 확장 기능을 사용하려는 경우 보조 범위가 각 페이지 세트에 대해 정의되는지 확인하십시오. thlqual.SCSQPROC(CSQ4PAGE)가 이를 수행하는 방법을 보여줍니다.
2. 페이지 세트를 다른 큐 관리자로부터 식별하는 데 도움을 받으려면 각 페이지 세트와 연관된 데이터 세트의 상위 레벨 규정자에 서브시스템 이름을 포함하십시오.
3. FORCE 옵션이 유틸리티 프로그램 CSQUTIL의 FORMAT 기능에서 사용되도록 허용하려는 경우 AMS DEFINE CLUSTER 명령문에 REUSE 속성을 추가해야 합니다. 이것은 IBM MQ for z/OS 관리에서 설명됩니다.
4. 페이지 세트가 4GB보다 더 클 경우 스토리지 관리 시스템(SMS) EXTENDED ADDRESSABILITY 함수를 사용해야 합니다.

관련 개념

641 페이지의 『Db2 테이블에 IBM MQ 입력 항목 추가』

큐 공유 그룹을 사용하는 경우, CSQ5PQSG 유틸리티를 실행하여 큐 공유 그룹 및 큐 관리자 항목을 Db2 데이터 공유 그룹의 IBM MQ 테이블에 추가하십시오.

z/OS 시스템 매개변수 모듈 조정

IBM MQ 시스템 매개변수 모듈은 IBM MQ가 그의 조작에서 사용하는 로깅, 아카이빙, 추적 및 연결 환경을 제어합니다. 기본 모듈이 제공됩니다. 일부 매개변수(예: 데이터 세트 이름)가 일반적으로 사이트에 한정되므로 고유 시스템 매개변수 모듈을 작성해야 합니다.

- 필요에 따라 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다. 자세한 내용은 z/OS에서 IBM MQ 마이그레이션을 참조하십시오.
- 기존 큐 관리자에서 *Advanced Message Security for z/OS*를 사용하려면 615 페이지의 『CSQ6SYSP 사용』에 설명된 대로 *SPLCAP*를 *YES*로 설정하기만 하면 됩니다. 이 큐 관리자를 처음으로 구성하는 경우 이 태스크 전체를 완료하십시오.

시스템 매개변수 모듈에는 다음과 같이 V 9.0.3 4개의 매크로가 있습니다.

매크로 이름	목적
CSQ6SYSP	연결 및 추적 매개변수를 지정합니다(615 페이지의 『CSQ6SYSP 사용』 참조).
CSQ6LOGP	로그 초기화 제어에 대해서는 623 페이지의 『CSQ6LOGP 사용』의 내용을 참조하십시오.
CSQ6ARVP	아카이브 초기화 제어에 대해서는 626 페이지의 『CSQ6ARVP 사용』의 내용을 참조하십시오.
V 9.0.3 V 9.0.3 CSQ6USGP	사용 기록을 제어합니다(633 페이지의 『CSQ6USGP 사용』 참조).

IBM MQ는 IBM MQ의 인스턴스를 시작하기 위해 PARM 매개변수 없이 START QMGR 명령을 실행하는 경우에 자동으로 호출되는 기본 시스템 매개변수 모듈(CSQZPARM)을 제공합니다. 또한 CSQZPARM은 IBM MQ에서 제공하는 APF 권한이 부여된 라이브러리 thlqual.SCSQAUTH에도 있습니다. 이러한 매개변수 값은 IBM MQ를 시작할 때 일련의 메시지로 표시됩니다.

이 명령이 사용되는 방법에 관한 자세한 정보는 START QMGR을 참조하십시오.

시스템 매개변수 모듈 작성

CSQZPARM에 원하는 시스템 매개변수가 없으면 thlqual.SCSQPROC(CSQ4ZPRM)에서 제공되는 샘플 JCL을 사용하여 시스템 매개변수 모듈을 작성할 수 있습니다.

사용자 고유의 시스템 매개변수 모듈을 작성하려면 다음을 수행하십시오.

1. JCL 샘플의 작업 사본을 작성하십시오.
2. 필요한 경우 사본에서 각 매크로에 대해 매개변수를 편집하십시오. 매크로 호출에서 매개변수를 제거하는 경우 기본값은 런타임에 자동으로 선택됩니다.
3. 로드 모듈에서 취하는 이름(CSQZPARM일 수 있음)으로 플레이스홀더 ++NAME++을 바꾸십시오.
4. 어셈블러가 상위 레벨 어셈블러가 아닌 경우 어셈블러에 필요한 대로, JCL을 변경하십시오.
5. JCL을 실행하여 로드 모듈을 생성하도록 시스템 매개변수 매크로의 조정된 버전을 어셈블 및 링크 편집하십시오. 이는 사용자가 지정한 이름의 새 시스템 매개변수 모듈입니다.
6. APF 승인된 사용자 라이브러리에 생성된 로드 모듈을 넣으십시오.
7. APF 승인된 사용자 라이브러리에 대한 읽기 액세스를 사용자에게 추가하십시오.
8. IBM MQ 큐 관리자 시작된 태스크 프로시저 STEPLIB에 이 라이브러리를 포함하십시오. 이 라이브러리 이름은 STEPLIB에서 라이브러리 thlqual.SCSQAUTH 앞에 와야 합니다.
9. 큐 관리자를 시작할 때 새 시스템 매개변수 모듈을 호출하십시오. 예를 들어 새 모듈 이름이 NEWMODS인 경우 다음 명령을 실행하십시오.

   ```
   START QMGR PARM(NEWMODS)
   ```

10. 작업 로그를 검사하여 명령이 성공적으로 완료되었는지 확인하십시오. 다음과 유사한 항목이 로그에 있어야 합니다.

    ```
    CSQ9022I CDL1 CSQYASCP 'START QMGR' NORMAL COMPLETION
    ```

또한 큐 관리자 시동 JCL에서 매개변수 모듈 이름도 지정할 수 있습니다. 자세한 정보는 큐 관리자 시작 및 중지를 참조하십시오.

참고: 모듈 CSQZPARM 이름을 지정하도록 선택한 경우 START QMGR 명령에서 PARM 매개변수를 지정하지 않아도 됩니다.

시스템 매개변수 모듈 성능 조정

또한 IBM MQ는 기존 시스템 매개변수 모듈을 자세히 조정하는 데 사용할 수 있는 세 개의 어셈블러 소스 모듈 세트도 제공합니다. 이러한 모듈은 라이브러리 thlqual.SCSQASMS에 있습니다. 일반적으로 테스트 환경에서 이들 모듈을 사용하여 시스템 매개변수 매크로에서 기본 매개변수를 변경합니다. 각 소스 모듈은 서로 다른 시스템 매개변수 매크로를 호출합니다.

이 어셈블러 소스 모듈에서...	이 매크로를 호출함...
CSQFSYSP	CSQ6SYSP(연결 및 추적 매개변수)
CSQJLOGP	CSQ6LOGP(로그 초기화)
CSQJARVP	CSQ6ARVP(아카이브 초기화)

다음은 이러한 모듈을 사용하는 방법입니다.

1. 사용자 어셈블러 라이브러리에서 각 어셈블러 소스 모듈의 작업 사본을 작성합니다.
2. 필요한 경우 매개변수 값을 추가 또는 대체하여 사본을 편집합니다.
3. 사용자 오브젝트 라이브러리에서 오브젝트 모듈을 작성하도록 편집된 모듈의 사본을 어셈블합니다.
4. 새 시스템 매개변수 모듈에 해당하는 로드 모듈을 생성하도록 기존 시스템 매개변수 모듈과 이러한 오브젝트 코드 모듈을 링크 편집합니다.
5. 새 시스템 매개변수 모듈이 사용자 권한이 부여된 라이브러리의 멤버인지 확인합니다.
6. 큐 관리자 시작된 태스크 프로시저 STEPLIB에 이 라이브러리를 포함합니다. 이 라이브러리는 STEPLIB에서 라이브러리 thlqual.SCSQAUTH 앞에 와야 합니다.
7. 이전과 같이 PARM 매개변수에서 새 모듈 이름을 지정하고 START QMGR 명령을 실행하여 새 시스템 매개변수 모듈을 호출하십시오.

샘플 usermod는 SMP/E 제어 아래에서 사용자 정의된 시스템 매개변수를 관리하는 방법을 설명하는 SCSQPROC의 CSQ4UZPR 멤버에서 제공됩니다.

시스템 매개변수 대체

큐 관리자가 실행하고 있는 중에 일부 시스템 매개변수를 변경해야 합니다. SET SYSTEM, SET LOG, SET ARCHIVE 명령을 참조하십시오.

큐 관리자를 시작할 때마다 적용할 수 있도록 초기화 입력 데이터 세트에 SET 명령을 넣습니다.

관련 개념

634 페이지의 『채널 시작기 매개변수 조정』
요구사항에 맞도록 채널 시작기를 사용자 정의하려면 ALTER QMGR을 사용하십시오.

z/OS *CSQ6SYSP* 사용

이 토픽을 CSQ6SYSP를 사용하여 시스템 매개변수를 설정하는 방법에 대한 참조로 사용하십시오.

CSQ6SYSP의 기본 매개변수 및 SET SYSTEM 명령을 사용하여 각 매개변수를 대체할 수 있는지 여부가 615 페이지의 표 44에 표시됩니다. 이러한 값을 변경하려는 경우 매개변수의 자세한 설명을 참조하십시오.

표 *44. CSQ6SYSP* 매개변수의 기본값

매개변수	설명	기본값	SET 명령
ACELIM	1KB 블록의 ACE 스토리지 풀 크기입니다.	0(한계가 없음)	✓
CLCACHE	사용할 클러스터 캐시 유형을 지정합니다.	정적	-
CMDUSER	명령 보안 검사를 위한 기본 사용자 ID입니다.	CSQOPR	-
CONNSWAP	**주의:** IBM MQ 9.0부터 이 키워드는 아무런 영향을 미치지 않습니다. 특정 IBM MQ API 호출을 발행하는 작업의 스왑 가능 여부를 지정합니다.	YES	-
EXCLMSG	로그에서 제외할 메시지 목록을 지정합니다. 이 목록의 메시지는 z/OS 콘솔 및 하드카피 로그로 송신되지 않습니다. 그러므로 EXCLMSG 매개변수를 사용하여 메시지를 제외시키는 것은 638 페이지의 『정보 메시지 억제』에서 설명하는 방법을 사용하는 것보다 CPU 관점에서 더 효율적입니다.	( )	✓
EXITLIM	각 호출 중에 큐 관리자 엑시트가 실행될 수 있는 시간(초)입니다.	30	-
EXITTCB	큐 관리자 엑시트를 실행하는 데 사용할 시작된 서버 태스크의 수입니다.	8	-

표 44. CSQ6SYSP 매개변수의 기본값 (계속)			
매개변수	**설명**	**기본값**	**SET 명령**
LOGLOAD	한 체크포인트의 시작과 다음 체크포인트의 시작 간에 IBM MQ에서 기록한 로그 레코드 수입니다.	500 000	✓
MULCCAPT	평가된 사용 라이센스 비용(MULC)에서 사용하는 데이터 수집 알고리즘을 제어하는 평가된 사용 가격 책정 특성을 판별합니다.	매개변수 설명 참조	-
OTMACON	OTMA 연결 매개변수입니다.	매개변수 설명 참조	-
QINDXBLD	큐 관리자 재시작이 모든 색인이 다시 빌드될 때까지 대기할지, 또는 모든 색인이 다시 빌드되기 전에 완료될지 결정합니다.	WAIT	-
QMCCSID	큐 관리자에 대한 코드화 문자 세트 ID입니다.	0	-
QSGDATA	큐 공유 그룹 매개변수입니다.	매개변수 설명 참조	-
RESAUDIT	RESLEVEL 감사 매개변수입니다.	YES	-
ROUTCDE	특정 콘솔에서 요청하지 않은 메시지에 지정된 메시지 라우팅 코드입니다.	1	-
SERVICE	IBM에서 사용하도록 예약됩니다.	0	✓
SMFACCT	큐 관리자가 시작될 때 SMF 회계 데이터를 수집할 것인지 여부를 지정합니다. 채널 시작기가 시작될 때만 클래스 4 채널 회계 데이터가 수집됩니다.	아니오	-
SMFSTAT	큐 관리자가 시작될 때 SMF 통계를 수집할 것인지 여부를 지정합니다. 채널 시작기가 시작될 때만 클래스 4 채널 통계 데이터가 수집됩니다.	아니오	-
SPLCAP	큐 보안 정책 기능을 이 큐 관리자에서 사용할 수 있는지 여부를 지정합니다. Advanced Message Security for z/OS의 경우 이 매개변수를 YES로 설정하십시오.	아니오	-
STATIME	각 통계 수집 간의 기본 간격(분)입니다.	30	✓
TRACSTR	추적의 자동 시작 여부를 지정합니다.	아니오	-
TRACTBL	글로벌 추적 기능에서 사용할 추적 테이블 크기(4KB 블록)입니다.	99(396KB)	✓
WLMTIME	WLM 관리 큐에 대한 큐 색인 스캔 사이의 시간입니다.	30	-
WLMTIMU	WLMTIME의 단위(분 또는 초)입니다.	MINS	-

ACELIM

1KB 블록 단위로 ACE 스토리지 풀의 최대 크기를 지정합니다. 숫자는 0-999999 범위에 있어야 합니다. 기본값 0은 시스템에 있는 제한조건 이외의 제한조건이 없음을 나타냅니다.

과도한 양의 ECSA 스토리지를 사용하는 것으로 식별된 큐 관리자에서는 ACELIM의 값만 설정해야 합니다. ACE 스토리지 풀을 제한하면 시스템에서의 연결 수가 제한되므로 큐 관리자에서 사용하는 ECSA 스토리지의 크기도 제한됩니다.

큐 관리자가 한계에 도달하면 애플리케이션이 새 연결을 확보할 수 없습니다. 새 연결이 부족하면 MQCONN 처리가 실패하게 되며 RRS를 통해 조정되는 애플리케이션이 모든 IBM MQ API에서 실패할 가능성이 높아집니다.

ACE는 연결을 위해 스레드 관련 제어 블록에 필요한 총 ECSA의 약 12.5%를 나타냅니다. 예를 들어, ACELIM=5120 을 지정하면 큐 관리자가 할당한 총 ECSA (스레드 관련 제어 블록의 경우) 가 약 40960K; 에서 5120에 8을 곱한 ECSA의 총량을 제한할 것으로 예상됩니다.

큐 관리자가 할당한 총 ECSA의 양을 제한하려면 5120K의 스레드 관련 제어 블록에 대해 ACELIM값 640이 필요합니다.

통계 CLASS(3) 추적에 의해 생성되는 SMF 115 하위 유형 5 레코드를 사용하여 'ACE/PEB' 스토리지 풀의 크기를 모니터하므로 ACELIM에 대한 적당한 값을 설정할 수 있습니다.

통계 CLASS(2) 추적에 의해 기록되는 SMF 115 하위 유형 7 레코드로부터 제어 블록을 위해 큐 관리자가 사용하는 ECSA 스토리지의 총량을 확보할 수 있습니다. 즉, 함께 추가되는 QSRSPHBT의 처음 두 요소입니다.

ACELIM을 큐 관리자에 대한 애플리케이션 연결을 제어하기 위한 수단으로서가 아니라 정상적으로 작동하지 않는 큐 관리자로부터 z/OS 이미지를 보호하기 위한 메커니즘으로 설정하는 것에 대해 고려해야 합니다.

CLCACHE

사용할 클러스터 캐시 유형을 지정합니다. 자세한 정보는 216 페이지의 『큐 관리자 클러스터 구성』의 내용을 참조하십시오.

정적

클러스터 캐시가 정적인 경우 해당 크기는 클러스터 정보의 현재 크기 및 확장을 위한 일부 공간에 충분하도록 큐 관리자 시동 시 고정됩니다. 큐 관리자가 활성 상태인 경우 크기는 늘릴 수 없습니다. 기본값입니다.

DYNAMIC

클러스터 캐시가 동적인 경우 큐 관리자 시동 시 할당된 초기 크기는 큐 관리자가 활성 상태인 동안 필요에 따라 자동으로 늘어날 수 있습니다.

CMDUSER

명령 보안 검사에 사용되는 기본 사용자 ID를 지정합니다. 이 사용자 ID는 ESM에 정의되어야 합니다(예: RACF). 1-8자의 영숫자로 구성된 이름을 지정하십시오. 첫 번째 문자는 영문자여야 합니다.

기본값은 CSQOPR입니다.

CONNSWAP

특정 IBM MQ API 호출을 발행하는 배치 작업이 IBM MQ API 요청의 지속 기간 동안 스왑 가능한지 또는 스왑 불가능한지 여부를 지정합니다. 다음 값 중 하나를 지정하십시오.

아니오

작업이 특정 IBM MQ API 호출 동안 스왑 가능하지 않습니다.

YES

작업이 모든 IBM MQ API 호출 동안 스왑 가능합니다.

기본값은 YES입니다.

다른 작업이나 태스크가 대기 중일 수 있는 IBM MQ 자원을 보유하는 동안 낮은 우선순위 작업이 스왑되는 경우 이 매개변수를 사용하십시오.

IBM MQ는 WebSphere Application Server를 RRSBATCH 환경의 파트로 봅니다. CONNSWAP 키워드가 사용될 때, 이것은 BATCH 또는 RRSBATCH 환경 내의 임의의 애플리케이션에 적용된다. CONNSWAP 키워드도 TSO 사용자에 적용될 수 있지만, CICS 또는 IMS 애플리케이션에 적용할 수 없습니다. CONNSWAP 변경은 큐 관리자의 재순환이 발생할 때 구현됩니다. CSQ6SYSP 매크로가 리어셈블링되고 큐 관리자가 이 매크로에 의해 업데이트되는 로드 모듈을 사용하여 재시작되었기 때문에 키워드가 변경된 후 재순환이 필요합니다.

그렇지 않으면 PPT를 사용하여 WebSphere Application Server 주소 공간을 스왑 불가능하게 만들 수 있습니다.

EXCLMSG

제외할 오류 메시지의 목록을 지정합니다.

이 목록은 동적이며 SET SYSTEM 명령을 사용하여 업데이트됩니다.

기본값은 빈 목록( )입니다.

메시지는 CSQ 접두부 및 조치 코드 접미부(I-D-E-A) 없이 제공됩니다. 예를 들어, 메시지 CSQX500I을 제외하려면 X500을 이 목록에 추가하십시오. 이 목록은 최대 16개의 메시지 ID를 포함할 수 있습니다.

목록에 포함되려면 메시지는 MSTR 또는 CHIN 주소 공간의 정상 시작 후 발행되어야 하며 문자 E, H, I, J, L, M, N, P, R, T, V, W, X, Y, 2 ,3, 5, 9 중 하나로 시작해야 합니다.

명령 처리의 결과로 발행된 메시지 ID는 목록에 추가할 수 있지만 제외되지는 않습니다. 예를 들면, 메시지 식별자는 DISPLAY USAGE psid(*) 명령의 결과로 발행되지만 이 메시지는 제외할 수 없습니다.

EXITLIM
큐 관리자 엑시트의 각 호출에 대해 허용되는 시간(초)을 지정합니다. (이 매개변수는 채널 엑시트에 영향을 주지 않습니다.)

5 - 9999 범위의 값을 지정하십시오.

기본값은 30입니다. 큐 관리자는 30초마다 실행되는 엑시트를 폴링합니다. 각 폴링에서 EXITLIM에서 지정한 시간을 초과하여 실행된 항목은 모두 강제 종료됩니다.

EXITTCB
큐 관리자에서 엑시트를 실행하는 데 사용할 시작된 서버 태스크 수를 지정합니다. (이 매개변수는 채널 엑시트에 영향을 주지 않습니다.) 큐 관리자가 실행해야 할 수 있는 엑시트(채널 엑시트 이외)의 최대 수만큼 높은 숫자를 지정해야 하며, 그렇지 않으면 6c6 이상종료와 함께 실패합니다.

0 - 99 범위의 값을 지정하십시오. 0의 값은 엑시트를 실행할 수 없음을 의미합니다.

기본값은 8입니다.

LOGLOAD
한 체크포인트의 시작과 다음 체크포인트의 시작 간에 IBM MQ에서 기록하는 로그 레코드 수를 지정합니다. IBM MQ는 지정하는 레코드 수만큼 기록된 이후에 새 체크포인트를 시작합니다.

200 - 16,000,000 범위의 값을 지정하십시오.

기본값은 500,000입니다.

값이 클수록 IBM MQ의 성능은 더 좋습니다. 그러나 매개변수가 큰 값으로 설정되면 재시작이 더 오래 걸립니다.

제안되는 설정값:

테스트 시스템	10000
프로덕션 시스템	500 000 프로덕션 시스템에서, 제공된 기본값으로 인해 너무 높은 체크포인트 빈도가 나타날 수 있습니다.

LOGLOAD의 값은 큐 관리자 체크포인트의 빈도를 판별합니다. 너무 큰 값은 체크포인트 간에 많은 양의 데이터가 로그에 기록되어 실패 후의 큐 관리자 정방향 복구 재시작 시간이 길어짐을 의미합니다. 너무 작은 값은 최대 로드 중에 체크포인트를 너무 자주 발생시켜 응답 시간과 프로세서 사용에 좋지 않은 영향을 줍니다.

LOGLOAD에 대해 500 000의 초기값이 제안됩니다. 초당 100 메시지의 1KB 지속 메시지 비율(즉, 커미트를 갖는 100개의 `MQPUT` 및 커미트를 갖는 100개의 `MQGET`)의 경우 체크포인트 사이의 간격은 대략 5분입니다.

참고: 이는 지침으로만 사용되며 이 매개변수에 대한 최적의 값은 개별 시스템의 특성에 종속됩니다.

MULCCAPT
MULC(Measured Usage License Charging)에서 데이터를 수집하는 데 사용되는 알고리즘을 지정합니다.

STANDARD
MULC는 IBM MQ API MQCONN 호출부터 IBM MQ API MQDISC 호출까지의 시간을 바탕으로 합니다.

REFINED
MULC는 IBM MQ API 호출의 시작부터 IBM MQ API 호출의 종료까지의 시간을 바탕으로 합니다.

기본값은 STANDARD입니다.

OTMACON

OTMA 매개변수입니다. 이 키워드는 다음 5개의 위치 매개변수를 사용합니다.

OTMACON = (Group , Member , Druexit , Age , Tpipepfx)

그룹

이것은 IBM MQ의 특정 인스턴스가 속하는 XCF 그룹의 이름입니다.

이름의 길이는 1 - 8자가 될 수 있으며 대문자로 입력되어야 합니다.

기본값은 공백이며, 이는 IBM MQ가 XCF 그룹을 조인하려고 시도하지 않아야 함을 표시합니다.

Member

이것은 XCF 그룹 내에 있는 IBM MQ의 이 특정 인스턴스의 멤버 이름입니다.

이름의 길이는 1 - 16자 사이일 수 있으며 대문자로 입력해야 합니다.

기본값은 4자의 큐 관리자 이름입니다.

Druexit

이것은 IMS가 실행할 OTMA 목적지 해석 사용자 엑시트의 이름을 지정합니다.

길이는 1 - 8자 사이일 수 있습니다.

기본값은 DFSYDRU0입니다.

이 매개변수는 선택사항입니다. IBM MQ가 IBM MQ에 의해 시작되지 않은 IMS 애플리케이션에서 메시지를 수신할 경우에는 필수입니다. 이름은 IMS 시스템에서 코드화된 목적지 해석 사용자 엑시트에 대응해야 합니다. 자세한 정보는 693 페이지의 『IMS에서 OTMA 엑시트 사용』의 내용을 참조하십시오.

유효 기간

이것은 IBM MQ의 사용자 ID가 IMS에 의해 이전에 확인된 것으로 간주되는 시간(초)을 나타냅니다.

0 - 2147483647의 범위에 있을 수 있습니다.

기본값은 2 147 483 647입니다.

메인프레임에서 보안 캐시 설정의 일관성을 유지하기 위해 ALTER SECURITY 명령의 `interval` 매개변수와 함께 이 매개변수를 설정하는 것이 좋습니다.

Tpipepfx

이는 Tpipe 이름에 사용될 접두부를 표시합니다.

세 개의 문자로 구성됩니다. 첫 번째 문자는 A-Z 범위에 있고, 후속 문자는 A-Z 또는 0-9입니다. 기본값은 CSQ입니다.

IBM MQ가 Tpipe를 작성할 때마다 사용됩니다. 이름의 나머지는 IBM MQ에 의해 지정됩니다. IBM MQ에 의해 작성된 Tpipe에 대해서는 전체 Tpipe 이름을 설정할 수 없습니다.

QINDXBLD

큐 관리자 재시작이 모든 큐 색인을 다시 빌드할 때까지 대기할 것인지 또는 모든 색인을 다시 빌드하기 전에 완료할 것인지 여부를 판별합니다.

WAIT

큐 관리자 재시작은 모든 큐 색인 빌드가 완료되길 기다립니다. 이는 애플리케이션이 큐 관리자에 연결하기 전에 모든 색인이 작성되므로 색인이 작성되는 동안에는 정상 IBM MQ API 처리 중에 애플리케이션이 지연되지 않음을 의미합니다.

기본값입니다.

NOWAIT

큐 관리자는 모든 큐 색인 빌드가 완료되기 전에 재시작될 수 있습니다.

QMCCSID

큐 관리자(및 분산 큐잉)에서 사용할 기본 코드화 문자 세트 ID를 지정합니다.

0 - 65535 범위의 값을 지정하십시오. 이 값은 자국어(NL)에 선택한 언어에 대한 기본 z/OS 코드 페이지로 나열된 EBCDIC 코드 페이지를 나타내야 합니다.

0(기본값)은 현재 설정된 CCSID를 사용함을 의미하며, 설정된 것이 없으면 CCSID 500을 사용함을 의미합니다. 이는 CCSID를 명시적으로 0이 아닌 값으로 설정한 경우에는 QMCCSID를 0으로 설정하여 이를 재설정할 수 없음을 의미합니다. 사용자는 이제 올바른 0이 아닌 CCSID를 사용해야 합니다. QMCCSID가 0인 경우 DISPLAY QMGR CCSID 명령을 실행하여 실제로 사용 중인 CCSID를 확인할 수 있습니다.

QSGDATA

큐 공유 그룹 데이터입니다. 이 키워드는 다음 5개의 위치 매개변수를 사용합니다.

QSGDATA=(Qsgname,Dsgname,Db2name,Db2serv ,Db2blob)

Qsgname

큐 관리자가 속한 큐 공유 그룹의 이름입니다.

올바른 문자에 대해서는 IBM MQ 오브젝트 이름 지정 규칙을 참조하십시오. 이름:

- 1 - 4자의 길이를 가질 수 있습니다.
- 숫자로 시작하지 않아야 합니다.
- @으로 끝나지 않아야 합니다.

  이는 구현 이유로 인해 네 자 미만의 이름이 내부적으로 @ 기호로 채워지기 때문입니다.

기본값은 공백이며, 이는 큐 관리자가 큐 공유 그룹의 멤버가 아님을 의미합니다.

Dsgname

이것은 큐 관리자가 연결할 Db2 데이터 공유 그룹의 이름입니다.

이름의 길이는 1 - 8자가 될 수 있으며 대문자로 입력되어야 합니다.

기본값은 공백이며, 큐 공유 그룹을 사용 중이 아님을 의미합니다.

Db2name

이것은 큐 관리자가 연결할 Db2 서브시스템 또는 그룹 첨부의 이름입니다.

길이는 1 - 4자 사이일 수 있으며, 대문자로 입력해야 합니다.

기본값은 공백이며, 큐 공유 그룹을 사용 중이 아님을 의미합니다.

참고: Db2 서브시스템 (또는 그룹 첨부) 은 `Dsgname`에 지정된 Db2 데이터 공유 그룹에 있어야 하며, 모든 큐 관리자는 동일한 Db2 데이터 공유 그룹을 지정해야 합니다.

Db2serv

이것은 Db2 액세스에 사용되는 서버 태스크의 수입니다.

4 - 10 범위의 값이 될 수 있습니다.

기본값은 4입니다.

Db2blob

이것은 2진 대형 오브젝트(BLOB) 액세스에 사용되는 Db2 태스크의 수입니다.

4 - 10 범위의 값이 될 수 있습니다.

기본값은 4입니다.

이름 매개변수(즉, **Qsgname**, **Dsgname** 또는 **Db2name**) 중 하나를 지정하는 경우 다른 이름에 대한 값을 입력해야 하며, 그렇지 않으면 IBM MQ가 실패합니다.

RESAUDIT

RACF 감사 레코드가 연결 처리 중에 수행되는 RESLEVEL 보안 검사에 대해 작성되는지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

아니오

RESLEVEL 감사가 수행되지 않습니다.

YES

RESLEVEL 감사가 수행됩니다.

디폴트는 YES입니다.

ROUTCDE

MQSC 명령에 대한 직접 응답에서 전송되지 않는 메시지에 지정되는 기본 z/OS 메시지 라우팅 코드를 지정합니다.

다음 중 하나를 지정하십시오.

1. 1 - 16 범위(경계값 포함)의 값입니다.
2. 쉼표로 분리되며 괄호로 묶인 값의 목록입니다. 각 값은 1 - 16 범위(경계값 포함)의 값이어야 합니다.

기본값은 1입니다.

z/OS 라우팅 코드에 대한 자세한 정보는 *z/OS MVS* 라우팅 및 디스크립터 코드 매뉴얼의 볼륨 중 하나에 있는 메시지 설명 을 참조하십시오.

SERVICE

이 필드는 IBM에서 사용하기 위해 예약됩니다.

SMFACCT

큐 관리자가 시작되면 IBM MQ가 회계 데이터를 SMF에 자동으로 송신할지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

아니오

회계 데이터 수집을 자동으로 시작하지 않습니다.

YES

기본 클래스 1에 대해 회계 데이터 수집을 자동으로 시작합니다.

정수

1 - 4 범위의 계정이 자동으로 수집되는 클래스 목록입니다.

기본값은 NO입니다.

SMFSTAT

큐 관리자를 시작할 때 SMF 통계를 자동 수집할 것인지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

아니오

통계 수집을 자동으로 시작하지 않습니다.

YES

기본 클래스 1에 대해 통계 수집을 자동으로 시작합니다.

정수

1 - 4 범위의 통계가 자동으로 수집되는 클래스 목록입니다.

기본값은 NO입니다.

SPLCAP

보안 정책 기능을 사용하면 메시지가 큐에 기록되거나 큐에서 읽히므로 메시지가 서명되는지 암호화되는지 제어하는 정책을 통해 더 높은 수준의 메시지 보안이 가능하게 됩니다.

사용은 SDRQAUTH 라이브러리에서 사용 설정 모듈을 제공하는 별도로 설치된 제품인 Advanced Message Security(AMS)에 의해 라이센스가 부여됩니다.

보안 정책 처리는 다음 값 중 하나로 SPLCAP를 구성함으로써 이 큐 관리자에 사용 가능합니다.

아니오

큐에 대한 메시지 보안 정책을 구현하는 기능은 큐 관리자 초기화 중에 사용으로 설정되지 않습니다.

YES

큐 관리자 초기화 중에 메시지 보안 기능이 사용으로 설정됩니다.

이 제어가 설정되면 큐 관리자가 초기화 동안 SDRQAUTH로부터 모듈을 사용 가능하게 하는 라이센스를 로드하려고 시도하며 추가 주소 공간(AMSM)을 시작합니다.

큐 관리자는 AMS가 라이센스되지 않는 한 시작되지 않으며, 메시지 보안에 대한 필수 구성은 준비가 됩니다.

기본값은 NO입니다.

STATIME
연속된 통계 수집 사이의 기본 시간(분)을 지정합니다.

0 - 1440 범위의 숫자를 지정하십시오.

값을 0으로 지정하면, SMF 데이터 수집 브로드캐스트에서 통계 데이터와 계정 데이터가 둘 다 수집됩니다. 설정에 대한 정보는 System Management Facility 사용을 참조하십시오.

기본값은 30입니다.

TRACSTR
글로벌 추적의 자동 시작 여부를 지정합니다.

다음 중 하나를 지정하십시오.

아니오
글로벌 추적을 자동으로 시작하지 않습니다.

YES
기본 클래스인 클래스 1에 대해 글로벌 추적을 자동으로 시작합니다.

정수
1 - 4 범위에 있고 글로벌 추적이 자동으로 시작되는 클래스의 목록입니다.

*****
모든 클래스에서 글로벌 추적을 자동으로 시작합니다.

매크로에서 키워드를 지정하지 않는 경우 기본값은 NO입니다.

참고: 제공되는 기본 시스템 매개변수 로드 모듈(CSQZPARM)에서 TRACSTR=YES가 설정됩니다(어셈블러 모듈 CSQFSYSP에 설정됨). 추적을 자동으로 시작하지 않으려면 사용자 고유의 시스템 매개변수 모듈을 작성하거나 큐 관리자를 시작한 후 STOP TRACE 명령을 실행하십시오.

STOP TRACE 명령에 대한 자세한 내용은 STOP TRACE를 참조하십시오.

TRACTBL
글로벌 추적 기능이 IBM MQ 추적 레코드를 저장하는 추적 테이블의 기본 크기(4KB 블록)를 지정합니다.

1 - 999 범위의 값을 지정하십시오.

기본값은 99입니다. 이는 396KB와 동일합니다.

참고: 추적 테이블의 스토리지는 ECSA에서 할당됩니다. 따라서 주의하여 이 값을 선택해야 합니다.

WLMTIME
WLM 관리 큐에 대한 색인의 각 스캔 사이의 시간(WLMTIMU의 값에 따라 분 또는 초)을 지정합니다.

1 - 9999 범위의 값을 지정하십시오.

기본값은 30입니다.

WLMTIMU
WLMTIME 매개변수에 사용된 시간 단위입니다.

다음 중 하나를 지정하십시오.

MINS
WLMTIME은 분 수를 나타냅니다.

SECS
WLMTIME은 초 수를 나타냅니다.

기본값은 MINS입니다.

관련 참조

623 페이지의 『CSQ6LOGP 사용』
CSQ6LOGP를 사용하여 로깅 옵션을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

626 페이지의 『CSQ6ARVP 사용』
CSQ6ARVP를 사용하여 아카이브 환경을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

z/OS *CSQ6LOGP* 사용

CSQ6LOGP를 사용하여 로깅 옵션을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

CSQ6LOGP를 사용하여 로깅 옵션을 설정하십시오.

CSQ6LOGP의 기본 매개변수 및 SET LOG 명령을 사용하여 각 매개변수를 변경할 수 있는지 여부는 CSQ6LOGP 매개변수의 기본값에 표시됩니다. 이러한 값을 변경해야 하는 경우 매개변수의 자세한 설명을 참조하십시오.

표 *45. CSQ6LOGP* 매개변수의 기본값

매개변수	설명	기본값	SET 명령
COMPLOG	로그 압축이 사용 가능한지 여부를 제어합니다.	NONE	X
DEALLCT	할당 취소하기 전에 아카이브 테이프 단위가 사용되지 않는 상태로 있는 시간입니다.	영(0)	X
INBUFF	활성 및 아카이브 로그 데이터 세트에 대한 입력 버퍼 스토리지의 크기	60KB	-
MAXARCH	기록할 수 있는 최대 아카이브 로그 볼륨 수	500	X
MAXCNOFF	병렬로 실행될 수 있는 CSQJOFF7 오프로드 태스크의 최대 수	31	-
MAXRTU	동시에 아카이브 로그 테이프 볼륨을 읽는 데 할당되는 최대 전용 테이프 장치 수	2	X
OFFLOAD	아카이브 설정 여부.	YES(ON)	-
OUTBUFF	활성 및 아카이브 로그 데이터 세트에 대한 출력 버퍼 스토리지의 크기	4,000KB	-
TWOACTV	단일 또는 이중 활성 로깅	YES(이중)	-
TWOARCH	단일 또는 이중 아카이브 로깅	YES(이중)	-
TWOBSDS	단일 또는 이중 BSDS	YES(이중 BSDS)	-
WRTHRSH	활성 로그 데이터 세트에 기록하기 전에 채울 출력 버퍼 수	20	X
ZHYWRITE	zHyperWrite 기능의 사용 여부를 지정합니다.	아니오	-

COMPLOG

로그 압축이 사용 가능한지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

NONE

로그 압축이 사용 가능하지 않습니다.

RLE

로그 압축이 실행 길이 인코딩을 사용하여 사용으로 설정됩니다.

ANY

큐 관리자가 로그 레코드를 가장 많이 압축하는 압축 알고리즘을 선택합니다. 이 옵션을 사용하면 RLE 압축이 수행됩니다.

기본값은 NONE입니다.

로그 압축에 대한 자세한 내용은 로그 압축을 참조하십시오.

DEALLCT

아카이브 읽기 테이프 장치가 할당 취소되기 전까지 사용하지 않는 상태로 유지될 수 있는 시간(분)을 지정합니다.

다음 중 하나를 지정하십시오.

- 0 - 1440 범위의 시간(분)
- NOLIMIT

1440 또는 NOLIMIT를 지정하면 테이프 장치가 할당 취소되지 않습니다.

디폴트는 0입니다.

아카이브 로그 데이터를 테이프에서 읽고 있을 때, IBM MQ가 복수 읽기 애플리케이션에 대한 테이프 처리를 최적화하도록 허용하기에 충분히 높은 값으로 설정할 것을 권장합니다.

INBUFF

복구 중 활성 및 아카이브 로그를 읽는 입력 버퍼의 크기(KB)를 지정합니다. 28 - 60 범위의 10진수를 사용하십시오. 지정된 값은 4의 배수로 올림됩니다.

기본값은 60KB입니다.

제안되는 설정값:

테스트 시스템	28KB
프로덕션 시스템	60KB 최상의 로그 읽기 성능을 취해 이를 최대값으로 설정하십시오.

MAXARCH

BSDS에 기록될 수 있는 최대 아카이브 로그 볼륨 수를 지정합니다. 이 수를 초과하면, BSDS 시작 시 기록이 다시 시작됩니다.

10 - 1000 범위의 10진수를 사용하십시오.

기본값은 500입니다.

제안되는 설정값:

테스트 시스템	500(기본값)
프로덕션 시스템	1 000 BSDS가 가능한 한 많은 로그를 기록할 수 있도록 이를 최대값으로 설정하십시오.

로그 및 BSDS에 대한 정보는 IBM MQ 자원 관리를 참조하십시오.

MAXCNOFF

병렬로 실행될 수 있는 CSQJOFF7 오프로드 태스크 수를 지정합니다.

이것은 큐 관리자가 모든 사용 가능한 테이프 장치를 사용하지 않도록 튜닝되도록 허용합니다.

대신 큐 관리자는 임의의 새 아카이브 데이터 세트를 할당하려고 시도하기 전에 CSQJOFF7 오프로드 태스크가 완료되었을 때까지 기다립니다.

큐 관리자가 테이프로 아카이브 중인 경우 동시 테이프 요청 수가 사용 가능한 테이프 장치의 수와 동일하거나 초과하지 않도록 이 매개변수를 설정하십시오. 그렇지 않으면 시스템이 정지할 수 있습니다.

이중 아카이빙이 사용 중인 경우에는 각 오프로드 태스크가 아카이브를 둘 다 수행하므로 매개변수가 그에 따라서 설정되어야 함을 주의하십시오. 예를 들어 큐 관리자가 테이프에 이중 아카이브 중인 경우, MAXCNOFF=2의 값은 최고 2개의 활성 로그가 4개의 테이프에 동시에 아카이브되도록 허용합니다.

여러 큐 관리자가 테이프 장치를 공유 중인 경우 각 큐 관리자에 대한 MAXCNOFF를 그에 따라서 설정해야 합니다.

기본값은 31입니다.

1 - 31 범위에 있는 값을 지정하십시오.

MAXRTU

동시에 아카이브 로그 테이프 볼륨을 읽는 데 설정할 수 있는 최대 전용 테이프 장치 수를 지정합니다.

이 매개변수와 DEALLCT 매개변수는 IBM MQ가 테이프 디바이스에서 아카이브 로그 읽기를 최적화할 수 있게 합니다.

1 - 99 범위의 값을 지정하십시오.

기본은 2입니다.

IBM MQ에 사용 가능한 테이프 장치의 수보다 최소한 하나 작게 값을 설정할 것을 권장합니다. 그렇지 않으면 오프로드 프로세스가 지연될 수 있으며, 시스템 성능에 영향을 줄 수 있습니다. 아카이브 로그 처리 중에 최대 처리량을 확보하려면 오프로드 처리를 위해 최소 하나의 테이프 장치만 필요하다는 점을 명심하고 이 옵션에 대해 가능한 가장 큰 값을 지정하십시오.

OFFLOAD
아카이브를 사용할지 또는 사용하지 않을지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

YES
아카이브가 작동됩니다.

아니오
아카이브가 작동 중지됩니다.

디폴트는 YES입니다.

주의: 테스트 환경에서 작업하지 않는 한, 아카이브를 끄지 **마십시오**. 작동을 중단하면, 시스템 또는 트랜잭션 실패시 데이터가 복구된다고 보장할 수 없습니다.

OUTBUFF
활성 및 아카이브 로그 데이터 세트 쓰기를 위한 출력 버퍼를 위해 IBM MQ가 사용할 스토리지의 총 크기를 KB 단위로 지정합니다. 각 출력 버퍼는 4KB입니다.

매개변수의 범위는 128 - 4000이어야 합니다. 지정된 값은 4의 배수로 올림됩니다. 40에서 128 사이의 값은 호환성 이유로 승인되며 128의 값으로 처리됩니다.

기본값은 4000KB입니다.

제안되는 설정값:

테스트 시스템	400KB
프로덕션 시스템	4,000KB 로그 출력 버퍼가 부족하지 않도록 하려면 이 값을 최대값으로 설정하십시오.

TWOACTV
단일 또는 이중 활성 로깅을 지정합니다.

다음 중 하나를 지정하십시오.

아니오
단일 활성 로그

YES
이중 활성 로그

디폴트는 YES입니다.

단일 및 이중 로깅의 사용에 관한 자세한 정보는 IBM MQ 자원 관리를 참조하십시오.

TWOARCH
IBM MQ가 활성 로그가 오프로드될 때 생성하는 아카이브 로그의 수를 지정합니다.

다음 중 하나를 지정하십시오.

아니오
단일 아카이브 로그

YES
이중 아카이브 로그

디폴트는 YES입니다.

제안되는 설정값:

테스트 시스템	아니오
프로덕션 시스템	YES(기본값)

단일 및 이중 로깅의 사용에 관한 자세한 정보는 IBM MQ 자원 관리를 참조하십시오.

TWOBSDS
부트스트랩 데이터 세트 수를 지정합니다.

다음 중 하나를 지정하십시오.

아니오
단일 BSDS

YES
이중 BSDS

디폴트는 YES입니다.

단일 및 이중 로깅의 사용에 관한 자세한 정보는 IBM MQ 자원 관리를 참조하십시오.

WRTHRSH
활성 로그 데이터 세트에 기록되기 전에 파일화될 4KB 출력 버퍼의 수를 지정합니다.

버퍼의 수가 많을수록 쓰기 발생이 적어지므로 IBM MQ의 성능은 향상됩니다. 확약점과 같은 중대한 이벤트가 발생할 경우 이 수에 도달하기 전에 버퍼가 기록될 수 있습니다.

버퍼의 수를 1 - 256으로 지정하십시오.

기본값은 20입니다.

ZHYWRITE
zHyperWrite 기능의 사용 여부를 지정합니다.

가능한 값은 다음과 같습니다.

아니오
zHyperWrite가 사용으로 설정되지 않습니다.

주의: zHyperWrite는 IBM MQ 9.0에서 사용 가능하지 않으므로 *NO* 값만 허용됩니다.

관련 참조

615 페이지의 『CSQ6SYSP 사용』
이 토픽을 CSQ6SYSP를 사용하여 시스템 매개변수를 설정하는 방법에 대한 참조로 사용하십시오.

626 페이지의 『CSQ6ARVP 사용』
CSQ6ARVP를 사용하여 아카이브 환경을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

z/OS *CSQ6ARVP* 사용

CSQ6ARVP를 사용하여 아카이브 환경을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

아카이브 환경을 설정하려면 CSQ6ARVP를 사용하십시오.

CSQ6ARVP의 기본 매개변수 및 SET ARCHIVE 명령을 사용하여 각 매개변수를 대체할 수 있는지 여부가 626 페이지의 표 46에 표시됩니다. 이러한 값을 변경해야 하는 경우 매개변수의 자세한 설명을 참조하십시오. 스토리지 계획에 대한 자세한 정보는 z/OS을 참조하십시오.

표 *46. CSQ6ARVP* 매개변수의 기본값

매개변수	설명	기본값	SET 명령
ALCUNIT	1차 및 2차 공간 할당이 작성되는 단위입니다.	BLK(블록)	X

표 *46. CSQ6ARVP* 매개변수의 기본값 *(*계속*)*

매개변수	설명	기본값	SET 명령
ARCPFX1	첫 번째 아카이브 로그 데이터 세트 이름의 접두부입니다.	CSQARC1	X
ARCPFX2	두 번째 아카이브 로그 데이터 세트 이름의 접두부입니다.	CSQARC2	X
ARCRETN	아카이브 로그 데이터 세트의 보존 기간(일)입니다.	9999	X
ARCWRTC	아카이브 로그 데이터 세트에 관한 연산자에 대한 메시지의 라우트 코드 목록입니다.	1,3,4	X
ARCWTOR	아카이브 로그 데이터 세트를 마운트하기 전에 연산자에 메시지를 보내고 응답을 기다리는지 여부입니다.	YES	X
BLKSIZE	아카이브 로그 데이터 세트의 블록 크기입니다.	28 672	X
CATALOG	ICF에서 아카이브 로그 데이터 세트를 카탈로깅하는지 여부입니다.	아니오	X
COMPACT	아카이브 로그 데이터 세트를 압축해야 하는지 여부입니다.	아니오	X
PRIQTY	DASD 데이터 세트의 1차 공간 할당입니다.	25 715	X
PROTECT	데이터 세트를 작성할 때 ESM 프로파일에서 아카이브 로그 데이터 세트를 보호하는지 여부입니다.	아니오	X
정지	MODE(QUIESCE)의 ARCHIVE LOG가 지정된 경우 일시정지에 대해 허용되는 최대 시간(초)입니다.	5	X
SECQTY	DASD 데이터 세트에 2차 공간 할당. 사용할 단위는 ALCUNIT 매개변수를 참조하십시오.	540	X
TSTAMP	아카이브 데이터 세트 이름이 시간소인을 포함해야 하는지 여부입니다.	아니오	X
UNIT	아카이브 로그 데이터 세트의 첫 번째 사본이 저장되는 디바이스 유형 또는 단위 이름입니다.	TAPE	X
UNIT2	아카이브 로그 데이터 세트의 2차 사본이 저장되는 디바이스 유형 또는 단위 이름입니다.	Blank	X

ALCUNIT

1차 및 2차 공간 할당이 수행되는 단위를 지정합니다.

다음 중 하나를 지정하십시오.

CYL

실린더

TRK

트랙

BLK

블록

디바이스 유형과는 독립되어 있으므로 BLK를 사용하는 것이 좋습니다.

기본값은 BLK입니다.

아카이브 DASD 볼륨의 여유 공간이 단편화된 경우 더 작은 1차 범위를 지정하고 2차 범위로 확장하는 것이 좋습니다. 활성 로그의 공간 할당에 대한 자세한 정보는 로그 아카이브 저장영역 계획을 참조하십시오.

ARCPFX1
첫 번째 아카이브 로그 데이터 세트 이름의 접두부를 지정합니다.

데이터 세트 이름 지정 방법에 대한 설명 및 ARCPFX1 길이에 대한 제한사항은 TSTAMP 매개변수를 참조하십시오.

이 매개변수는 공백으로 둘 수 없습니다.

기본값은 CSQARC1입니다.

이 접두부를 갖는 아카이브 로그를 작성하려면 IBM MQ 큐 관리자 주소 공간과 연관된 사용자 ID를 권한 부여해야 할 수 있습니다.

ARCPFX2
두 번째 아카이브 로그 데이터 세트 이름의 접두부를 지정합니다.

데이터 세트 이름 지정 방법에 대한 설명 및 ARCPFX2 길이에 대한 제한사항은 TSTAMP 매개변수를 참조하십시오.

이 매개변수는 TWOARCH 매개변수가 NO로 지정된 경우에도 공백일 수 없습니다.

기본값은 CSQARC2입니다.

이 접두부를 갖는 아카이브 로그를 작성하려면 IBM MQ 큐 관리자 주소 공간과 연관된 사용자 ID를 권한 부여해야 할 수 있습니다.

ARCRETN
아카이브 로그 데이터 세트가 작성될 때 사용할 일 단위의 보존 기간을 지정합니다.

매개변수는 0 - 9999 범위에 있어야 합니다.

기본값은 9999입니다.

제안되는 설정값:

테스트 시스템	3 테스트 시스템에서 아카이브 로그는 오랫동안 필요하지 않을 수도 있습니다.
프로덕션 시스템	9 999(기본값) 자동 아카이브 로그 삭제를 효과적으로 전환하려면 이 값을 높게 설정하십시오.

아카이브 로그 데이터 세트를 제거하는 데 관한 자세한 정보는 아카이브 로그 데이터 세트 제거를 참조하십시오.

ARCWRTC
아카이브 로그 데이터 세트에 대한 메시지의 z/OS 라우팅 코드 목록을 운영자에게 지정합니다. ARCWTOR이 NO로 설정된 경우 이 필드는 무시됩니다.

1 - 16 범위 내의 값으로 각각 최대 14개의 라우팅 코드를 지정하십시오. 최소한 하나의 코드를 지정해야 합니다. 목록의 코드는 공백 없이 쉼표로 구분하십시오.

기본값은 값의 목록(1,3,4)입니다.

z/OS 라우팅 코드에 대한 자세한 정보는 *z/OS MVS* 라우팅 및 디스크립터 코드 매뉴얼의 볼륨 중 하나에 있는 메시지 설명을 참조하십시오.

ARCWTOR
메시지가 운영자에게 송신되는지와 아카이브 로그 데이터 세트 마운트를 시도하기 전에 응답을 수신하는지 여부를 지정합니다.

기타 IBM MQ 사용자는 데이터 세트가 마운트될 때까지 강제로 대기해야 할 수 있지만, IBM MQ가 메시지에 대한 응답을 대기할 동안 영향을 받지 않습니다.

다음 중 하나를 지정하십시오.

YES
: 디바이스가 아카이브 로그 데이터 세트를 마운트하는 데 긴 시간이 필요합니다. 예로는 테이프 드라이브가 있습니다.

아니오
: 디바이스가 긴 지연 시간을 갖지 않습니다 (예: DASD).

디폴트는 YES입니다.

제안되는 설정값:

테스트 시스템	아니오
프로덕션 시스템	YES(기본값) 이것은 조작 프로시저에 종속됩니다. 테이프 로봇이 사용되는 경우 NO가 더 적절할 수 있습니다.

BLKSIZE

아카이브 로그 데이터 세트의 블록 크기를 지정합니다. 지정한 블록 크기는 UNIT 매개변수에서 지정한 디바이스 유형과 호환 가능해야 합니다.

매개변수는 4,097 - 28,672 범위에 있어야 합니다. 지정하는 값은 4 096의 배수로 반올림됩니다.

기본값은 28 672입니다.

제공되는 경우, 이 매개변수는 SMS(storage management subsystem) 데이터 클래스 블록 크기로 대체됩니다.

아카이브 로그 데이터 세트가 DASD에 기록된 경우 각 트랙당 2개의 블록을 허용하는 최대 블록 크기를 선택하는 것이 좋습니다. 예를 들어 3390 디바이스의 경우 블록 크기 24 576을 사용해야 합니다.

아카이브 로그 데이터 세트가 테이프에 기록된 경우 가장 큰 가능한 블록 크기를 지정하면 아카이브 로그를 읽는 속도가 향상됩니다. 28 672의 블록 크기를 사용해야 합니다.

제안되는 설정값:

테스트 시스템	아카이브 로그에 사용되는 매체에 따라서 블록 크기 권장사항을 사용하십시오. 디스크의 경우 24 576, 테이프의 경우 28 672입니다.
프로덕션 시스템	아카이브 로그에 사용되는 매체에 따라서 블록 크기 권장사항을 사용하십시오. 디스크의 경우 24 576, 테이프의 경우 28 672입니다.

CATALOG

아카이브 로그 데이터 세트가 1차 ICF(integrated catalog facility) 카탈로그에서 카탈로그되었는지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

아니오
: 아카이브 로그 데이터 세트가 카탈로그화되지 않음

YES
: 아카이브 로그 데이터 세트가 카탈로그화됨

기본값은 NO입니다.

DASD에 할당된 모든 아카이브 로그 데이터 세트는 카탈로그화되어야 합니다. NO로 설정된 CATALOG 매개변수를 사용하여 DASD에 아카이브하는 경우, 아카이브 로그 데이터 세트가 할당되고 IBM MQ가 데이터 세트를 카탈로깅할 때마다 CSQJ072E 메시지가 표시됩니다.

제안되는 설정값:

테스트 시스템	YES
프로덕션 시스템	아카이브가 DASD에서 할당될 때 YES

COMPACT

아카이브 로그에 기록된 데이터가 압축될 것인지 여부를 지정합니다. 이 옵션은 IDRC(improved data recording capability) 기능이 있는 3480 또는 3490 디바이스에만 적용됩니다. 이 기능이 설정되면 테이프 제어 장치의 하드웨어가 정상보다 훨씬 높은 밀도로 데이터를 쓰므로 각 볼륨에 보다 많은 데이터가 존재할 수 있습니다. 3490E를 제외하고 3490 기본 모델 또는 IDRC 기능을 사용하여 3480 디바이스를 사용하지 않는 경우 NO를 지정하십시오. 데이터를 압축하려면, YES를 지정하십시오.

다음 중 하나를 지정하십시오.

아니오

데이터 세트를 압축하지 않음

YES

데이터 세트를 압축함

기본값은 NO입니다.

YES로 지정하면 성능이 떨어집니다. 또한 IDRC 기능을 지원하는 디바이스를 사용해서만 테이프에 압축된 데이터를 읽을 수 있다는 점을 인식하십시오. 이는 원격 복구를 위해 다른 사이트로 아카이브 테이프를 전송해야 하는 경우 문제가 될 수 있습니다.

제안되는 설정값:

테스트 시스템	적용할 수 없음
프로덕션 시스템	NO(기본값) 이는 3480 및 3490 IDR 압축에만 적용됩니다. 이를 YES로 설정하면 복구 및 재시작 중에 아카이브 로그 읽기 성능을 떨어뜨릴 수 있습니다. 그러나 테이프에 쓰는 작업에 영향을 주지 않습니다.

PRIQTY

ALCUNIT에서 DASD 데이터 세트에 1차 공간 할당을 지정합니다.

값은 0보다 커야 합니다.

기본값은 25 715입니다.

이 값은 어느 쪽이 더 크든지 로그 데이터 세트나 해당 BSDS 복사에 충분해야 합니다. 필요한 값을 판별하려면 다음 프로시저를 수행하십시오.

1. 611 페이지의 『부트스트랩 및 로그 데이터 세트 작성』에 설명된 대로 할당된 활성 로그 레코드 수 (c)를 판별하십시오.
2. 각 아카이브 로그 블록에 있는 4096바이트 블록 수를 판별하십시오.

   ```
   d = BLKSIZE / 4096
   ```

   여기서 BLKSIZE는 올림된 값입니다.
3. ALCUNIT=BLK인 경우

   ```
   PRIQTY = INT(c / d) + 1
   ```

   여기서 INT는 정수로 내림된 값을 의미합니다.

   ALCUNIT=TRK인 경우

   ```
   PRIQTY = INT(c / (d * INT(e/BLKSIZE))) + 1
   ```

여기서 e는 각 트랙에 대한 바이트 수(3390 디바이스의 경우 56664)이며 INT는 정수로 내림된 값임을 의미합니다.

ALCUNIT=CYL인 경우

```
PRIQTY = INT(c / (d * INT(e/BLKSIZE) * f)) + 1
```

여기서 f는 각 실린더의 트랙 수(3390 디바이스의 경우 15)이며 INT는 정수로 내림된 값임을 의미합니다.

로그 및 아카이브 데이터 세트를 작성하는 데 드는 크기에 대한 정보는 611 페이지의 『부트스트랩 및 로그 데이터 세트 작성』 및 612 페이지의 『페이지 세트 정의』의 내용을 참조하십시오.

제안되는 설정값:

테스트 시스템	1 680 전체 활성 로그를 보유하는 데 충분합니다. 즉 다음과 같습니다. `10  080 / 6 = 1  680 blocks`
프로덕션 시스템	테이프에 아카이브하는 경우 적용할 수 없습니다.

아카이브 DASD 볼륨의 여유 공간이 단편화된 경우 더 작은 1차 범위를 지정하고 2차 범위로 확장하는 것이 좋습니다. 활성 로그에 대한 공간 할당에 대한 자세한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

PROTECT

데이터 세트 작성시 분리 ESM(external security manager) 프로파일에서 아카이브 로그 데이터 세트가 보호되는지 여부를 지정합니다.

다음 중 하나를 지정하십시오.

아니오

프로파일이 작성되지 않습니다.

YES

로그를 오프로드할 때 개별 데이터 세트 프로파일을 작성합니다. YES를 지정할 경우:

- ESM 보호는 IBM MQ에 활성이어야 합니다.
- IBM MQ 큐 관리자 주소 공간과 연관된 사용자 ID는 이들 프로파일을 작성할 권한을 가져야 합니다.
- TAPEVOL 클래스는 테이프에 아카이브할 경우 활성화되어야 합니다.

그렇지 않으면, 오프로딩에 실패합니다.

기본값은 NO입니다.

정지

MODE(QUIESCE)를 지정한 상태에서 ARCHIVE LOG 명령이 실행될 때 일시정지에 허용되는 최대 시간(초)을 지정하십시오.

매개변수는 1 - 999 범위에 있어야 합니다.

기본값은 5입니다.

SECQTY

ALCUNIT에서 DASD 데이터 세트에 2차 공간 할당을 지정합니다. 보조 범위는 15번까지 할당될 수 있습니다. 자세한 내용은 *z/OS MVS JCL Reference* 및 *z/OS MVS JCL User's Guide*를 참조하십시오.

매개변수는 0보다 커야 합니다.

기본값은 540입니다.

TSTAMP

아카이브 로그 데이터 세트 이름에 시간 소인이 있는지를 지정합니다.

다음 중 하나를 지정하십시오.

아니오

이름에 시간 소인이 포함되지 않습니다. 아카이브 로그 데이터 세트는 다음과 같이 이름이 지정됩니다.

```
arcpfxi.A nnnnnnn
```

여기서 *arcpfxi*는 ARCPFX1 또는 ARCPFX2로 지정된 데이터 세트 이름 접두부입니다. *arcpfxi*는 최대 35자일 수 있습니다.

YES

이름에 시간 소인이 포함됩니다. 아카이브 로그 데이터 세트는 다음과 같이 이름이 지정됩니다.

```
arcpfxi.cyyddd.T hhmmsst.A nnnnnnn
```

여기서 *c*는 최대 1999년을 포함하는 연도의 경우 'E'이고 2000년 이후의 연도의 경우에는 'D'이며, *arcpfxi*는 ARCPFX1 또는 ARCPFX2로 지정된 데이터 세트 이름 접두부입니다. *arcpfxi*는 최대 19자일 수 있습니다.

EXT

이름에 시간 소인이 포함됩니다. 아카이브 로그 데이터 세트는 다음과 같이 이름이 지정됩니다.

```
arcpfxi.D yyyyddd.T hhmmsst.A nnnnnnn
```

여기서 *arcpfxi*는 ARCPFX1 또는 ARCPFX2로 지정된 데이터 세트 이름 접두부입니다. *arcpfxi*는 최대 17자일 수 있습니다.

기본값은 NO입니다.

UNIT

아카이브 로그 데이터 세트의 첫 번째 사본을 저장하는 데 사용되는 디바이스의 디바이스 유형 또는 장치 이름을 지정합니다.

1 - 8자 사이의 영숫자로 구성된 디바이스 유형 또는 단위 이름을 지정하십시오. 첫 번째 문자는 영문자여야 합니다.

이 매개변수는 공백일 수 없습니다.

기본값은 TAPE입니다.

DASD에 아카이브하는 경우, 제한된 볼륨 범위를 갖는 일반 디바이스 유형을 지정할 수 있습니다(예: UNIT=3390).

DASD에 아카이브하는 경우 다음을 확인하십시오.

- 1차 공간 할당이 활성 로그 데이터 세트의 모든 데이터를 포함하기에 충분히 큽니다.
- 아카이브 로그 데이터 세트 카탈로그 옵션(CATALOG)이 YES로 설정됩니다.
- BLKSIZE에 대해 적절한 값을 사용했습니다.

TAPE에 아카이브하는 경우 IBM MQ는 최대 20 볼륨으로 확장할 수 있습니다.

제안되는 설정값:

테스트 시스템	DASD
프로덕션 시스템	TAPE

아카이브 로그의 위치를 선택하는 방법에 대한 자세한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

UNIT2

아카이브 로그 데이터 세트의 두 번째 사본을 저장하는 데 사용되는 디바이스의 디바이스 유형 또는 장치 이름을 지정합니다.

1 - 8자 사이의 영숫자로 구성된 디바이스 유형 또는 단위 이름을 지정하십시오. 첫 번째 문자는 영문자여야 합니다. 이 매개변수가 공백이면, UNIT 매개변수의 값 세트가 사용됩니다.

기본값은 공백입니다.

관련 참조

615 페이지의 『CSQ6SYSP 사용』
이 토픽을 CSQ6SYSP를 사용하여 시스템 매개변수를 설정하는 방법에 대한 참조로 사용하십시오.

623 페이지의 『CSQ6LOGP 사용』
CSQ6LOGP를 사용하여 로깅 옵션을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

z/OS V 9.0.3 *CSQ6USGP* 사용

이 토픽을 CSQ6USGP를 사용하여 시스템 매개변수를 설정하는 방법에 대한 참조로 사용하십시오.

CSQ6USGP를 사용하여 제품 사용 기록을 제어하십시오.

CSQ6USGP에 대한 기본 매개변수가 633 페이지의 표 47에 표시됩니다. 이러한 값을 변경해야 하는 경우 매개변수의 자세한 설명을 참조하십시오.

주의: SET SYSTEM 명령을 사용하여 이 매개변수를 변경할 수 없습니다.

표 *47. CSQ6USGP* 매개변수의 기본값

매개변수	**설명**	**기본값**
QMGRPROD	큐 관리자 사용량을 기록할 제품	Blank
AMSPROD	AMS(Advanced Message Security) 사용량을 기록할 제품	Blank

QMGRPROD

큐 관리자 사용량을 기록할 제품을 지정합니다.

다음 중 하나를 지정하십시오.

MQ

큐 관리자 사용법은 제품 ID가 5655-MQ9인 독립형 IBM MQ for z/OS 제품으로 기록됩니다.

VUE

큐 관리자 사용법은 제품 ID가 5655-VU9인 독립형 IBM MQ for z/OS Value Unit Edition (VUE) 제품으로 기록됩니다.

ADVANCEDVUE

큐 관리자 사용법은 제품 ID가 5655-AV1인 IBM MQ Advanced for z/OS, Value Unit Edition 제품의 일부로 기록됩니다.

AMSPROD

AMS 사용량을 기록할 제품을 지정합니다(사용된 경우).

다음 중 하나를 지정하십시오.

AMS

AMS 사용량은 제품 ID가 5655-AM9인 독립형 Advanced Message Security for z/OS 제품으로 기록됩니다.

ADVANCED

AMS 사용량은 제품 ID가 5655-AV9인 IBM MQ Advanced for z/OS 제품의 일부로 기록됩니다.

ADVANCEDVUE

AMS 사용량은 제품 ID가 5655-AV1인 IBM MQ Advanced for z/OS, Value Unit Edition 제품의 일부로 기록됩니다.

제품 사용 레코딩에 대한 자세한 정보는 제품 정보 보고를 참조하십시오.

관련 참조

615 페이지의 『CSQ6SYSP 사용』

이 토픽을 CSQ6SYSP를 사용하여 시스템 매개변수를 설정하는 방법에 대한 참조로 사용하십시오.

623 페이지의 『CSQ6LOGP 사용』
CSQ6LOGP를 사용하여 로깅 옵션을 지정하는 방법에 대한 참조로 이 토픽을 사용하십시오.

z/OS 채널 시작기 매개변수 조정

요구사항에 맞도록 채널 시작기를 사용자 정의하려면 ALTER QMGR을 사용하십시오.

- 필요에 따라 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션하는 경우 이 태스크를 수행해야 합니다.

많은 큐 관리자 속성이 분산 큐잉이 동작하는 방법을 제어합니다. MQSC 명령 ALTER QMGR을 사용하여 이 속성을 설정하십시오. 초기화 데이터 세트 샘플 thlqual.SCSQPROC(CSQ4INYG)에는 사용자 정의할 수 있는 일부 설정이 있습니다. 자세한 정보는 ALTER QMGR을 참조하십시오.

이 매개변수의 값은 채널 시작기를 시작할 때마다 일련의 메시지로 표시됩니다.

어댑터, 디스패처 및 최대 채널 수 사이의 관계

ALTER QMGR 매개변수인 CHIADAPS와 CHIDISPS는 채널 시작기가 사용하는 태스크 제어 블록(TCB)의 수를 정의합니다. CHIADAPS(어댑터) TCB가 큐 관리자에 대한 IBM MQ API 호출을 작성하는 데 사용됩니다. CHIDISPS(디스패처) TCB는 통신 네트워크에 대한 호출을 실행하는 데 사용합니다.

ALTER QMGR 매개변수 MAXCHL은 디스패처 TCB에 채널을 분배하는 데 영향을 미칩니다.

CHIDISPS

채널 수가 적은 경우 기본값을 사용하십시오.

각 프로세서에 대한 하나의 태스크가 시스템 성능을 최적화합니다. 디스패처 태스크는 CPU 집중적이므로, 원리는 스레드를 찾고 시작하는 데 걸리는 시간이 최소화되도록 가능한 작은 태스크를 사용 중으로 유지하는 것입니다.

CHIDISPS(20)이 100개가 넘는 채널을 갖는 시스템에 적합합니다. CHIDISPS(20)이 필요한 디스패처 TCB 수보다 많은 수인 경우 이 값을 사용해도 무방합니다.

1000개가 넘는 채널이 있는 경우 50개의 현재 채널마다 하나의 디스패처를 허용하는 방법을 지침으로 사용하십시오. 예를 들어, 최대 2000개의 활성 채널을 처리하려면 CHIDISPS(40)을 지정하십시오.

TCP/IP를 사용하는 경우 CHIDISPS에 더 큰 값을 지정해도 TCP/IP 채널에 사용하는 최대 디스패처 수는 100입니다.

CHIADAPS

큐 관리자에 대한 각 IBM MQ API 호출은 다른 모든 것과 독립이며 모든 어댑터 TCB에서 작성될 수 있습니다. 지속적 메시지를 사용하는 호출은 로그 입출력으로 인해 비지속 메시지에 대한 호출보다 훨씬 오래 걸릴 수 있습니다. 따라서 다수의 채널에 걸쳐 많은 수의 지속적 메시지를 처리하는 채널 시작기는 최적의 성능을 위해 기본 8개의 어댑터 TCB 보다 더 많은 것이 필요할 수 있습니다. 확보한 배치 크기가 작은 경우(배치 처리 끝에 로그 입출력(I/O)이 필요하기 때문)와 씬 클라이언트 채널을 사용하는 경우 특히 많이 필요합니다.

프로덕션 환경에 제안되는 값은 CHIADAPS(30)입니다. 이 값보다 많이 사용해도 그다지 큰 이점이 없으며 CHIADAPS(30)이 필요한 어댑터 TCB 수보다 많은 수인 경우 이 값을 사용해도 무방합니다.

MAXCHL

각 채널은 채널 시작 시 특정 디스패처 TCB와 연관되며 채널이 중지할 때까지 해당 TCB와 연관된 상태로 남아 있습니다. 여러 채널이 각 TCB를 공유할 수 있습니다. MAXCHL은 사용 가능한 여러 디스패처 TCB에 채널을 분산하는 데 사용합니다. 모든 디스패처 TCB가 사용 중일 때까지, 시작할 첫 번째(MIN((MAXCHL / CHIDISPS) , 10)) 채널은 첫 번째 디스패처 TCB와 연관되는 방식입니다.

그러면 채널의 수가 적고 MAXCHL이 큰 경우 디스패처에 채널이 고르게 분배되지 않습니다. 예를 들어 CHIDISPS(10)을 설정하고 MAXCHL은 기본값인 200으로 설정했지만 50개의 채널밖에 없는 경우 5개의 디스패처가 각각 10개의 채널과 연관되며 5개는 사용되지 않습니다. MAXCHL을 실제로 사용할 채널 수(고정된 작은 수)로 설정하도록 제안합니다.

이 큐 관리자 특성을 변경하는 경우 값이 호환 가능하도록 ACTCHL, LU62CHL 및 TCPCHL 큐 관리자 특성도 검토해야 합니다. 이 특성에 대한 전체 설명 및 해당 관계는 큐 관리자 매개변수를 참조하십시오.

채널 시작기에 대해 z/OS UNIX 시스템 서비스 환경 설정

채널 시작기(CHINIT)는 OMVS 스레드를 사용합니다. 새 CHINIT를 작성하거나 SSLTASKS 또는 디스패처의 수를 수정하기 전에 OMVS 구성 매개변수를 검토하십시오.

각 CHINIT는 3 + CHIDISP + SSLTASKS OMVS 스레드를 사용합니다. 이 값은 LPAR에서 사용하는 OMVS 스레드의 총 수에 합산되고 CHINIT 시작된 태스크 사용자 ID가 사용하는 스레드 수에 합산됩니다.

D OMVS,L을 사용하고 현재 사용, 하이워터 사용 및 MAXPROCSYS의 시스템 한계(시스템이 허용하는 최대 프로세스 수)를 검토할 수 있습니다.

새 CHINIT를 추가하거나 CHIDISPS 또는 SSLTASKS의 값을 늘리는 경우 스레드 수의 증가를 계산하고 MAXPROCSYS 값에 미치는 영향을 검토해야 합니다. **SETOMVS** 명령을 사용하여 동적으로 MAXPROCSYS를 변경하거나 BPXPRCxx parmlib 값을 업데이트 또는 둘 다를 수행할 수 있습니다.

OMVS 매개변수 MAXPROCUSER는 UID가 동일한 하나의 OMVS 사용자가 소유할 수 있는 OMVS 스레드 수입니다. 스레드는 이 값을 향해 계수합니다. 그러므로 시작된 태스크 사용자 ID가 동일한 2개의 CHINITS가 있으며 각각 10개의 디스패처와 3개의 SSLTASKS가 있는 경우 OMVS uid의 스레드는 2 *(3 +10 + 3) = 32개입니다.

D OMVS,O 명령을 실행하여 기본 MAXPROCUSER를 표시할 수 있으며 **SETOMVS** 명령을 사용하여 MAXPROCUSER를 동적으로 변경하거나 BPXPRCxx parmlib 값을 업데이트 또는 둘 다를 수행할 수 있습니다.

RACF 명령 **ALTUSER userid OMVS(PROCUSERMAX(nnnn))** 또는 동등한 값을 사용하여 사용자별로 이 값을 겹쳐쓸 수 있습니다.

채널 시작기를 시작하려면 다음 명령을 실행하십시오.

```
START CHINIT
```

채널 시작기가 성공적으로 시작했는지 확인하려면 xxxxCHIN(ssidCHIN) 작업 로그에 ICH408I 오류가 없는지 확인하십시오.

관련 개념

635 페이지의 『배치, TSO, RRS 어댑터 설정』
적절한 STEPLIB 연결에 라이브러리를 추가하여 애플리케이션에서 어댑터를 사용할 수 있도록 설정하십시오. 어댑터에서 발행된 SNMP 덤프를 제공하려면 CSQSNAP DDname을 할당하십시오. 애플리케이션 프로그램의 이식성을 향상시키려면 CSQBDEFV 사용을 고려하십시오.

관련 정보

채널 시작기 통계 데이터 레코드

z/OS 배치, *TSO, RRS* 어댑터 설정

적절한 STEPLIB 연결에 라이브러리를 추가하여 애플리케이션에서 어댑터를 사용할 수 있도록 설정하십시오. 어댑터에서 발행된 SNMP 덤프를 제공하려면 CSQSNAP DDname을 할당하십시오. 애플리케이션 프로그램의 이식성을 향상시키려면 CSQBDEFV 사용을 고려하십시오.

- 필요에 따라 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.

배치 연결을 사용하는 기타 애플리케이션 및 배치에서 어댑터를 사용할 수 있도록 설정하려면, 배치 애플리케이션에 대한 STEPLIB 연결에 다음 IBM MQ 라이브러리를 추가하십시오.

- thlqual.SCSQANL *x*
- thlqual.SCSQAUTH

여기서 *x*는 자국어(NL)의 언어 문자입니다. (라이브러리가 LPA 또는 링크 목록에 있는 경우 이를 수행하지 않아도 됩니다.)

TSO 애플리케이션의 경우 TSO 명령 TSOLIB를 사용하여 활성화하거나 TSO 로그온 프로시저에서 STEPLIB 연결에 라이브러리를 추가하십시오.

어댑터가 예상치 못한 IBM MQ 오류를 발견하는 경우, z/OS SNAP 덤프를 DDname CSQSNAP로 발행하고 이유 코드 MQRC_UNEXPECTED_ERROR를 애플리케이션에 발행합니다.CSQSNAP DD문이 애플리케이션 JCL에 없거나 CSQSNAP가 TSO 아래 데이터 세트에 할당되지 않은 경우 덤프는 생성되지 않습니다. 이 경우 애플리케이션

JCL에 CSQSNAP DD 문을 포함하거나 TSO 아래 데이터 세트에 CSQSNAP를 할당하여 애플리케이션을 다시 실행할 수 있습니다. 그러나 일부 문제점은 일시적이므로 문제가 발생한 경우 실패 이유를 캡처하기 위해 애플리케이션 JCL에 CSQSNAP 문을 포함하거나 TSO 로그온 프로시저에 있는 데이터 세트에 CSQSNAP를 할당할 것을 권장합니다.

제공된 프로그램 CSQBDEFV는 애플리케이션 프로그램의 이식성을 향상시킵니다. CSQBDEFV에서, 애플리케이션 프로그램의 `MQCONN` 또는 `MQCONNX` 호출에서 이를 지정하는 대신, 연결을 위해 큐 관리자 또는 큐 공유 그룹의 이름을 지정할 수 있습니다. 각 큐 관리자 또는 큐 공유 그룹에 대해 CSQBDEFV의 새 버전을 작성할 수 있습니다. 이를 수행하려면 다음 단계를 수행하십시오.

1. thlqual.SCSQASMS에서 사용자 라이브러리로 IBM MQ 어셈블러 프로그램 CSQBDEFV를 복사하십시오.
2. 제공되는 프로그램은 기본 서브시스템 이름 CSQ1을 포함합니다. 테스트 및 설치 확인을 위해 이 이름을 유지할 수 있습니다. 프로덕션 서브시스템의 경우 NAME=CSQ1을 1 - 4자의 서브시스템 이름으로 변경하거나 CSQ1을 사용할 수 있습니다.

   큐 공유 그룹을 사용 중인 경우, CSQ1 대신 큐 공유 그룹 이름을 지정할 수 있습니다. 이를 수행하려면 프로그램에서 해당 그룹 내에서 활성 큐 관리자로의 연결 요청을 실행합니다.
3. 프로그램을 어셈블 및 링크 편집하여 CSQBDEFV 로드 모듈을 생성하십시오. 어셈블리의 경우 SYSLIB 연결에 라이브러리 thlqual.SCSQMACS를 포함하십시오. 링크 편집 매개변수 `RENT,AMODE=31,RMODE=ANY`를 사용하십시오. 이는 thlqual.SCSQPROC(CSQ4DEFV)의 샘플 JCL에서 표시됩니다. 그런 다음 thlqual.SCSQAUTH 앞에서 z/OS 배치 또는 TSO STEPLIB에 로드 라이브러리를 포함하십시오.

관련 개념

636 페이지의 『조작 및 제어판 설정』
조작 및 제어판을 설정하려면 먼저 필요한 패널, EXEC, 메시지 및 테이블이 포함된 라이브러리를 설정해야 합니다. 이를 수행하려면 패널에 사용할 자국어(NL)를 고려해야 합니다. 이를 수행한 경우, IBM MQ 운영에 대한 기본 ISPF 메뉴 및 제어판을 선택에 따라 업데이트하고 기능 키 설정을 변경할 수 있습니다.

z/OS 조작 및 제어판 설정

조작 및 제어판을 설정하려면 먼저 필요한 패널, EXEC, 메시지 및 테이블이 포함된 라이브러리를 설정해야 합니다. 이를 수행하려면 패널에 사용할 자국어(NL)를 고려해야 합니다. 이를 수행한 경우, IBM MQ 운영에 대한 기본 ISPF 메뉴 및 제어판을 선택에 따라 업데이트하고 기능 키 설정을 변경할 수 있습니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 한 번씩 이 태스크를 수행해야 합니다.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.

라이브러리 설정

IBM MQ 운영 및 제어판을 설정하려면 다음 단계를 수행하십시오.

1. 연결에 포함된 모든 라이브러리가 동일한 형식(F, FB, V, VB)으로 되어 있고 동일한 블록 크기를 갖고 있는지 블록 크기를 줄일 목적인지 확인하십시오. 그렇지 않으면 이러한 패널을 사용하려고 하는 중에 문제가 생길 수 있습니다.
2. SYSEXEC 또는 SYSPROC 연결에 라이브러리 thlqual.SCSQEXEC을 포함하거나 TSO ALTLIB 명령을 사용하여 이를 활성화하십시오. 설치 동안 고정 블록 80 레코드 형식으로 할당된 이 라이브러리에는 필수 EXEC가 포함됩니다.

   라이브러리를 SYSEXEC 연결에 넣는 것이 좋습니다. 그러나 이를 SYSPROC에 넣으려면 라이브러리가 80바이트의 레코드 길이를 가져야 합니다.
3. 링크 목록 또는 LPA에 없는 경우 thlqual.SCSQAUTH 및 thlqual.SCSQANLx를 TSO 로그온 프로시저 STEPLIB에 추가하거나 TSO TSOLIB 명령을 사용하여 이를 활성화하십시오.
4. IBM MQ 패널 라이브러리를 영구적으로 ISPF 라이브러리 설정에 추가하거나 패널이 사용될 때 동적으로 설정되도록 허용할 수 있습니다. 전자를 선택할 경우 다음을 수행해야 합니다.

   a. 조작 및 제어판 정의가 들어 있는 라이브러리를 ISPPLIB 연결에 포함하십시오. 이름은 thlqual.SCSQPNLx입니다. 여기서 x는 자국어(NL)의 언어 문자입니다.

b. 필수 테이블이 들어 있는 라이브러리를 ISPTLIB 연결에 포함하십시오. 이름은 thlqual.SCSQTBLx입니다. 여기서 x는 자국어(NL)의 언어 문자입니다.

c. 필수 메시지가 들어 있는 라이브러리를 ISPMLIB 연결에 포함하십시오. 이름은 thlqual.SCSQMSGx입니다. 여기서 x는 자국어(NL)의 언어 문자입니다.

d. 필수 로드 모듈이 들어 있는 라이브러리를 ISPLLIB 연결에 포함하십시오. 이 라이브러리의 이름은 thlqual.SCSQAUTH입니다.

5. TSO 명령 프로세서 패널에서 IBM MQ 패널에 액세스할 수 있는지 테스트하십시오. 이는 보통 ISPF/PDF 기본 옵션 메뉴의 옵션 6입니다. 실행하는 EXEC의 이름은 CSQOREXX입니다.IBM MQ 라이브러리를 4단계에서와 같이 ISPF 설정에 영구적으로 넣은 경우 지정할 매개변수가 없습니다. 그렇지 않은 경우 다음을 사용하십시오.

```
CSQOREXX thlqual langletter
```

여기서 `langletter`는 사용할 자국어(NL)를 식별하는 문자입니다.

C
: 중국어

E
: 미국령 영어(대소문자 혼용)

F
: 프랑스어

K
: 일본어

U
: 미국령 영어(대문자)

ISPF 메뉴 업데이트

ISPF에서 IBM MQ 운영 및 제어판에 액세스할 수 있도록 ISPF 기본 메뉴를 업데이트할 수 있습니다. &ZSEL에 대한 필수 설정은 다음과 같습니다.

```
CMD(%CSQOREXX thlqual langletter)
```

`thlqual` 및 `langletter`에 관한 정보는 637 페이지의 『5』 단계를 참조하십시오.

자세한 내용은 *z/OS: ISPF Dialog Developer's Guide and Reference* 매뉴얼을 참조하십시오.

기능 키 및 명령 설정 업데이트

정상 ISPF 프로시저를 사용하여 패널에서 사용된 기능 키와 명령 설정을 변경할 수 있습니다. 애플리케이션 ID는 CSQO입니다.

하지만 도움말 정보는 사용자가 작성한 변경사항을 반영하도록 업데이트되지 않으므로 권장되지 않습니다.

관련 개념

637 페이지의 『IBM MQ 덤프 형식화 멤버 포함』
IPCS(Interactive Problem Control System)를 사용하여 IBM MQ 덤프를 형식화할 수 있으려면 일부 시스템 라이브러리를 업데이트해야 합니다.

z/OS *IBM MQ* 덤프 형식화 멤버 포함

IPCS(Interactive Problem Control System)를 사용하여 IBM MQ 덤프를 형식화할 수 있으려면 일부 시스템 라이브러리를 업데이트해야 합니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 한 번씩 이 태스크를 수행해야 합니다.

- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 합니다.

IPCS(Interactive Problem Control System)를 사용하여 IBM MQ 덤프를 형식화할 수 있으려면 데이터 세트 thlqual.SCSQPROC(CSQ7IPCS)를 SYS1.PARMLIB로 복사하십시오. 이 데이터 세트를 편집하면 안 됩니다.

IPCS에 대한 TSO 프로시저를 사용자 정의한 경우, thlqual.SCSQPROC(CSQ7IPCS)가 IPCSPARM 정의의 라이브러리에 복사될 수 있습니다. IPCSPARM에 대한 세부사항은 *z/OS MVS IPCS* 사용자 정의 매뉴얼을 참조하십시오.

또한 ISPPLIB 연결 시에 라이브러리 thlqual.SCSQPNLA를 포함해야 합니다.

덤프 형식화 프로그램을 TSO 세션 또는 IPCS 작업에 사용 가능하도록 만들려면 또한 STEPLIB 연결 시에 라이브러리 thlqual.SCSQAUTH를 포함하거나 TSO TSOLIB 명령을 사용하여(링크 목록 또는 LPA에 이미 있더라도) 이를 활성화해야 합니다.

관련 개념

638 페이지의 『정보 메시지 억제』
IBM MQ 시스템이 많은 수의 정보 메시지를 생성할 수 있습니다. 메시지가 콘솔이나 하드카피 로그로 전송되지 못하게 할 수 있습니다.

z/OS 정보 메시지 억제

IBM MQ 시스템이 많은 수의 정보 메시지를 생성할 수 있습니다. 메시지가 콘솔이나 하드카피 로그로 전송되지 못하게 할 수 있습니다.

- *IBM MQ*를 실행하려는 각 *z/OS* 시스템에 대해 한 번씩 이 태스크를 수행해야 합니다.
- 이전 버전에서 마이그레이션할 때는 이 태스크를 수행할 필요가 없습니다.

IBM MQ 시스템이 과도하게 사용된 경우, 많은 채널이 중지 및 시작되고 많은 정보 메시지가 z/OS 콘솔 및 하드카피 로그로 전송됩니다. IBM MQ - IMS 브릿지 및 버퍼 관리자도 많은 수의 정보 메시지를 생성할 수 있습니다.

필요한 경우 SYS1.PARMLIB의 MPFLSTxx 멤버에 의해 지정되는 z/OS 메시지 처리 기능 목록을 사용하여 이들 콘솔 메시지의 일부를 억제할 수 있습니다. 사용자가 지정하는 메시지가 여전히 하드카피 로그에 나타나지만 콘솔에는 나타나지 않습니다.

샘플 thlqual.SCSQPROC(CSQ4MPFL)이 MPFLSTxx에 대해 제안되는 설정을 표시합니다. MPFLSTxx에 대한 자세한 정보는 *z/OS MVS* 초기화 및 튜닝 참조 매뉴얼을 참조하십시오.

하드카피 로그에서 선택된 정보 메시지를 억제하려는 경우 z/OS 설치 엑시트 IEAVMXIT를 사용할 수 있습니다. 필수 메시지에 대해 다음 비트 스위치를 ON으로 설정할 수 있습니다.

CTXTRDTM
: 메시지를 삭제합니다.

  메시지가 콘솔에 표시되거나 하드카피에 로깅되지 않습니다.

CTXTESJL
: 작업 로그에서 억제합니다.

  메시지가 JES 작업 로그로 이동하지 않습니다.

CTXTNWTP
: WTP 처리를 수행하지 않습니다.

  메시지가 TSO 터미널이나 배치 작업의 시스템 메시지 데이터 세트로 전송되지 않습니다.

참고:

1. 기타 매개변수에 대한 자세한 정보는 MVS 설치 종료 문서를 참조하십시오.
2. 제안된 억제 목록 CSQ4MPFL의 메시지를 제외한 다른 메시지를 억제하는 것은 권장되지 않습니다.

또한 다음의 추가 매개변수를 지정할 수 있습니다.

EXCLMSG
: 로그에서 제외할 메시지 목록을 지정합니다.

이 목록의 메시지는 z/OS 콘솔 및 하드카피 로그로 송신되지 않습니다. 추가 정보는 615 페이지의 『CSQ6SYSP 사용』의 EXCLMSG를 참조하십시오.

관련 태스크

649 페이지의 『z/OS에서 큐 관리자 테스트』
큐 관리자를 사용자 정의하거나 마이그레이션한 경우 IBM MQ for z/OS에서 제공하는 일부 샘플 애플리케이션 및 설치 확인 프로그램을 실행하여 이를 테스트할 수 있습니다.

z/OS 큐 공유 그룹 구성

고가용성을 위해 공유 큐를 사용하려면 이러한 주제를 큐 공유 그룹 구성을 위한 단계별 안내서로 사용합니다.

IBM MQ for z/OS 시스템을 설정하는 프로세스의 해당 부분에서 단계를 완료할 때 큐 공유 그룹 데이터를 추가하려면 613 페이지의 『시스템 매개변수 모듈 조정』해야 합니다. CSQ6SYSP를 수정하여 QSGDATA 매개변수를 지정해야 합니다.

z/OS *Db2* 환경 설정

큐 공유 그룹을 사용하는 경우 여러 샘플 작업을 사용자 정의 및 실행하여 필요한 Db2 오브젝트를 작성해야 합니다.

Db2 환경 설정

여러 샘플 작업을 사용자 정의 및 실행하여 필요한 Db2 오브젝트를 작성 및 바인드해야 합니다.

- 각 Db2 데이터 공유 그룹에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 경우 `bind` 및 `grant` 단계를 수행해야 합니다.
- 큐 공유 그룹을 사용하지 않는 경우, 이 태스크를 생략하십시오.

  나중에 큐 공유 그룹을 사용하려는 경우, 해당 시점에서 이 태스크를 수행하십시오.

V 9.0.4 IBM MQ에서는 두 개의 동등한 작업 세트를 제공합니다. CSQ45 접두부가 있는 작업은 IBM MQ의 이전 버전과의 호환 및 Db2 버전 11 및 이전 버전에서 사용하기 위한 용도입니다. Db2 V12 이상에서 새 데이터 공유 그룹을 설정하는 경우, CSQ4X 접두부가 있는 작업을 사용하는 것이 좋습니다. 왜냐하면 이 작업은 동적 크기 조정 및 범용 테이블스페이스에 최근 Db2 기능을 사용하기 때문입니다.

IBM MQ가 큐 공유 그룹에 사용되는 Db2 계획을 액세스 및 실행할 수 있는 환경을 설정해야 합니다.

각 새로운 Db2 데이터 공유 그룹에 대해 다음 단계가 수행되어야 합니다. 모든 샘플 JCL은 thlqual.SCSQPROC에 있습니다.

1. 샘플 JCL CSQ45CSG V 9.0.4 (또는 CSQ4XCSG)를 사용자 정의하고 실행하여 IBM MQ 데이터베이스, 테이블스페이스 및 테이블에 사용할 스토리지 그룹을 작성하십시오.
2. 샘플 JCL CSQ45CDB V 9.0.4 (또는 CSQ4XCDB)를 사용자 정의하고 실행하여 이 Db2 데이터 공유 그룹에 연결되는 모든 큐 관리자에서 사용할 데이터베이스를 작성하십시오.
3. 샘플 JCL CSQ45CTS V 9.0.4 (또는 CSQ4XCTS)를 사용자 정의하고 실행하여 큐 공유 그룹(1단계에서 작성됨)에 사용되는 큐 관리자 및 채널 시작기 테이블을 포함하는 테이블스페이스를 작성하십시오.
4. 샘플 JCL CSQ45CTB V 9.0.4 (또는 CSQ4XCTB)를 사용자 정의하고 실행하여 12개의 Db2 테이블 및 연관된 색인을 작성하십시오. 어떤 행 이름이나 속성도 변경하지 마십시오.
5. 샘플 JCL CSQ45BPL을 사용자 정의하고 실행하여 큐 관리자, 유틸리티 및 채널 시작기에 대한 Db2 계획을 바인드하십시오.
6. 샘플 JCL CSQ45GEX를 사용자 정의하고 실행하여 큐 관리자, 유틸리티 및 채널 시작기가 사용하는 사용자 ID에 대한 계획에 실행 권한을 부여하십시오. 큐 관리자 및 채널 시작기에 대한 사용자 ID는 시작된 태스크 프로시저가 실행하는 사용자 ID입니다. 유틸리티에 대한 사용자 ID는 배치 작업이 제출될 수 있는 사용자 ID입니다.

   적절한 계획 이름은 다음 표에 표시되어 있습니다.

   - LTS LTS 열의 Long Term Support 버전

- CD CD 컬럼의 Continuous Delivery 버전입니다. 여기서 n 는 CD 릴리스를 나타냅니다. 각 릴리스에서 n은 1씩 증가합니다. 예를 들어 IBM MQ 9.0.3에서 CSQ5A90n은 CSQ5A903입니다.

사용자	계획(LTS)	계획(CD)
큐 관리자	CSQ5A 900, CSQ5C 900, CSQ5D 900, CSQ5K 900, CSQ5L 900, CSQ5M 900, CSQ5P 900, CSQ5R 900, CSQ5S 900, CSQ5T 900, CSQ5U 900, CSQ5W 900	CSQ5A 90n, CSQ5C 90n, CSQ5D 90n, CSQ5K 90n, CSQ5L 90n, CSQ5M 90n, CSQ5P 90n, CSQ5R 90n, CSQ5S 90n, CSQ5T 90n, CSQ5U 90n, CSQ5W 90n
CSQUTIL 배치 유틸리티의 SDEFS 기능	CSQ52 900	CSQ52 90n
CSQ5PQSG 및 CSQJUCNV 배치 유틸리티	CSQ5B 900	CSQ5B 90n
CSQUZAP 서비스 유틸리티	CSQ5Z 900	CSQ5Z 90n

Db2 설정 중의 장애 이벤트에서는 다음 작업을 사용자 정의하고 실행할 수 있습니다.

- CSQ45DTB - 테이블 및 색인을 삭제합니다.
- CSQ45DTS V 9.0.4 (또는 CSQ4XDTS) - 테이블스페이스를 삭제합니다.
- CSQ45DDB V 9.0.4 (또는 CSQ4XDDB) - 데이터베이스를 삭제합니다.
- CSQ45DSG V 9.0.4 (또는 CSQ4XDSG) - 스토리지 그룹을 삭제합니다.

참고: 이들 작업이 Db2 잠금 문제점 때문에 실패하는 경우, 특히 시스템이 과도하게 사용되고 있는 경우 Db2 자원에 대한 경합 때문일 수 있습니다. 나중에 작업을 다시 제출하십시오. 사용률이 낮거나 일시정지되는 경우에 이러한 작업을 실행하는 것이 선호됩니다.

Db2설정에 대한 자세한 정보는 *Db2 for z/OS 11.0.0* 의 Db2 관리 를 참조하십시오.

V 9.0.4 Db2설정에 대한 자세한 정보는 *Db2 for z/OS 12.0.0* 의 Db2 관리 를 참조하십시오.

Db2 테이블 크기에 관한 정보는 z/OS 에 대한 계획의 내용을 참조하십시오.

관련 개념

640 페이지의 『커플링 기능 설정』
큐 공유 그룹을 사용하는 경우 IXCMIAPU를 사용하여 커플링 기능 자원 관리(CFRM) 정책 데이트 세트의 큐 공유 그룹(QSG)에 있는 큐 관리자에서 사용하는 커플링 기능 구조를 정의하십시오.

z/OS 커플링 기능 설정

큐 공유 그룹을 사용하는 경우 IXCMIAPU를 사용하여 커플링 기능 자원 관리(CFRM) 정책 데이트 세트의 큐 공유 그룹(QSG)에 있는 큐 관리자에서 사용하는 커플링 기능 구조를 정의하십시오.

- 각 큐 공유 그룹에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.
- 큐 공유 그룹을 사용하지 않는 경우 이 태스크를 생략하십시오.

  나중에 큐 공유 그룹을 사용하려는 경우 해당 시점에서 이 태스크를 수행하십시오.

큐 공유 그룹에 대한 모든 구조는 큐 공유 그룹 이름으로 시작합니다. 다음 구조를 정의하십시오.

- 관리 구조 *qsg-name* CSQ_ADMIN. 이 구조는 IBM MQ에서 사용되며 사용자 데이터를 포함하지 않습니다.
- 시스템 애플리케이션 구조 *qsg-name* CSQSYSAPPL. 이 구조는 상태 정보를 저장하기 위해 IBM MQ 시스템 큐에서 사용됩니다.
- 공유 큐의 메시지를 보유하는 데 사용되는 하나 이상의 구조. 이들은 최대 16자의 길이로 선택 가능한 이름을 보유할 수 있습니다.
  - 처음 4자는 큐 공유 그룹 이름이어야 합니다. (큐 공유 그룹 이름이 4자 미만인 경우 네 자에 @ 기호를 채워야 합니다.)
  - 5번째 문자는 영문자여야 하며 후속 문자는 영문자 또는 숫자일 수 있습니다. 이름의 이 파트(큐 공유 그룹 이름 제외)는 CF 구조 오브젝트 또는 공유 큐를 정의할 때 CFSTRUCT 이름에 대해 지정한 항목입니다.

  공유 큐의 메시지를 보유하는 데 사용되는 구조의 이름에서는 영문자 및 숫자만 사용할 수 있으며, 다른 문자(예: 관리 구조의 이름에 사용되는 _ 문자)는 사용할 수 없습니다.

IXCMIAPU의 샘플 제어 명령문은 데이터 세트 thlqual.SCSQPROC(CSQ4CFRM)에 있습니다. 이들을 사용자 정의하고 커플링 기능에 대한 IXCMIAPU 작업에 추가한 후 이를 실행하십시오.

구조를 성공적으로 정의한 경우 사용되는 CFRM 정책을 활성화하십시오. 이를 수행하려면 다음 z/OS 명령을 실행하십시오.

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME= policy-name
```

CF 구조 및 해당 크기 계획에 대한 정보는 커플링 기능 자원 정의를 참조하십시오.

관련 개념

608 페이지의 『ESM 보안 제어 구현』
큐 관리자 및 채널 시작기에 대한 보안 제어를 구현합니다.

z/OS *SMDS* 환경 설정

SMDS를 사용하여 공유 큐에 있는 메시지를 오프로드하려면 SMDS 오프로드 스토리지 환경을 설정하십시오.

- *SMDS*로 데이터를 오프로드하도록 구성하려는 큐 공유 그룹의 각 큐 관리자 및 구조에 대해 이 태스크를 수행하십시오.
- 나중에 *SMDS*에 데이터를 오프로드하도록 추가 구조를 구성하려는 경우 이 태스크는 그 때 다시 수행할 수 있습니다.
- 큐 공유 그룹을 사용하지 않는 경우 이 태스크를 생략하십시오.

  나중에 큐 공유 그룹을 사용하려는 경우 해당 시점에서 이 태스크를 수행하십시오.

SMDS 환경 설정

1. 구조와 데이터 세트 공간 요구사항을 추정하십시오. 공유 메시지 데이터 세트 용량 고려사항을 참조하십시오.
2. 데이터 세트를 할당 및 사전 포맷하십시오. 공유 메시지 데이터 세트 작성을 참조하십시오.
3. IBM MQ에 CF 구조를 정의할 때, CFLEVEL(5) 및 OFFLOAD(SMDS)로 CFSTRUCT를 정의했는지 확인하십시오.

관련 개념

640 페이지의 『커플링 기능 설정』
큐 공유 그룹을 사용하는 경우 IXCMIAPU를 사용하여 커플링 기능 자원 관리(CFRM) 정책 데이트 세트의 큐 공유 그룹(QSG)에 있는 큐 관리자에서 사용하는 커플링 기능 구조를 정의하십시오.

z/OS *Db2* 테이블에 *IBM MQ* 입력 항목 추가

큐 공유 그룹을 사용하는 경우, CSQ5PQSG 유틸리티를 실행하여 큐 공유 그룹 및 큐 관리자 항목을 Db2 데이터 공유 그룹의 IBM MQ 테이블에 추가하십시오.

- 각 *IBM MQ* 큐 공유 그룹 및 각 큐 관리자에 대해 이 태스크를 반복하십시오.

- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.
- 큐 공유 그룹을 사용하지 않는 경우 이 태스크를 생략하십시오.

  나중에 큐 공유 그룹을 사용하려면 그 때 이 태스크를 수행하십시오.

각 큐 공유 그룹과 큐 공유 그룹의 멤버가 될 수 있는 각 큐 관리자에 대해 CSQ5PQSG를 실행하십시오. (CSQ5PQSG는 IBM MQ for z/OS 관리에 설명되어 있습니다.)

다음 조치를 지정된 순서로 수행하십시오.

1. CSQ5PQSG 프로그램의 ADD QSG 기능을 사용하여 IBM MQ Db2 테이블에 큐 공유 그룹 항목을 추가하십시오. thlqual.SCSQPROC(CSQ45AQS)에 샘플이 제공됩니다.

   Db2 데이터 공유 그룹에서 정의되는 각 큐 공유 그룹에 대해 이 기능을 한 번 수행하십시오. 큐 공유 그룹 항목이 큐 공유 그룹을 참조하는 큐 관리자 항목을 추가하기 전에 존재해야 합니다.
2. CSQ5PQSG 프로그램의 ADD QMGR 기능을 사용하여 큐 관리자 항목을 IBM MQ Db2 테이블에 추가하십시오. thlqual.SCSQPROC(CSQ45AQM)에 샘플이 제공됩니다.

   큐 공유 그룹의 멤버가 될 수 있는 각 큐 관리자에 대해 이 기능을 수행하십시오.

   참고:

   a. 큐 관리자는 하나의 큐 공유 그룹의 멤버만 될 수 있습니다.
   b. 큐 공유 그룹을 사용하기 위해서는 사용 가능한 RRS가 있어야 합니다.

관련 개념

613 페이지의 『시스템 매개변수 모듈 조정』
IBM MQ 시스템 매개변수 모듈은 IBM MQ가 그의 조작에서 사용하는 로깅, 아카이빙, 추적 및 연결 환경을 제어합니다. 기본 모듈이 제공됩니다. 일부 매개변수(예: 데이터 세트 이름)가 일반적으로 사이트에 한정되므로 고유 시스템 매개변수 모듈을 작성해야 합니다.

z/OS 큐 공유 그룹에 대한 *ESM* 보안 제어 구현

큐 공유 그룹에서 모든 큐 관리자에 대한 보안 제어를 구현하여 Db2 및 커플링 기능 목록 구조에 액세스합니다.

- 큐 공유 그룹에 있는 각 *IBM MQ* 큐 관리자에 대해 이 태스크를 반복하십시오.
- 이전 버전에서 마이그레이션할 때 이 태스크를 수행해야 할 수 있습니다.

큐 관리자, 채널 시작기 및 유틸리티와 연관된 사용자 ID에 연결할 각 Db2 서브시스템에 대한 RRSAF 연결을 설정하는 권한이 있는지 확인하십시오. 큐 관리자 및 채널 시작기에 대한 사용자 ID는 시작된 태스크 프로시저가 실행하는 사용자 ID입니다.

유틸리티에 대한 사용자 ID는 배치 작업이 제출될 수 있는 사용자 ID입니다. 사용자 ID가 READ 액세스 권한을 필요로 하는 RACF 프로파일은 DSNR 자원 클래스의 `Db2ssid.RRSAF`입니다.

큐 공유 그룹의 각 큐 관리자와 연관된 사용자 ID에 커플링 기능 목록 구조에 대한 적절한 레벨의 액세스 권한을 부여해야 합니다. RACF 클래스는 FACILITY입니다.

다음 사용자 ID는 ALTER 액세스가 필요합니다.

- `IXLSTR.structure-name` 프로파일에 대한 큐 관리자 ID
- CSQ5PQSG를 실행하는 사용자 ID

관련 개념

608 페이지의 『ESM 보안 제어 구현』
큐 관리자 및 채널 시작기에 대한 보안 제어를 구현합니다.

z/OS z/OS용 Advanced Message Security 구성

이러한 주제를 Advanced Message Security 구성을 위한 단계별 안내서로 사용합니다.

z/OS *Advanced Message Security*용 프로시저 작성

AMS(Advanced Message Security)를 사용하도록 구성될 각 IBM MQ 서브시스템은 AMS 주소 공간을 시작하기 위한 카탈로그된 프로시저가 필요합니다. 사용자 자신의 프로시저를 작성하거나 IBM이 제공하는 프로시저 라이브러리를 사용할 수 있습니다.

AMS(Advanced Message Security)를 사용하도록 구성될 각 IBM MQ 서브시스템의 경우 샘플 프로시저 CSQ4AMSM의 사본을 조정하십시오. 이를 수행하려면 다음 단계를 수행하십시오.

1. 샘플 시작된 태스크 프로시저 *thlqual*.SCSQPROC(CSQ4AMSM)을 SYS1.PROCLIB 또는 프로시저 라이브러리(SYS1.PROCLIB을 사용하지 않는 경우)로 복사하십시오. 프로시저의 이름을 xxxxAMSM으로 지정하십시오. 여기서 xxxx는 IBM MQ 서브시스템의 이름입니다. 예를 들어 CSQ1AMSM은 큐 관리자 CSQ1에 대한 AMS 시작된 태스크 프로시저입니다.
2. 사용하려는 각 IBM MQ 서브시스템에 대한 사본을 작성하십시오.
3. 샘플 프로시저 CSQ4AMSM의 지시사항을 사용하여 요구사항에 맞게 프로시저를 조정하십시오. 프로시저가 시작될 때 수정할 수 있도록 허용하기 위해 JCL에서 기호 매개변수를 사용할 수도 있습니다.
4. Language Environment® _CEE_ENVFILE 파일을 사용하여 AMS 태스크에 전달된 매개변수를 검토하고 선택적으로 변경하십시오. thlqual.SCSQPROC(CSQ40ENV) 샘플이 지원되는 매개변수를 나열합니다.

참고: 이 태스크가 각 IBM MQ 큐 관리자에 대해 반복되어야 합니다.

z/OS 시작된 태스크 사용자 고급 메시지 보안 설정

Advanced Message Security 태스크는 UNIX 시스템 서비스 프로세스로 알려질 수 있는 사용자 ID가 필요합니다.

또한, 태스크가 그를 위해 작업하는 사용자는 UNIX UID(사용자 ID) 및 GID(그룹 ID)의 적절한 정의도 가져서 이들 사용자가 UNIX 시스템 서비스 사용자로 알려져야 합니다. UNIX 시스템 서비스 UID 및 GID 정의에 대한 자세한 정보는 *z/OS: Security Server RACF Security Administrator's Guide*를 참조하십시오.

*z/OS: UNIX System Services Planning*은 전통적인 UNIX 보안을 z/OS 보안과 비교합니다. 두 전통적인 UNIX 보안 및 z/OS 보안 사이의 차이점은 커널 서비스가 적절한 권한의 두 레벨인 UNIX 레벨 및 z/OS UNIX 레벨을 지원한다는 점입니다.

사용자 설치의 보안 정책에 따라서는 Advanced Message Security 태스크가 superuser 권한(uid(0)) 또는 RACF FACILITY 클래스 BPX.DAEMON 및 BPX.SERVER 프로파일에 허용되는 그의 RACF ID 중 하나로 실행할 수 있습니다. 이 태스크는 해당 사용자의 RACF ID를 가정할 수 있어야 하기 때문입니다.

후자의 방법이 사용되거나 이미 BPX.DAEMON 또는 BPX.SERVER 프로파일을 활성화한 경우, Advanced Message Security 태스크 프로그램(thlqual.SCSQAUTH(CSQ0DSRV))이 RACF 프로그램 제어 라이브러리에 위치해야 합니다.

*z/OS: UNIX System Services Planning*을 검토하여 전통적인 UNIX 보안과 z/OS UNIX 보안 사이의 보안 차이점을 이해하십시오. 그러면 권한이 있는 USS(UNIX System Services) 프로세스 배치 및 실행에 대한 설치의 보안 정책에 따라 Advanced Message Security 태스크를 관리할 수 있습니다.

참조용으로 이 검토에 유용한 발행물은 다음과 같습니다.

- *z/OS: UNIX* 시스템 서비스 계획.
- *z/OS:* 보안 서버 *RACF* 보안 관리자 안내서.

참고: Advanced Message Security 수신인 인증서가 사용자 ID와 연관된 키 링으로 로드되기 때문에 이 태스크에 대한 사용자 ID를 주의해서 선택하십시오. 이 고려사항에 대해서는 z/OS에서 인증서 사용에서 설명합니다.

여기에 표시된 단계에서는 Advanced Message Security 시작된 태스크 사용자 설정 방법에 대해 설명합니다. 단계는 RACF 명령을 예로 사용합니다. 다른 보안 관리자를 사용하는 경우 해당 명령을 사용해야 합니다.

참고: 이 절의 예는 사용자가 RACF STARTED, FACILITY 및 SURROGAT 클래스를 위한 일반 프로파일 명령 처리와 일반 프로파일 검사를 활성화했다고 가정합니다. RACF가 일반 프로파일을 처리하는 방법에 대한 자세한 정보는 *z/OS: Security Server RACF Command Language Reference*를 참조하십시오.

1. 먼저 Advanced Message Security 시작 태스크 사용자에 대한 RACF 사용자 프로파일을 정의하십시오. 이들은 동일한 사용자일 수 있습니다.

   ```
   ADDUSER WMQAMSM NAME(' Advanced Message Security user') OMVS (UID(0)) DFLTGRP(group)
   ```

   설치 표준에 적합한 기본 '그룹'을 선택하십시오.

   참고: USS superuser 권한(UID(0))을 부여하길 원하지 않는 경우 Advanced Message Security 사용자 ID에 BPX.DAEMON 및 BPX.SERVER 기능 클래스 프로파일을 허용해야 합니다.

   ```
   PERMIT BPX.DAEMON CLASS(FACILITY) ID(WMQAMSM) ACCESS(READ)
   ```

   그리고 Advanced Message Security 태스크 프로그램(*thlqual*.SCSQAUTH(CSQ0DSRV))이 RACF 프로그램 제어 라이브러리에 위치해야 합니다.

   SCSQAUTH 라이브러리 프로그램을 제어하기 위해 다음 명령을 사용할 수 있습니다.

   ```
   RALTER PROGRAM * ADDMEM('thlqual.SCSQAUTH'//NOPADCHK) -or-
   RALTER PROGRAM ** ADDMEM('thlqual.SCSQAUTH'//NOPADCHK)
   SETROPTS WHEN(PROGRAM) REFRESH
   ```

   또한 Advanced Message Security 태스크에 의해 사용되는 자국어(NL) 라이브러리(*thlqual*.SCSQANLx)에 대한 프로그램 제어도 사용을 설정할 수 있습니다.

2. RACF STARTED 클래스가 활성인지 판별하십시오. 그렇지 않은 경우 RACF STARTED 클래스를 활성화하십시오.

   ```
   SETROPTS CLASSACT(STARTED)
   ```

3. 1단계에서 선택 또는 작성한 사용자 ID를 지정하여 Advanced Message Security 태스크에 대해 시작된 클래스 프로파일을 정의하십시오.

   ```
   RDEFINE STARTED qmgr AMSM.* STDATA(USER(WMQAMSM))
   ```

   여기서 *qmgr*은 시작된 태스크 이름의 접두부 이름입니다. 예를 들어, 시작된 태스크의 이름은 CSQ1AMSM으로 지정할 수 있습니다. 이 경우에는 `qmgr AMSM.*`를 `CSQ1AMSM.*`로 대체합니다.

   시작된 태스크 이름은 `qmgr AMSM.*`로 지정되어야 합니다.

4. SETROPTS RACF 명령을 사용하여 스토리지 내 RACLIST된 시작된 클래스 프로파일을 새로 고치십시오.

   ```
   SETROPTS RACLIST(STARTED) REFRESH
   ```

5. Advanced Message Security 태스크는 임시로 IBM MQ 메시지의 보호 처리 동안 클라이언트 요청자의 호스트 사용자 ID의 ID를 가정합니다. 따라서 요청하는 각 사용자 ID의 SURROGAT 클래스에 프로파일을 정의해야 합니다.

   이것은 RACF SURROGAT 클래스가 활성인 경우 단일 일반 프로파일로 수행될 수 있습니다. SURROGAT 클래스가 활성 상태가 아니면 검사는 무시됩니다. 필요한 SURROGAT 프로파일은 *z/OS: UNIX System Services Planning*에 설명되어 있습니다.

   SURROGAT 클래스에서 프로파일을 정의하려면 다음을 수행하십시오.

   a. RACF SETROPTS 명령을 사용하여 RACF SURROGAT 클래스를 활성화하십시오.

      ```
      SETROPTS CLASSACT(SURROGAT)
      ```

   b. RACF SURROGAT 클래스에 대한 일반 프로파일 처리 활성화:

      ```
      SETROPTS GENERIC(SURROGAT)
      ```

c. RACF SURROGAT 클래스에 대한 일반 프로파일 명령 처리를 활성화하십시오.

```
SETROPTS GENCMD(SURROGAT)
```

d. 대리 클래스 일반 프로파일을 정의하십시오.

```
RDEFINE SURROGAT BPX.SRV.* UACC(NONE)
```

e. Advanced Message Security 사용자 ID를 일반 SURROGAT 클래스 프로파일에 허용하십시오.

```
PERMIT BPX.SRV.* CLASS(SURROGAT) ID(WMQAMSM) ACCESS(UPDATE)
```

참고: *z/OS: UNIX System Services Planning*에 설명된 대로 특정 사용자가 Advanced Message Security 태스크에 의해 처리되도록 제한하려는 경우 더 구체적인 프로파일을 정의할 수 있습니다.

f. Advanced Message Security 사용자 ID를 BPX.SERVER 기능에 허용(인증서 및 키 링 작성에서 아직 수행하지 않은 경우):

```
PERMIT BPX.SERVER CLASS(FACILITY) ID(WMQAMSM) ACCESS(READ)
```

6. Advanced Message Security 태스크는 z/OS 시스템 SSL 서비스가 제공하는 기능을 사용하여 SAF 관리 키 링을 엽니다. 키 링의 컨텐츠에 액세스하는 기본 SAF(System Authorization Facility)는 RACF 또는 동등한 보안 관리자에 의해 제어됩니다.

   이 서비스는 IRRSDL00(R_datalib) 호출 가능 서비스입니다. 이 호출 가능 서비스는 RACF FACILITY 클래스에 정의되는 RACF RACDCERT 명령을 보호하는 데 사용되는 것과 동일한 프로파일로 보호됩니다. 따라서 Advanced Message Security 사용자 ID가 다음 명령을 사용하여 프로파일에 허용되어야 합니다.

   a. 아직 수행하지 않은 경우, RACDCERT 명령 및 IRRSDL00 호출 가능 서비스를 보호하는 RACF FACILITY 클래스에 RACF 일반 프로파일을 정의하십시오.

   ```
   RDEFINE FACILITY IRR.DIGTCERT.* UACC(NONE)
   SETROPTS RACLIST(FACILITY) REFRESH
   ```

   b. 시작된 태스크 사용자 ID에 대한 권한을 RACF 일반 프로파일에 부여하십시오.

   ```
   PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(WMQAMSM) ACC(READ)
   ```

   또는 다음과 같이 RDATALIB 클래스에 있는 데이터 서비스 태스크 사용자의 키 링에 대한 읽기 액세스 권한을 부여할 수 있습니다.

   ```
   PERMIT WMQASMD.DRQ.AMS.KEYRING.LST CLASS(RDATALIB) ID(WMQAMSM) ACC(READ)
   ```

AMS에 대한 자원 보안

시작된 태스크 사용자는 SYSTEM.PROTECTION.POLICY.QUEUE에 대한 읽기 권한이 필요합니다.

시작 태스크 사용자는 배치 애플리케이션으로 큐 관리자에 연결할 권한이 필요합니다. 추가 정보는 배치 연결에 대한 연결 보안 프로파일을 참조하십시오.

z/OS *Advanced Message Security*의 보안 관리자에게 *RACDCERT* 권한 부여

Advanced Message Security 보안 관리자는 디지털 인증서를 작성 및 관리하기 위해 RACDCERT 명령을 사용할 권한이 필요합니다.

이 역할에 적절한 사용자 ID를 식별하고 RACDCERT 명령을 사용하도록 권한을 부여하십시오. 예를 들면, 다음과 같습니다.

```
PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID(admin) ACCESS(CONTROL)
SETROPTS RACLIST(FACILITY) REFRESH
```

여기서 `admin` 는 Advanced Message Security 보안 관리자의 사용자 ID입니다.

z/OS 사용자에게 *Advanced Message Security*의 자원 권한 부여

Advanced Message Security 사용자는 관련 자원 권한이 필요합니다.

Advanced Message Security 보호 메시지를 넣거나 가져오는 사용자인 Advanced Message Security 사용자는 다음이 필요합니다.

- 사용자 ID와 연관된 OMVS 세그먼트
- IRR.DIGTCERT.LISTRING 또는 RDATALIB의 권한
- ICSF 클래스 CSFSERV 및 CSFKEYS 프로파일의 권한

Advanced Message Security 태스크는 일시적으로 해당 클라이언트의 ID를 가정합니다. 즉, 이 태스크는 Advanced Message Security에 의해 보호되는 큐에 대한 IBM MQ 메시지 처리 중에 Advanced Message Security 사용자의 z/OS 사용자 ID의 대리 역할을 수행합니다.

태스크가 사용자의 z/OS ID를 가정하기 위해서, 클라이언트 z/OS 사용자 ID가 그의 사용자 프로파일과 연관된 정의된 OMVS 세그먼트를 가져야 합니다.

관리 도움으로서, RACF는 RACF 사용자 및 그룹 프로파일과 연관될 수 있는 기본 OMVS 세그먼트를 정의하는 기능을 제공합니다. z/OS 사용자 ID 또는 그룹 프로파일이 명시적으로 정의된 OMVS 세그먼트를 갖지 않는 경우 이 기본값이 사용됩니다. 다수의 사용자가 Advanced Message Security를 사용할 계획인 경우 사용자마다 OMVS 세그먼트를 명시적으로 정의하지 않고 이 기본값을 사용하도록 선택할 수 있습니다.

*z/OS: Security Server RACF Security Administrator's Guide*에 기본 OMVS 세그먼트 정의에 대한 상세한 프로시저가 들어 있습니다. 이 서적에서 요약되는 프로시저를 검토하여 RACF 사용자 및 그룹 프로파일의 기본 OMVS 세그먼트 정의가 사용자 설치에 적합한지 판별하십시오.

모든 Advanced Message Security 사용자에게 IRR.DIGTCERT.LISTRING 클래스 기능에 대한 읽기 권한을 부여하려면 다음 명령을 실행하십시오.

```
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(READ)
```

또는 다음 명령을 실행하여 사용자별로 읽기 권한을 부여하십시오.

```
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(userid) ACCESS(READ)
```

여기서 사용자 ID는 Advanced Message Security 사용자의 이름입니다.

또는 RDATALIB 클래스를 사용하여 특정 키 링에 액세스 권한을 부여할 수 있습니다(RDATALIB 권한은 IRR.DIGTCERT.LISTRING 권한보다 우선함). 예를 들면, 다음과 같습니다.

```
PERMIT user.DRQ.AMS.KEYRING.LST CLASS(RDATALIB) ID(user) ACC(READ)
```

ICSF 관리 인증서 및 개인 키를 사용하는 경우 Advanced Message Security 사용자는 특정 클래스 CSFSERV 및 CSFKEYS 프로파일에 대한 액세스 권한이 있어야 합니다. 이 액세스는 다음 표에 자세히 설명되어 있습니다.

표 *48.* 클래스 *CSFSERV* 및 *CSFKEYS* 프로파일에 필요한 사용자 액세스 권한

클래스	프로파일	권한
CSFSERV	CSFDSG	READ
CSFSERV	CSFPKE	READ

표 48. 클래스 CSFSERV 및 CSFKEYS 프로파일에 필요한 사용자 액세스 권한 (계속)		
클래스	**프로파일**	**권한**
CSFSERV	CSFPKD	READ
CSFSERV	CSFDSV	READ
CSFKEYS	ICSF PKDS Label	READ

V 9.0.1 z/OS mqweb 서버 구성

이러한 주제를 mqweb 서버 구성을 위한 단계별 안내서로 사용합니다.

관련 태스크

545 페이지의 『IBM MQ Console 및 REST API 구성』

IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에 기본 구성이 제공됩니다. 이 컴포넌트 중 하나를 사용하려면 사용자의 로그인을 허용하는 보안 구성 같은 여러 구성 태스크를 수행해야 합니다. 이 주제에서는 사용 가능한 모든 구성 옵션에 대해 설명합니다.

V 9.0.1 z/OS *Liberty* 서버 정의 작성

IBM MQ for z/OS Unix System Services 웹 컴포넌트를 설치했으며, MQ Console 또는 REST API를 사용하려는 경우 Liberty 서버 정의를 작성하고 사용자 정의해야 합니다.

시작하기 전에

Liberty 서버를 사용하려면 SYSTEM.REST.REPLY.QUEUE를 작성해야 합니다. 609 페이지의 『초기화 입력 데이터 세트 사용자 정의』의 최신 **CSQ4INSG** 샘플을 사용하여 이를 수행하십시오.

이 태스크 정보

- MQ Console 또는 REST API를 실행하려는 z/OS 시스템마다 한 번씩 이 태스크를 수행해야 합니다.
- 실행 중인 각 버전의 IBM MQ 에 대해 Liberty 서버가 필요합니다.
- 이전 버전에서 마이그레이션할 때 서버 구성을 새로 고치거나 수정해야 할 수 있습니다.

IBM MQ for z/OS Unix System Services웹 구성요소에는 mqweb라고 하는 단일 Liberty 서버를 작성해야 합니다.

서버 구성 및 로그 파일은 모두 Liberty 사용자 디렉토리에 저장됩니다.

다음 프로시저를 수행하여 mqweb 서버 정의를 작성하십시오.

프로시저

1. Liberty 사용자 디렉토리에 적합한 위치를 선택하십시오.

   mqweb 서버가 실행되는 사용자 ID에는 이 사용자 디렉토리 및 해당 컨텐츠에 대한 읽기 및 쓰기 액세스 권한이 필요합니다. 이 사용자 디렉토리에 로그 파일뿐만 아니라 서버 구성이 포함되어 있기 때문에 별도 파일 시스템에 이 디렉토리를 작성해야 합니다.
2. 사용자의 현재 디렉토리가 **crtmqweb.sh** 스크립트의 위치인 `PathPrefix/web/bin`인지 확인하십시오.

   `PathPrefix`는 IBM MQ UNIX System Services 컴포넌트 설치 경로입니다.
3. **crtmqweb.sh** 스크립트를 실행하여 템플리트 mqweb 서버 정의를 포함하는 Liberty 사용자 디렉토리를 작성하십시오.

   참고: **crtmqweb.sh** 스크립트는 Liberty 사용자 디렉토리의 이름인 하나의 선택적 매개변수만 승인합니다.

   Liberty 사용자 디렉토리 이름을 제공하지 않으면 `/var/mqm/web/installation1`의 기본값이 사용됩니다.
4. 다음 명령을 사용하여 Liberty 사용자 디렉토리의 디렉토리 및 파일의 소유권을 변경하여 mqweb 서버가 실행되는 사용자 ID및 그룹에 속하도록 하십시오.

```
chown -R userid:group path
```

경로에 그룹 쓰기 액세스 권한을 제공하려면 다음 명령을 실행하십시오.

```
chmod -R 770 path
```

다음에 수행할 작업

648 페이지의 『Liberty 서버에 대한 프로시저 작성』

관련 태스크

545 페이지의 『IBM MQ Console 및 REST API 구성』

IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에 기본 구성이 제공됩니다. 이 컴포넌트 중 하나를 사용하려면 사용자의 로그인을 허용하는 보안 구성 같은 여러 구성 태스크를 수행해야 합니다. 이 주제에서는 사용 가능한 모든 구성 옵션에 대해 설명합니다.

V 9.0.1 z/OS *Liberty* 서버에 대한 프로시저 작성

IBM MQ for z/OS Unix System Services 웹 컴포넌트를 설치했으며, MQ Console 또는 REST API를 사용하려는 경우 Liberty mqweb 서버를 시작하는 카탈로그화된 프로시저를 작성해야 합니다.

- IBM MQ를 실행하려는 각 z/OS 시스템에 대해 이 태스크를 한 번 수행해야 합니다.
- 실행 중인 IBM MQ 의 각 버전에 대해 Liberty 서버 인스턴스가 필요합니다. 예를 들어 IBM MQ 9.0.1의 큐 관리자용 시작된 태스크 MQWB0901 및 IBM MQ 9.0.2의 큐 관리자용 시작된 태스크 MQWB0902가 필요합니다.

  큐 관리자가 한 개뿐인 경우, Liberty 서버의 단일 시작된 태스크를 실행하고 큐 관리자를 마이그레이션할 때 사용하는 라이브러리를 변경할 수 있습니다.
- 이전 버전에서 마이그레이션할 때 카탈로그화된 프로시저를 수정해야 할 수 있습니다.

다음 프로시저를 수행하여 카탈로그화된 프로시저를 작성하십시오.

1. 샘플 시작된 태스크 프로시저 `thlqual.SCSQPROC(CSQ4WEBS)`를 프로시저 라이브러리에 복사하십시오.

   엔터프라이즈 표준에 따라 프로시저 이름을 지정하십시오.

   For example `MQWB0901`, indicating that this is the cataloged procedure for Liberty for IBM MQ 9.0.1
2. 샘플 프로시저 CSQ4WEBS의 지시사항을 사용하여 사용자 요구사항에 맞게 프로시저를 조정하십시오.

   Liberty 사용자 디렉토리는 **`crtmqweb.sh`** 스크립트를 실행하여 mqweb 서버 정의를 작성할 때 지정된 디렉토리임을 유의하십시오.

   자세한 내용은 647 페이지의 『Liberty 서버 정의 작성』의 내용을 참조하십시오.
3. 외부 보안 관리자 아래에서 실행하도록 프로시저에 권한을 부여하십시오.
4. **`S procname`** 명령을 사용하여 프로시저를 시작하십시오.

   +CWWKE0001I: 서버 mqweb가 시작되었다는 메시지가 생성됩니다.

   서버가 시작되지 않으면 메시지를 검토하십시오.

   프로시저가 시작되면 결과는 USERDIR 매개변수 아래의 파일에 저장됩니다. 예를 들어, 사용자 디렉토리가 `/u/mq/mqweb`일 경우 `/u/mq/mqweb/servers/mqweb/logs`를 확인하십시오.

   파일은 ASCII로 작성되기 때문에 일반 시스템 도구를 사용하여 파일을 볼 수 있습니다.
5. IBM 워크로드 관리자(WLM)를 사용하여 이 주소 공간을 분류하십시오.

   Liberty 서버는 IBM MQ 애플리케이션이며, 사용자는 이 애플리케이션과 상호 작용합니다. 애플리케이션에서 WLM에 높은 중요도를 지정할 필요가 없으므로 **STCUSER**의 서비스 클래스가 적합할 수 있습니다.
6. **`P procname`** 명령을 사용하여 프로시저를 중지하십시오.

   참고사항:

   a. 파일에는 소문자 데이터가 있기 때문에 멤버를 편집할 때 **대문자 해제**를 지정해야 합니다.
   b. 웹 서버를 시작하거나 종료하는 데는 상당한 시간(예: 1분 이상)이 걸릴 수 있습니다.

다음에 수행할 작업

사용자 및 역할 구성

관련 태스크

545 페이지의 『IBM MQ Console 및 REST API 구성』
IBM MQ Console 및 REST API를 호스팅하는 mqweb 서버에 기본 구성이 제공됩니다. 이 컴포넌트 중 하나를 사용하려면 사용자의 로그인을 허용하는 보안 구성 같은 여러 구성 태스크를 수행해야 합니다. 이 주제에서는 사용 가능한 모든 구성 옵션에 대해 설명합니다.

z/OS MQ Adv. VUE V 9.0.3 ReportingService (이전 BluemixRegistration) 스탠자 구성

이 태스크는 IBM Cloud (formerly Bluemix)에서 IBM Cloud Product Insights 서비스로 등록 및 사용 데이터를 공개하는 태스크의 일부입니다. 해당 IBM Cloud Product Insights 서비스가 더 이상 사용 가능하지 않습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

z/OS z/OS에서 큐 관리자 테스트

큐 관리자를 사용자 정의하거나 마이그레이션한 경우 IBM MQ for z/OS에서 제공하는 일부 샘플 애플리케이션 및 설치 확인 프로그램을 실행하여 이를 테스트할 수 있습니다.

이 태스크 정보

IBM MQ for z/OS를 설치 및 사용자 정의한 후, 제공되는 설치 확인 프로그램인 CSQ4IVP1을 사용하여 IBM MQ for z/OS가 작동하는지 확인할 수 있습니다.

기본 설치 확인 프로그램 CSQ4IVP1은 C, COBOL 또는 CICS 샘플을 사용하지 않고 비공유 큐를 테스트하고 기본 IBM MQ를 확인합니다.

기본 설치 확인을 실행한 후에는 서로 다른 큐에서 CSQ4IVP1을 사용하여 공유 큐를 테스트하고, Db2 및 커플링 기능이 올바르게 설정되었는지도 테스트할 수 있습니다. 분산 큐잉이 작동하는지 확인하는 데 제공된 설치 확인 프로그램, CSQ4IVPX를 사용할 수 있습니다.

CSQ4IVP1는 로드 모듈로 제공되며, 일련의 프로시저 샘플 애플리케이션을 MQI(Message Queue Interface)의 일반적인 사용을 보여주는 소스 모듈로 제공합니다. 이들 소스 모듈을 사용하여 여러 가지 프로그래밍 언어 환경을 테스트할 수 있습니다. 제공되는 샘플 JCL을 사용하여 설치에 적절한 다른 샘플을 컴파일하고 링크-편집할 수 있습니다.

프로시저

- z/OS에서 큐 관리자를 테스트하는 방법은 다음 하위 주제를 참조하십시오.
  - 650 페이지의 『기본 설치 확인 프로그램 실행』
  - 653 페이지의 『큐 공유 그룹 테스트』
  - 654 페이지의 『분산 큐잉 테스트』
  - 656 페이지의 『IBM MQ for z/OS로 C, C++, COBOL, PL/I 및 CICS 프로그램 테스트』

관련 태스크

592 페이지의 『z/OS에서 큐 관리자 구성』
IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

관련 정보

IBM MQ for z/OS 개념
z/OS에서 IBM MQ 환경 계획
IBM MQ for z/OS 관리

z/OS 기본 설치 확인 프로그램 실행

IBM MQ를 설치 및 사용자 정의한 후, 제공되는 설치 확인 프로그램인 CSQ4IVP1을 사용하여 IBM MQ가 작동하는지 확인할 수 있습니다.

기본 설치 확인 프로그램은 C, COBOL 또는 CICS 샘플을 사용하지 않고 기본 IBM MQ를 확인하는 배치 어셈블러 IVP입니다.

배치 어셈블러 IVP는 SMP/E에 의해 링크-편집되며 로드 모듈은 라이브러리 thlqual.SCSQLOAD에서 제공됩니다.

SMP/E APPLY 단계 및 사용자 정의 단계를 모두 완료한 후에 배치 어셈블러 IVP를 실행하십시오.

추가로 자세한 내용은 다음 절을 참조하십시오.

- CSQ4IVP1 애플리케이션 개요
- CSQ4IVP1 실행 준비
- CSQ4IVP1 실행
- CSQ4IVP1 결과 검사

CSQ4IVP1 애플리케이션 개요

CSQ4IVP1은 IBM MQ 서브시스템에 연결하여 다음 기본 기능을 수행하는 배치 애플리케이션입니다.

- IBM MQ 호출 발행
- 명령 서버와 통신
- 트리거가 활성 상태인지 확인
- 동적 큐 생성 및 삭제
- 메시지 만기 처리 확인
- 메시지 커미트 처리 확인

CSQ4IVP1 실행 준비

CSQ4IVP1을 실행하기 전에 다음을 수행하십시오.

1. IVP 항목이 큐 관리자 시동 프로그램에서 CSQINP2 데이터 세트 연결에 있는지 확인하십시오. IVP 항목은 멤버 thlqual.SCSQPROC(CSQ4IVPQ)에서 제공됩니다. 그렇지 않은 경우 CSQINP2 연결에 thlqual.SCSQPROC(CSQ4IVPQ)에서 제공되는 정의를 추가하십시오. 큐 관리자가 현재 실행 중인 경우 이러한 정의를 적용할 수 있도록 이를 재시작해야 합니다.
2. 설치 확인 프로그램을 실행하는 데 필요한 샘플 JCL인 CSQ4IVPR은 라이브러리 thlqual.SCSQPROC에 있습니다.

   IBM MQ 라이브러리에 대한 상위 레벨 규정자, 사용하려는 자국어(NL), 4자로 된 IBM MQ 큐 관리자 이름 및 작업 출력의 목적지로 CSQ4IVPR JCL을 사용자 정의하십시오.
3. IBM MQ 보안이 사용 중인 경우 CSQ4IVP1이 그의 자원에 액세스하도록 허용하려면 RACF를 업데이트하십시오.

   IBM MQ 보안이 사용으로 설정될 때 CSQ4IVP1 을 실행하려면 오브젝트에 액세스할 권한이 있는 RACF 사용자 ID가 필요합니다. RACF에 대한 자원 정의의 자세한 내용은 z/OS에서 보안 설정을 참조하십시오. IVP를 실행하는 사용자 ID에는 다음 액세스 권한이 필요합니다.

권한	프로파일	클래스
READ	ssid.DISPLAY.PROCESS	MQCMDS
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.COMMAND.REPLY.MODEL	MQQUEUE

권한	프로파일	클래스
UPDATE	ssid.CSQ4IVP1.**	MQQUEUE
READ	ssid.BATCH	MQCONN

이들 요구사항은 모든 IBM MQ 보안이 활성 상태라고 가정합니다. IBM MQ 보안을 활성화하기 위한 RACF 명령이 651 페이지의 그림 100에 표시됩니다. 이 예에서는 큐 관리자 이름이 CSQ1이고 샘플 CSQ4IVP1을 실행하는 개인의 사용자 ID는 TS101이라고 가정합니다.

```
RDEFINE MQCMDS CSQ1.DISPLAY.PROCESS
PERMIT CSQ1.DISPLAY.PROCESS CLASS(MQCMDS) ID(TS101) ACCESS(READ)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.INPUT
PERMIT CSQ1.SYSTEM.COMMAND.INPUT CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.REPLY.MODEL
PERMIT CSQ1.SYSTEM.COMMAND.REPLY.MODEL CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.CSQ4IVP1.**
PERMIT CSQ1.CSQ4IVP1.** CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQCONN CSQ1.BATCH
PERMIT CSQ1.BATCH CLASS(MQCONN) ID(TS101) ACCESS(READ)
```

그림 *100. CSQ4IVP1*에 대한 *RACF* 명령

CSQ4IVP1 실행

이 단계를 완료하면 큐 관리자를 시작하십시오. 큐 관리자가 이미 실행 중이고 CSQINP2를 변경하는 경우 큐 관리자를 중지하고 재시작해야 합니다.

IVP는 배치 작업으로 실행됩니다. 작업 카드를 사용자 정의하여 설치의 제출 요구사항을 충족하십시오.

CSQ4IVP1의 결과 검사

IVP는 10개 단계로 분할됩니다. 다음 단계가 실행되기 전에 각 단계는 완료 코드 0으로 완료되어야 합니다. IVP는 다음을 나열하는 보고서를 생성합니다.

- 연결된 큐 관리자의 이름.
- 각 단계에서 리턴된 완료 코드 및 이유 코드를 표시하는 1개 행의 메시지.
- 해당되는 경우 1개 행의 정보용 메시지.

샘플 보고서는 653 페이지의 그림 101에서 제공됩니다.

z/OS 완료 및 이유 코드에 대한 설명을 보려면 IBM MQ for z/OS 메시지, 완료 및 이유 코드를 참조하십시오.

일부 단계는 둘 이상의 IBM MQ 호출을 갖고 있으며, 장애 이벤트 시에 실패를 리턴한 특정 IBM MQ 호출을 표시하는 메시지가 발행됩니다. 또한 일부 단계의 경우 IVP는 설명 및 진단 정보를 주석 필드에 넣습니다.

IVP 작업은 특정 큐 관리자 오브젝트의 독점 제어를 요청하므로 시스템을 통해 단일 스레드 처리되어야 합니다. 그러나 큐 관리자에 대해 IVP가 실행할 수 있는 횟수에는 제한이 없습니다.

각 단계에서 수행하는 기능은 다음과 같습니다.

단계 1
: MQCONN API 호출을 발행하여 큐 관리자에 연결하십시오.

2단계
: 명령 서버에서 사용하는 시스템 명령 입력 큐의 이름을 판별하여 요청 메시지를 검색하십시오. 이 큐는 5단계의 표시 요청을 수신합니다.

이를 수행하려면 호출 순서는 다음과 같습니다.

1. 큐 관리자 이름을 지정하는 `MQOPEN` 호출을 발행하여 큐 관리자 오브젝트를 엽니다.
2. `MQINQ` 호출을 발행하여 시스템 명령 입력 큐의 이름을 찾습니다.
3. `MQINQ` 호출을 발행하여 다양한 큐 관리자 이벤트 스위치에 대해 알아냅니다.
4. `MQCLOSE` 호출을 발행하여 큐 관리자 오브젝트를 닫습니다.

이 단계가 성공적으로 완료되면 시스템 명령 입력 큐의 이름이 주석 필드에 표시됩니다.

3단계

`MQOPEN` 호출을 사용하여 이니시에이션 큐를 여십시오.

이 큐는 트리거 메시지를 예상하여 이 단계에서 열려 있으며, 이는 5단계의 요청에 응답하는 명령 서버의 결과로 도착합니다. 트리거 기준을 충족하기 위해 입력을 위해 큐를 열어야 합니다.

4단계

CSQ4IVP1.MODEL 큐를 모델로 사용하여 영구적 동적 큐를 작성하십시오. 동적 큐는 작성된 모델과 동일한 속성을 보유합니다. 즉, 5단계에서 명령 서버 요청의 응답이 이 큐에 기록되면 트리거 메시지는 3단계에서 열린 이니시에이션 큐에 기록됨을 의미합니다.

이 단계가 성공적으로 완료되면 영구적 동적 큐의 이름이 주석 필드에 표시됩니다.

5단계

명령 서버 명령 큐에 대한 `MQPUT1` 요청을 실행하십시오.

유형 MQMT_REQUEST의 메시지는 프로세스 CSQ4IVP1의 표시를 요청하는 시스템 명령 입력 큐에 기록됩니다. 메시지의 메시지 디스크립터는 명령 서버의 응답에 대한 응답 대상 큐로 4단계에서 작성된 영구적 동적 큐를 지정합니다.

6단계

이니시에이션 큐에서 **`MQGET`** 요청을 실행하십시오. 이 단계에서는 3단계에서 열린 이니시에이션 큐에 대해 1분 간격의 GET WAIT가 발행됩니다. 리턴된 메시지는 리플라이-투 큐에 기록되는 명령 서버의 응답 메시지에 의해 생성된 트리거 메시지가 될 것으로 예상됩니다.

7단계

4단계에서 작성된 영구 동적 큐를 삭제하십시오. 큐에 여전히 메시지가 있기 때문에, MQCO_PURGE_DELETE 옵션이 사용됩니다.

8단계

1. 동적 큐를 여십시오.
2. 설정된 만기 간격으로 메시지 MQPUT 작업을 수행하십시오.
3. 메시지가 만기되길 기다리십시오.
4. 만기된 메시지 MQGET 작업을 시도하십시오.
5. 큐 MQCLOSE 작업을 수행하십시오.

9단계

1. 동적 큐를 여십시오.
2. 메시지 MQPUT 작업을 수행하십시오.
3. MQCMIT를 실행하여 현재 작업 단위를 커미트하십시오.
4. 메시지 MQGET 작업을 수행하십시오.
5. MQBACK을 실행하여 메시지를 백아웃하십시오.
6. 동일한 메시지 MQGET을 수행하고 백아웃 수가 1로 설정되었는지 확인하십시오.
7. MQCLOSE를 실행하여 큐를 닫으십시오.

10단계

`MQDISC`를 사용하여 큐 관리자에서 연결을 끊으십시오.

IVP를 실행한 후 더 이상 필요하지 않은 오브젝트를 삭제할 수 있습니다.

IVP가 성공적으로 실행되지 않으면 수동으로 각 단계를 시도하여 실패한 기능을 확인하십시오.

```
DATE : 2005.035                  IBM MQ for z/OS - V6            PAGE : 0001
INSTALLATION VERIFICATION PROGRAM
PARAMETERS ACCEPTED. PROGRAM WILL CONNECT TO : CSQ1
,OBJECT QUALIFER : CSQ4IVP1
INSTALLATION VERIFICATION BEGINS :
STAGE 01 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR BRIDGE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS EXCP FOR CHANNEL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR SSL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR INHIBITED EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR LOCAL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR PERFORMENCE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR REMOTE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR START/STOP EVENTS
STAGE 02 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 SYSTEM.COMMAND.INPUT
STAGE 03 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 04 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9810EFEAC8980
STAGE 05 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 06 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 07 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 08 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9810F0070E645
STAGE 09 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9812BA8706803
STAGE 10 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
>>>>>>>>>>   END OF REPORT   <<<<<<<<<<
```

그림 *101. CSQ4IVP1*의 샘플 보고서

z/OS 큐 공유 그룹 테스트

기본 설치 확인 프로그램 CSQ4IVP1은 비공유 큐를 테스트합니다.

CSQ4IVP1은 큐 관리자가 큐 공유 그룹의 멤버이든 또는 아니든 간에 사용할 수 있습니다. 기본 IVP를 실행한 후, 다른 큐와 함께 CSQ4IVP1 설치 확인 프로그램을 사용하여 공유 큐에 대해 테스트할 수 있습니다. 또한 이것은 Db2 및 커플링 기능이 올바르게 설정되었는지 테스트합니다.

큐 공유 그룹에 대해 CSQ4IVP1 실행 준비

CSQ4IVP1을 실행하기 전에 다음을 수행하십시오.

1. 640 페이지의 『커플링 기능 설정』에서 설명하는 대로, IVP가 사용하는 커플링 기능 구조를 CFRM 정책 데이터 세트에 추가하십시오. 제공되는 샘플은 APPLICATION1이라는 구조를 사용하지만, 원하는 경우 변경할 수 있습니다.
2. IVP 항목이 큐 관리자 시동 프로그램에서 CSQINP2 데이터 세트 연결에 있는지 확인하십시오. IVP 항목은 thlqual.SCSQPROC(CSQ4IVPG) 멤버에서 제공됩니다. 제공되지 않는 경우 thlqual.SCSQPROC(CSQ4IVPG)에서 제공되는 정의를 CSQINP2 연결에 추가하십시오. 큐 관리자가 현재 실행 중인 경우 이러한 정의를 적용할 수 있도록 이를 재시작해야 합니다.
3. 필요한 경우 thlqual.SCSQPROC(CSQ4IVPG)에서 사용되는 커플링 기능 구조의 이름을 변경하십시오.
4. 큐 공유 그룹에 대한 설치 확인 프로그램을 실행하기 위해 필요한 샘플 JCL인 CSQ4IVPS는 thlqual.SCSQPROC 라이브러리에 있습니다.

   IBM MQ 라이브러리에 대한 상위 레벨 규정자, 사용하려는 자국어(NL), 4자로 된 IBM MQ 큐 관리자 이름 및 작업 출력의 목적지로 CSQ4IVPS JCL을 사용자 정의하십시오.
5. IBM MQ 보안이 사용 중인 경우 CSQ4IVP1이 그의 자원에 액세스하도록 허용하려면 RACF를 업데이트하십시오.

   IBM MQ 보안이 사용으로 설정될 때 CSQ4IVP1 을 실행하려면 오브젝트에 액세스할 권한이 있는 RACF 사용자 ID가 필요합니다. RACF에 대한 자원 정의의 자세한 내용은 z/OS에서 보안 설정을 참조하십시오. IVP를 실행하는 사용자 ID에는 기본 IVP를 실행하기 위해 필요한 권한 외에 다음 액세스 권한이 있어야 합니다.

권한	프로파일	클래스
UPDATE	ssid.CSQ4IVPG.**	MQQUEUE

   이들 요구사항은 모든 IBM MQ 보안이 활성 상태라고 가정합니다. IBM MQ 보안을 활성화하기 위한 RACF 명령이 654 페이지의 그림 102에 표시됩니다. 이 예에서는 큐 관리자 이름이 CSQ1이고 샘플 CSQ4IVP1을 실행하는 개인의 사용자 ID는 TS101이라고 가정합니다.

```
RDEFINE MQQUEUE CSQ1.CSQ4IVPG.**
PERMIT CSQ1.CSQ4IVPG.** CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)
```

그림 *102.* 큐 공유 그룹에 대한 *CSQ4IVP1*를 위한 *RACF* 명령

큐 공유 그룹을 위한 CSQ4IVP1 실행

이 단계를 완료하면 큐 관리자를 시작하십시오. 큐 관리자가 이미 실행 중이고 CSQINP2를 변경하는 경우 큐 관리자를 중지하고 재시작해야 합니다.

IVP는 배치 작업으로 실행됩니다. 작업 카드를 사용자 정의하여 설치의 제출 요구사항을 충족하십시오.

큐 공유 그룹에 대한 CSQ4IVP1의 결과 검사

작성된 큐를 CSQIVPG로 지칭하는 것을 제외하면 큐 공유 그룹의 IVP는 기본 IVP와 동일한 방식으로 작동합니다. *xx*로 부르는 외에는 기본 IVP와 동일한 방식으로 작동합니다. 651 페이지의 『CSQ4IVP1의 결과 검사』에서 제공되는 지시사항에 따라 큐 공유 그룹에 대한 IVP의 결과를 검사하십시오.

z/OS 분산 큐잉 테스트

제공된 설치 확인 프로그램인 CSQ4IVPX를 사용하여 분산 큐잉이 동작하는지 확인할 수 있습니다.

CSQ4IVPX 작업의 개요

CSQ4IVPX는 채널 시작기를 시작하고 IBM MQ DISPLAY CHINIT 명령을 실행하는 배치 작업입니다. 이것은 채널 및 네트워크 정의를 설정할 필요성을 피하면서 분산 큐잉의 모든 주요 측면이 작동하는지 확인합니다.

CSQ4IVPX 실행 준비

CSQ4IVPX을 실행하기 전에 다음을 수행하십시오.

1. 설치 확인 프로그램을 실행하는 데 필요한 샘플 JCL인 CSQ4IVPX가 thlqual.SCSQPROC 라이브러리에 있습니다.

   IBM MQ 라이브러리에 대한 상위 레벨 규정자, 사용하려는 자국어(NL), 4자로 된 큐 관리자 이름 및 작업 출력의 목적지로 CSQ4IVPX JCL을 사용자 정의하십시오.

2. IBM MQ 보안이 사용 중인 경우 CSQ4IVPX가 그의 자원에 액세스하도록 허용하려면 RACF를 업데이트하십시오. IBM MQ 보안이 사용으로 설정될 때 CSQ4IVPX를 실행하려면 오브젝트에 액세스할 권한이 있는 RACF 사용자 ID가 필요합니다. RACF에 대한 자원 정의의 자세한 내용은 z/OS에서 보안 설정을 참조하십시오. IVP를 실행하는 사용자 ID에는 다음 액세스 권한이 필요합니다.

권한	프로파일	클래스
CONTROL	ssid.START.CHINIT 및 ssid.STOP.CHINIT	MQCMDS
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.CSQUTIL.*	MQQUEUE
READ	ssid.BATCH	MQCONN
READ	ssid.DISPLAY.CHINIT	MQCMDS

   이들 요구사항은 연결 보안 프로파일 ssid.CHIN이 정의되었고(채널 시작기에 대한 연결 보안 프로파일에 표시됨) 모든 IBM MQ 보안이 활성 상태라고 가정합니다. 이를 수행하기 위한 RACF 명령은 655 페이지의 그림 103에 표시됩니다. 이 예는 다음을 가정합니다.

   - 큐 관리자 이름이 CSQ1입니다.
   - 샘플 CSQ4IVPX를 실행하는 사용자의 사용자 ID가 TS101입니다.

- 채널 시작기 주소 공간이 사용자 ID CSQ1MSTR 아래에서 실행 중입니다.

3. 채널 시작기 주소 공간이 다음 액세스 권한을 허용하도록 RACF를 업데이트하십시오.

권한	**프로파일**	**클래스**
READ	ssid.CHIN	MQCONN
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.CHANNEL.INITQ	MQQUEUE
UPDATE	ssid.SYSTEM.CHANNEL.SYNCQ	MQQUEUE
ALTER	ssid.SYSTEM.CLUSTER.COMMAND.QUEUE	MQQUEUE
UPDATE	ssid.SYSTEM.CLUSTER.TRANSMIT.QUEUE	MQQUEUE
ALTER	ssid.SYSTEM.CLUSTER.REPOSITORY.QUEUE	MQQUEUE
CONTROL	ssid.CONTEXT.**	MQADMIN

이를 수행하기 위한 RACF 명령은 655 페이지의 그림 103에 표시됩니다.

```
RDEFINE MQCMDS CSQ1.DISPLAY.DQM
PERMIT CSQ1.DISPLAY.DQM CLASS(MQCMDS) ID(TS101) ACCESS(READ)

RDEFINE MQCMDS CSQ1.START.CHINIT
PERMIT CSQ1.START.CHINIT CLASS(MQCMDS) ID(TS101) ACCESS(CONTROL)

RDEFINE MQCMDS CSQ1.STOP.CHINIT
PERMIT CSQ1.STOP.CHINIT CLASS(MQCMDS) ID(TS101) ACCESS(CONTROL)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.INPUT
PERMIT CSQ1.SYSTEM.COMMAND.INPUT CLASS(MQQUEUE) ID(TS101,CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CSQUTIL.*
PERMIT CSQ1.SYSTEM.CSQUTIL.* CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQCONN CSQ1.BATCH
PERMIT CSQ1.BATCH CLASS(MQCONN) ID(TS101) ACCESS(READ)

RDEFINE MQCONN CSQ1.CHIN
PERMIT CSQ1.CHIN CLASS(MQCONN) ID(CSQ1MSTR) ACCESS(READ)

RDEFINE MQQUEUE CSQ1.SYSTEM.CHANNEL.SYNCQ
PERMIT CSQ1.SYSTEM.CHANNEL.SYNCQ CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.COMMAND.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.COMMAND.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(ALTER)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.TRANSMIT.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.TRANSMIT.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.REPOSITORY.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.REPOSITORY.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(ALTER)

RDEFINE MQQUEUE CSQ1.SYSTEM.CHANNEL.INITQ
PERMIT CSQ1.SYSTEM.CHANNEL.INITQ CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQADMIN CSQ1.CONTEXT.**
PERMIT CSQ1.CONTEXT.** CLASS(MQADMIN) ID(CSQ1MSTR) ACCESS(CONTROL)
```

그림 *103. CSQ4IVPX*에 대한 *RACF* 명령

CSQ4IVPX 실행

이 단계를 완료하면 큐 관리자를 시작하십시오.

IVP는 배치 작업으로 실행됩니다. 작업 카드를 사용자 정의하여 설치의 제출 요구사항을 충족하십시오.

CSQ4IVPX의 결과 검사

CSQ4IVPX는 CSQUTIL IBM MQ 유틸리티를 실행하여 세 개의 MQSC 명령을 발행합니다. SYSPRINT 출력 데이터 세트는 세부사항은 큐 관리자 속성에 따라 다를 수 있지만 656 페이지의 그림 104와 비슷해야 합니다.

- 각각 여러 메시지가 뒤따르는 명령 **(1)**이 표시되어야 합니다.
- 각 명령의 마지막 메시지는 "CSQ9022I...이어야 합니다. 정상 완료" **(2)**.
- 작업은 전체적으로 리턴 코드 0 **(3)**으로 완료되어야 합니다.

```
CSQU000I CSQUTIL IBM MQ for z/OS - V6
CSQU001I CSQUTIL Queue Manager Utility - 2005-05-09 09:06:48
COMMAND
CSQU127I CSQUTIL Executing COMMAND using input from CSQUCMD data set
CSQU120I CSQUTIL Connecting to queue manager CSQ1
CSQU121I CSQUTIL Connected to queue manager CSQ1
CSQU055I CSQUTIL Target queue manager is CSQ1
START CHINIT
(1)
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000004
CSQM138I +CSQ1 CSQMSCHI CHANNEL INITIATOR STARTING
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' START CHINIT' NORMAL COMPLETION
(2)
DISPLAY CHINIT
(1)
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMDDQM DISPLAY CHINIT COMMAND ACCEPTED
CSQN205I  COUNT=   12, RETURN=00000000, REASON=00000000
CSQX830I +CSQ1 CSQXRDQM Channel initiator active
CSQX002I +CSQ1 CSQXRDQM Queue sharing group is QSG1
CSQX831I +CSQ1 CSQXRDQM 8 adapter subtasks started, 8 requested
CSQX832I +CSQ1 CSQXRDQM 5 dispatchers started, 5 requested
CSQX833I +CSQ1 CSQXRDQM 0 SSL server subtasks started, 0 requested
CSQX840I +CSQ1 CSQXRDQM 0 channel connections current, maximum 200
CSQX841I +CSQ1 CSQXRDQM 0 channel connections active, maximum 200,
including 0 paused
CSQX842I +CSQ1 CSQXRDQM 0 channel connections starting,
0 stopped, 0 retrying
CSQX836I +CSQ1 Maximum channels - TCP/IP 200, LU 6.2 200
CSQX845I +CSQ1 CSQXRDQM TCP/IP system name is TCPIP
CSQX848I +CSQ1 CSQXRDQM TCP/IP listener INDISP=QMGR not started
CSQX848I +CSQ1 CSQXRDQM TCP/IP listener INDISP=GROUP not started
CSQX849I +CSQ1 CSQXRDQM LU 6.2 listener INDISP=QMGR not started
CSQX849I +CSQ1 CSQXRDQM LU 6.2 listener INDISP=GROUP not started
CSQ9022I +CSQ1 CSQXCRPS ' DISPLAY CHINIT' NORMAL COMPLETION
(2)
STOP CHINIT
(1)
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMTCHI STOP CHINIT COMMAND ACCEPTED
CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' STOP CHINIT' NORMAL COMPLETION
(2)
CSQU057I CSQUCMDS 3 commands read
CSQU058I CSQUCMDS 3 commands issued and responses received, 0 failed
CSQU143I CSQUTIL 1 COMMAND statements attempted
CSQU144I CSQUTIL 1 COMMAND statements executed successfully
CSQU148I CSQUTIL Utility completed, return code=0
(3)
```

그림 *104. CSQ4IVPX*의 출력 예

z/OS IBM MQ for z/OS로 C, C++, COBOL, PL/I 및 CICS 프로그램 테스트

IBM MQ에서 제공되는 샘플 애플리케이션을 사용하여 C, C++, COBOL, PL/I 또는 CICS를 테스트할 수 있습니다.

IVP(CSQ4IVP1)가 로드 모듈로서 공급되며 샘플을 소스 모듈로 제공합니다. 이들 소스 모듈을 사용하여 여러 가지 프로그래밍 언어 환경을 테스트할 수 있습니다.

샘플 애플리케이션에 대한 자세한 정보는 IBM MQ for z/OS의 샘플 프로그램을 참조하십시오.

z/OS 다른 큐 관리자와의 통신 설정

이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

분산 큐잉 요구사항을 정의하려면 다음 항목을 정의해야 합니다.

- 채널 시작기 프로시저 및 데이터 세트 정의
- 채널 정의 정의
- 큐 및 기타 오브젝트 정의
- 액세스 보안 정의

분산 큐잉을 사용으로 설정하려면 다음 세 태스크를 수행해야 합니다.

- 분산 큐잉 기능을 사용자 정의하고 시스템 오브젝트 정의 및 593 페이지의 『z/OS에서 큐 관리자 사용자 정의 준비』에 설명된 대로 필요한 IBM MQ 오브젝트를 정의하십시오.
- z/OS에서 채널 시작기에 대한 보안 고려사항에 설명한 대로 액세스 보안을 정의하십시오.
- 675 페이지의 『z/OS에 대한 통신 설정』에 설명된 대로 통신을 설정하십시오.

큐 공유 그룹을 사용하는 경우 분산 큐잉 및 큐 공유 그룹의 내용을 참조하십시오.

IBM MQ for z/OS에서 분산 큐잉 사용에 대한 추가 고려사항에 대해서는 다음 절을 참조하십시오.

운영자 메시지

채널 시작기는 여러 비동기 조작 디스패처를 사용하기 때문에, 운영자 메시지는 연대순의 로그아웃에서 발생할 수 있습니다.

채널 조작 명령

채널 조작 명령은 일반적으로 두 단계와 관련됩니다. 명령 구문이 검사되고 채널 존재가 확인되면 요청이 채널 시작기에 전송됩니다. 메시지 CSQM134I 또는 CSQM137I가 명령 발행자에게 전송되어 첫 번째 단계의 완료를 표시합니다. 채널 시작기가 명령을 처리하면 성공 또는 실패를 표시하는 추가 메시지가 메시지 CSQ9022I 또는 CSQ9023I과 함께 명령 발행자에게 전송됩니다. 생성되는 모든 오류 메시지도 z/OS 콘솔로 전송될 수 있습니다.

그러나 DISPLAY CLUSQMGR를 제외한 모든 클러스터 명령은 비동기로 작동합니다. 오브젝트 속성을 변경하는 명령이 오브젝트를 업데이트하며 요청을 채널 시작기로 전송합니다. 클러스터 작업을 위한 명령이 구문에 대해 확인되며 요청이 채널 시작기로 전송됩니다. 두 경우 모두, 메시지 CSQM130I은 명령 발행자에게 전송되어 요청이 전송되었음을 표시합니다. 이 메시지는 메시지 CSQ9022I 다음에 나와 명령이 완료되어 요청이 전송되었음을 표시합니다. 클러스터 요청이 성공적으로 완료되었음을 나타내지는 않습니다. 채널 시작기에 전송된 요청은 클러스터의 다른 멤버에서 수신된 클러스터 요청과 함께 비동기로 처리됩니다. 일부 경우, 성공인지 실패인지를 판별하려면 이 요청은 전체 클러스터에 전송되어야 합니다. 모든 오류는 채널 시작기가 실행 중인 시스템의 z/OS에 보고됩니다. 명령 발행자에게 전송되지 않습니다.

미전달 메시지 큐

데드 레터 핸들러는 IBM MQ for z/OS에서 제공됩니다. 자세한 정보는 데드-레터 큐 핸들러 유틸리티 (CSQUDLQH)를 참조하십시오.

사용 중인 큐

메시지가 전송 중이 아니더라도 수신자 채널용 MCA는 목적지 큐를 계속 열어둘 수 있습니다. 이 작동은 큐가 '사용 중'인 것으로 나타나게 만듭니다.

보안 변경

사용자 ID에 대한 보안 액세스를 변경하는 경우, 변경사항은 즉시 적용되지 않습니다. (자세한 정보는 z/OS에서 채널 시작기의 보안 고려사항, 큐 보안을 위한 프로파일 및 608 페이지의 『ESM 보안 제어 구현』 참조.)

통신 중지됨 - TCP

TCP가 어떤 이유로 중지된 후 재시작되는 경우, TCP 포트에서 대기하는 IBM MQ for z/OS TCP 리스너가 중지됩니다.

자동 채널 다시 연결을 사용하면 채널 시작기가 TCP/IP가 사용 불가능임을 감지하여 TCP/IP가 리턴될 때 TCP/IP 리스너가 자동으로 다시 시작하도록 할 수 있습니다. 이 자동 재시작은 운영 스태프가 TCP/IP 문제를 알리고 수동으로 리스너를 다시 시작해야 하는 필요성을 완화시킵니다. 리스너가 작동하지 않는 동안 채널 시작기를 사용하여 LSTRTMR에 의해 지정된 간격으로 다시 리스너를 시도할 수 있다. 이러한 시도는 TCP/IP가 리턴하고 리스너가 자동 재시작을 성공할 때까지 계속됩니다. LSTRTMR에 대한 정보는 ALTER QMGR 및 분산 큐잉 메시지 (CSQX ...)를 참조하십시오.

통신 중지됨 - LU6.2

APPC가 중지되면 리스너도 중지됩니다. 다시, 이 경우에서는 리스너가 자동으로 LSTRTMR 간격으로 다시 시도되어, APPC가 다시 시작되면 리스너도 다시 시작될 수 있습니다.

Db2가 실패하는 경우, 이미 실행 중인 공유 채널은 계속 실행하지만 모든 새 채널 시작 요청은 실패합니다. Db2가 복원될 때 새 요청이 완료할 수 있습니다.

z/OS 자동 재시작 관리(ARM)

자동 재시작 관리(ARM)는 특정 배치 작업 또는 시작된 태스크(예: 서브시스템)의 가용성을 개선할 수 있는 z/OS 복구 기능입니다. 그러므로 생산적인 작업을 신속히 재개할 수 있습니다.

ARM을 사용하려면 큐 관리자와 채널 시작기를 특정 방법으로 설정하여 이들이 자동으로 재시작하게 해야 합니다. 자세한 정보는 z/OS 자동 재시작 관리자(ARM) 사용을 참조하십시오.

관련 개념

597 페이지의 『IBM MQ for z/OS 설정』
이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

660 페이지의 『z/OS에서 채널 모니터링 및 제어』
DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다.

675 페이지의 『z/OS에 대한 통신 설정』
분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 성공하려면 연결이 정의되고 사용 가능해야 합니다. 이 절에서는 연결을 정의하는 방법에 대해 설명합니다.

679 페이지의 『큐 공유 그룹으로 DQM을 위한 IBM MQ for z/OS 준비』
IBM MQ for z/OS에서 큐 공유 그룹을 갖는 분산 큐잉을 구성하려면 이 절의 지시사항을 사용하십시오.

682 페이지의 『큐 공유 그룹을 사용하는 IBM MQ for z/OS에 대한 통신 설정』
분산 큐잉 관리 채널이 시작되면, 채널 정의에서 지정된 연결을 사용하려고 합니다. 이 시도가 성공하려면 연결이 정의되고 사용 가능해야 합니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

z/OS IBM MQ 오브젝트 정의

IBM MQ 오브젝트를 정의하려면 IBM MQ 명령 입력 방법 중 하나를 사용하십시오. 이들 오브젝트 정의에 대한 추가 세부사항에 대해서는 이 토픽 내의 정보를 참조하십시오.

오브젝트 정의에 대한 정보는 660 페이지의 『z/OS에서 채널 모니터링 및 제어』의 내용을 참조하십시오.

전송 큐 및 트리거링 채널

다음을 정의하십시오.

- 각 송신 메시지 채널에 XMITQ를 사용하는 로컬 큐.
- 리모트 큐 정의.

  리모트 큐 오브젝트는 이름 및 내용이 지정되는 방식에 따라 세 가지 개별적인 용도를 갖습니다.

  - 리모트 큐 정의
  - 큐 관리자 알리어스 정의
  - 응답 대상 큐 알리어스 정의

  이런 세 가지 방법은 리모트 큐 정의 오브젝트를 사용하는 3가지 방법에 표시됩니다.

전송 큐에서 TRIGDATA 필드를 사용하여 지정된 채널을 트리거하십시오. 예를 들면 다음과 같습니다.

```
DEFINE QLOCAL(MYXMITQ) USAGE(XMITQ) TRIGGER +
INITQ(SYSTEM.CHANNEL.INITQ) TRIGDATA(MYCHANNEL)
DEFINE CHL(MYCHANNEL) CHLTYPE(SDR) TRPTYPE(TCP) +
XMITQ(MYXMITQ) CONNAME('9.20.9.30(1555)')
```

제공된 샘플 CSQ4INYD는 필수 정의에 대한 예를 추가로 제공합니다.

z/OS 공유 채널에 대한 동기화 큐가 정의되는 CF 구조에 대한 연결 유실 또는 비슷한 문제점이 일시적으로 채널이 시작하지 못하게 할 수 있습니다. 문제점 해결 후에 FIRST의 트리거 유형을 사용 중이고 채널이 트리거될 때 시작하지 못하는 경우 채널을 수동으로 시작해야 합니다. 문제점 해결 후에 트리거되는 채널을 자동으로 시작하기 원하는 경우 큐 관리자 TRIGINT 속성을 기본값이 아닌 다른 값으로 설정할 것을 고려하십시오. TRIGINT 속성을 기본값이 아닌 다른 값으로 설정하면 채널 시작기가 전송 큐에 메시지가 있는 중에 채널을 주기적으로 시작하려고 재시도합니다.

동기화 큐

DQM에는 순서 번호 및 논리 작업 단위 ID(LUWID)와 함께 사용할 큐가 필요합니다. 이름이 SYSTEM.CHANNEL.SYNCQ인 큐가 사용 가능한지 확인해야 합니다(z/OS 에 대한 계획 참조). 채널 시작기를 시작할 수 없는 경우를 제외하고는 이 큐를 사용할 수 있어야 합니다.

반드시 INDXTYPE(MSGID)을 사용하여 이들 큐를 정의하십시오. 이 속성은 채널에 액세스될 수 있는 속도를 향상시킵니다.

채널 명령 큐

채널 명령 큐가 SYSTEM.CHANNEL.INITQ 이름으로 사용자 시스템용으로 존재하는지 확인해야 합니다.

채널 시작기가 SYSTEM.CHANNEL.INITQ의 문제점을 감지하는 경우, 문제점이 정정될 때까지 정상적으로 계속할 수 없습니다. 문제점은 다음 중 하나일 수 있습니다.

- 큐가 가득 참
- 큐가 넣기(put)용으로 사용할 수 없음
- 큐가 있는 페이지 세트가 가득 찼음
- 채널 시작기가 큐에 적절한 보안 권한을 가지고 있지 않음

채널 시작기가 실행 중인 동안 큐의 정의가 GET(DISABLED)로 변경되는 경우, 시작기는 큐에서 메시지를 가져올 수 없으며 종료합니다.

채널 시작기 시작

트리거링은 채널 시작기를 사용하여 구현됩니다. IBM MQ for z/OS에서는 시작기가 MQSC 명령 START CHINIT로 시작됩니다.

채널 시작기 중지

채널 시작기는 큐 관리자를 중지시킬 때 자동으로 중지됩니다. 채널 시작기를 중지해야 하지만 큐 관리자는 그럴 필요가 없는 경우, MQSC 명령 `STOP CHINIT`를 사용할 수 있습니다.

z/OS z/OS에서 채널 모니터링 및 제어

DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다.

각 z/OS 큐 관리자는 고유 z/OS 기능을 사용하여 리모트 큐 관리자에 대한 상호 연결 제어를 위한 DQM 프로그램(채널 시작기)을 갖고 있습니다.

z/OS에서 이들 패널 및 명령의 구현은 조작과 제어판 및 MQSC 명령에 통합됩니다. 두 세트의 패널 및 명령 조직에는 차이가 없습니다.

또한 프로그래밍 가능 명령 형식(PCF) 명령을 사용하여 명령을 입력할 수도 있습니다. 이들 명령 사용에 대한 정보는 관리 태스크 자동화를 참조하십시오.

이 절의 정보는 채널 시작기가 분산 큐잉을 위해 사용되는 모든 경우에 적용됩니다. 큐 공유 그룹 또는 그룹 내 큐잉을 사용하는지 여부를 적용합니다.

DQM 채널 제어 기능

분산 큐 관리 모델에 대한 개요는 148 페이지의 『메시지 송신 및 수신』의 내용을 참조하십시오.

채널 제어 기능은 패널, 명령 및 프로그램, 두 개의 동기화 큐, 채널 명령 큐, 그리고 채널 정의로 구성됩니다. 이 토픽은 채널 제어 기능의 컴포넌트에 대한 간단한 설명입니다.

- 채널 정의는 페이지 세트 0 또는 z/OS의 다른 IBM MQ 오브젝트처럼 Db2에서 오브젝트로 보유됩니다.
- 조작 및 제어판, MQSC 명령 또는 PCF 명령을 사용하여 다음을 수행합니다.
  - 채널 정의를 작성, 복사, 표시, 대체 및 삭제
  - 채널 시작기 및 리스너를 시작 및 중지
  - 채널 시작, 중지, ping, 채널 순서 번호 재설정, 링크를 재설정할 수 없을 때 인다우트 메시지 해결
  - 채널에 대한 상태 정보 표시
  - DQM에 대한 정보 표시

  특히, CSQINPX 초기화 입력 데이터 세트를 사용하여 MQSC 명령을 실행할 수 있습니다. 채널 시작기를 시작할 때마다 이 세트를 처리할 수 있습니다. 자세한 정보는 초기화 명령을 참조하십시오.
- 채널 재 동기화를 위해 두 가지 큐(SYSTEM.CHANNEL.SYNCQ 및 SYSTEM.QSG.CHANNEL.SYNCQ)를 사용할 수 있습니다. 성능상의 이유로 이 큐를 INDXTYPE(MSGID)으로 정의하십시오.
- 채널 명령 큐(SYSTEM.CHANNEL.INITQ)는 채널 시작기, 채널 및 리스너에 대한 명령을 보유하는 데 사용됩니다.
- 채널 제어 기능 프로그램은 큐 관리자와는 별도의 자체 주소 공간에서 수행되며, 채널 시작기, 리스너, MCA, 트리거 모니터 및 명령 핸들러로 구성됩니다.
- 큐 공유 그룹 및 공유 채널의 경우 공유 큐 및 큐 공유 그룹을 참조하십시오.
- 그룹 내 큐잉의 경우, 그룹 내 큐잉을 참조하십시오.

z/OS에서 채널 관리

채널, 채널 시작기 및 리스너를 관리하는 방법에 대한 정보는 다음 표의 링크를 사용하십시오.

표 *49.* 채널 태스크

수행할 태스크	MQSC 명령어
채널 정의	DEFINE CHANNEL
채널 정의 대체	ALTER CHANNEL

표 49. 채널 태스크 (계속)	
수행할 태스크	**MQSC 명령어**
채널 정의 표시	DISPLAY CHANNEL
채널 정의 삭제	채널 삭제
채널 시작기 시작	START CHINIT
채널 시작기 중지	STOP CHINIT
채널 시작기 정보 표시	DISPLAY CHINIT
채널 리스너 시작	START LISTENER
채널 리스너 중지	STOP LISTENER
채널 시작	START CHANNEL
채널 테스트	PING CHANNEL
채널의 메시지 순서 번호 재설정	RESET CHANNEL
채널에서 인다우트 메시지 해석	RESOLVE CHANNEL
채널 중지	STOP CHANNEL
채널 상태 표시	DISPLAY CHSTATUS
클러스터 채널 표시	DISPLAY CLUSQMGR

관련 개념

661 페이지의 『패널 및 명령 사용』
DQM을 관리하기 위해 MQSC 명령, PCF 명령 또는 조작 및 제어판을 사용할 수 있습니다.

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

597 페이지의 『IBM MQ for z/OS 설정』
이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

675 페이지의 『z/OS에 대한 통신 설정』
분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 성공하려면 연결이 정의되고 사용 가능해야 합니다. 이 절에서는 연결을 정의하는 방법에 대해 설명합니다.

679 페이지의 『큐 공유 그룹으로 DQM을 위한 IBM MQ for z/OS 준비』
IBM MQ for z/OS에서 큐 공유 그룹을 갖는 분산 큐잉을 구성하려면 이 절의 지시사항을 사용하십시오.

682 페이지의 『큐 공유 그룹을 사용하는 IBM MQ for z/OS에 대한 통신 설정』
분산 큐잉 관리 채널이 시작되면, 채널 정의에서 지정된 연결을 사용하려고 합니다. 이 시도가 성공하려면 연결이 정의되고 사용 가능해야 합니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

z/OS 패널 및 명령 사용

DQM을 관리하기 위해 MQSC 명령, PCF 명령 또는 조작 및 제어판을 사용할 수 있습니다.

MQSC 명령의 구문에 대한 정보는 스크립트(MQSC) 명령을 참조하십시오. PCF 명령에 관한 정보는 프로그램 가능 명령 형식 소개를 참조하십시오.

초기 패널 사용

조작 및 제어판 호출, 기능 키 사용 및 도움말 보기에 대한 소개는 IBM MQ for z/OS 관리의 내용을 참조하십시오.

참고: 조작 및 제어판을 사용하려면 올바른 보안 권한이 있어야 합니다. 자세한 정보는 IBM MQ for z/OS 관리 및 하위 주제를 참조하십시오. 662 페이지의 그림 105에서는 패널 세션을 시작할 때 표시되는 패널을 보여줍니다. 패널 뒤의 텍스트가 이 패널에서 수행할 조치를 설명합니다.

```
IBM MQ for z/OS - Main Menu

Complete fields. Then press Enter.

Action . . . . . . . . . . 1   0. List with filter  4. Manage
1. List or Display  5. Perform
2. Define like    6. Start
3. Alter       7. Stop
8. Command
Object type . . . . . . . . CHANNEL    +
Name . . . . . . . . . . . *
Disposition . . . . . . . . A Q=Qmgr, C=Copy, P=Private, G=Group,
S=Shared, A=All

Connect name . . . . . . . MQ25 - local queue manager or group
Target queue manager . . . MQ25
- connected or remote queue manager for command input
Action queue manager . . . MQ25 - command scope in group
Response wait time . . . . 10  5 - 999 seconds

(C) Copyright IBM Corporation 1993, 2023. All rights reserved.

Command ===> ________________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Prompt  F9=SwapNext F10=Messages
F12=Cancel
```

그림 *105.* 조작 및 제어 초기 패널

이 패널에서 다음을 수행할 수 있습니다.

- **조치** 필드에 적절한 번호를 입력하여 실행하려는 조치를 선택하십시오.
- 작업을 수행하려는 오브젝트 유형을 지정하십시오. 오브젝트 유형이 무엇인지 확인할 수 없으면, F4를 눌러 오브젝트 유형의 목록을 표시하십시오.
- 지정된 유형의 오브젝트 목록을 표시하십시오. 이미 이 서브시스템에서 정의된 오브젝트(또는 지정된 유형)의 목록을 표시하려면 **이름** 필드에 별표(*)를 입력하고 Enter를 누르십시오. 그런 다음 작업을 수행하려는 하나 이상의 오브젝트를 차례로 선택할 수 있습니다. 663 페이지의 그림 106에서는 이 방식으로 생성된 채널의 목록을 보여줍니다.
- **속성 지정** 필드에 작업할 오브젝트의 큐 공유 그룹에 대한 파일 속성을 지정하십시오. 속성 지정은 오브젝트가 보관된 위치 및 오브젝트의 행동 방법을 결정합니다.
- **연결 이름** 필드에서 로컬 큐 관리자 또는 연결할 큐 공유 그룹을 선택하십시오. 리모트 큐 관리자에서 명령을 실행하려는 경우에는 리모트 큐 관리자가 큐 공유 그룹의 멤버인지 여부에 따라 **대상 큐 관리자** 필드나 **조치 큐 관리자** 필드를 선택하십시오. 리모트 큐 관리자가 큐 공유 그룹의 멤버가 아닌 경우에는 **대상 큐 관리자** 필드를 선택하십시오. 리모트 큐 관리자가 큐 공유 그룹의 멤버인 경우에는 **조치 큐 관리자** 필드를 선택하십시오.
- **응답 대기 시간** 필드에서 응답 수신에 대한 대기 시간을 선택하십시오.

```
List Channels - MQ25          Row 1 of 8

Type action codes, then press Enter. Press F11 to display connection status.
1=Display  2=Define like  3=Alter  4=Manage  5=Perform
6=Start   7=Stop

Name         Type      Disposition  Status
<> *           CHANNEL    ALL   MQ25
_  SYSTEM.DEF.CLNTCONN  CLNTCONN   QMGR  MQ25
_  SYSTEM.DEF.CLUSRCVR  CLUSRCVR   QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.CLUSSDR  CLUSSDR    QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.RECEIVER  RECEIVER   QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.REQUESTER REQUESTER   QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.SENDER   SENDER    QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.SERVER   SERVER    QMGR  MQ25 INACTIVE
_  SYSTEM.DEF.SVRCONN  SVRCONN    QMGR  MQ25 INACTIVE
******** End of list ********



Command ===> ________________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd    F9=SwapNext F10=Messages F11=Status  F12=Cancel
```

그림 *106.* 채널 목록

z/OS *z/OS* 에서 채널 정의

z/OS에서 MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널을 정의할 수 있습니다.

MQSC 채널을 사용하여 채널을 정의하려면 DEFINE CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	가치
Action	2(유사 정의)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	
처리	새 오브젝트의 위치

정의하는 채널에 필요한 이름 및 속성 정보를 채우기 위한 일부 패널이 제시됩니다. 이들은 기본 속성 값으로 초기화되어 있습니다. Enter를 누르기 전에 원하는 모든 것을 변경하십시오.

참고: **object type** 필드에 CHANNEL을 입력한 경우 유효한 채널 유형 선택 패널이 먼저 표시됩니다.

기존 채널과 동일한 속성을 가진 채널을 정의하려면 초기 패널의 **Name** 필드에 복사하려는 채널의 이름을 입력하십시오. 패널이 기존 오브젝트의 속성으로 초기화됩니다.

채널 속성에 관한 정보는 채널 속성을 참조하십시오.

참고:

1. 네트워크의 모든 채널 이름을 고유하게 지정하십시오. 모든 채널을 표시하는 네트워크 다이어그램에 표시된 것처럼, 채널 이름에 소스 및 대상 큐 관리자 이름을 포함하는 것이 이 이름 지정을 수행할 좋은 방법입니다.

채널을 정의한 후 채널을 보호해야 합니다. 665 페이지의 『채널 보안』의 내용을 참조하십시오.

z/OS 채널 정의 대체

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 정의를 대체할 수 있습니다.

MQSC 명령을 사용해서 채널 정의를 변경하려면 ALTER CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	3(대체)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	CHANNEL.TO.ALTER
처리	저장된 오브젝트의 위치

채널의 현재 속성에 관한 정보가 들어 있는 몇몇 패널이 표시됩니다. 새 값을 겹쳐 입력하여 원하는 모든 비보호 필드를 변경한 후, Enter를 눌러서 채널 정의를 변경하십시오.

채널 속성에 관한 정보는 채널 속성을 참조하십시오.

z/OS 채널 정의 표시

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 정의를 표시할 수 있습니다.

MQSC 명령을 사용해서 채널 정의를 표시하려면 DISPLAY CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	1(목록 또는 표시)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	CHANNEL.TO.DISPLAY
처리	오브젝트의 위치

채널의 현재 속성에 관한 정보를 표시하는 일부 패널이 표시됩니다.

채널 속성에 관한 정보는 채널 속성을 참조하십시오.

z/OS 채널 정의 삭제

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 정의를 삭제할 수 있습니다.

MQSC 명령을 사용해서 채널 정의를 삭제하려면 DELETE CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	4(관리)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	CHANNEL.TO.DELETE
처리	오브젝트의 위치

별도의 패널이 표시됩니다. 이 패널에서 기능 유형 1을 선택하십시오.

채널 정의를 삭제하려면 Enter를 누르십시오. Enter를 다시 눌러서 채널 정의를 삭제하기 원하는지 확인하도록 요청받습니다.

참고: 채널 정의를 삭제하려면 채널 시작기가 실행 중이어야 합니다(클라이언트-연결 채널은 예외).

z/OS 채널 시작기에 관한 정보 표시

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 시작기에 관한 정보를 표시할 수 있습니다.

MQSC 명령을 사용하여 채널 시작기에 관한 정보를 표시하려면 DISPLAY CHINIT를 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	1(표시)
오브젝트 유형	SYSTEM
이름	공백

별도의 패널이 표시됩니다. 이 패널에서 기능 유형 1을 선택하십시오.

참고:

1. 많은 채널이 있는 경우 분산 큐잉 정보 표시는 약간의 시간이 걸릴 수 있습니다.
2. 채널 시작기는 분산 큐잉에 대한 정보를 표시하기 전에 실행되어야 합니다.

z/OS 채널 보안

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널을 보호할 수 있습니다.

MQSC 명령을 사용하여 채널 보안을 설정하려면 SET CHLAUTH를 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	가치
Action	8

MQSC 명령을 제공할 수 있는 편집기(이 경우 CHLAUTH 명령)가 표시됩니다(665 페이지의 그림 107참조). 명령 입력을 마쳤을 때 더하기 부호(+)가 필요합니다. 편집기에서 나가서 명령을 명령 서버로 제출하려면 PF3를 입력하십시오.

```
****************************** Top of Data ******************************
000001 SET CHLAUTH(SYSTEM.DEF.SVRCONN) +
000002 TYPE(SSLPEERMAP) +
000003 SSLPEER('CN="John Smith"') +
000004 MCAUSER('PUBLIC')
****** **************************** Bottom of Data ****************************

Command ===>                         Scroll ===> PAGE
F1=Help   F3=Exit   F4=LineEdit F12=Cancel
```

그림 *107.* 명령 입력 항목

명령의 출력은 다음과 같이 표시됩니다. 665 페이지의 그림 108을(를) 참조하십시오.

```
****** ****************************** Top of Data ******************************
000001 CSQU000I CSQUTIL IBM MQ for z/OS 7.1.0
000002 CSQU001I CSQUTIL Queue Manager Utility - 2011-04-20 14:42:58
000003 COMMAND TGTQMGR(MQ23) RESPTIME(30)
000004 CSQU127I Executing COMMAND using input from CSQUCMD data set
000005 CSQU120I Connecting to MQ23
000006 CSQU121I Connected to queue manager MQ23
000007 CSQU055I Target queue manager is MQ23
000008 SET CHLAUTH(SYSTEM.DEF.SVRCONN) +
000009 TYPE(SSLPEERMAP) +
000010 SSLPEER('CN="John Smith"') +
000011 MCAUSER('PUBLIC')
000012 CSQN205I  COUNT=    2, RETURN=00000000, REASON=00000000
000013 CSQ9022I !MQ23 CSQMSCA ' SET CHLAUTH' NORMAL COMPLETION
000014 CSQU057I 1 commands read
000015 CSQU058I 1 commands issued and responses received, 0 failed
000016 CSQU143I 1 COMMAND statements attempted
000017 CSQU144I 1 COMMAND statements executed successfully
000018 CSQU148I CSQUTIL Utility completed, return code=0
Command ===>                         Scroll ===> PAGE
F1=Help   F3=Exit   F5=Rfind   F6=Rchange  F9=SwapNext F12=Cancel
```

그림 *108.* 명령 출력

z/OS 채널 시작기 시작

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 시작기를 시작할 수 있습니다.

MQSC 명령을 사용해서 채널 정의를 시작하려면 START CHINIT를 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	6(시작)
오브젝트 유형	SYSTEM
이름	공백

시스템 기능 시작 패널이 표시됩니다. 다음 패널 뒤에 있는 텍스트가 취할 조치를 설명합니다.

```
Start a System Function

Select function type, complete fields, then press Enter to start system
function.

함수 유형........ 1. 채널 시작기
2. Channel listener
조치 큐 관리자.... : MQ25

채널 시작기
JCL 대체...... ________________________________________________
________________________________________________

Channel listener
인바운드 처리........ Q G=Group, Q=Qmgr
전송 유형...... _ L=LU6.2, T = TCP/IP
LU 이름 (LU6.2)..... _________________
포트 번호 (TCP/IP)..... 도면 1
IP 주소 (TCP/IP).... ________________________________________________



명령 == = > ________________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

그림 *109.* 시스템 기능 시작

기능 유형 1(채널 시작기)을 선택하고 Enter를 누르십시오.

z/OS 채널 시작기 중지

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 시작기를 중지할 수 있습니다.

MQSC 명령을 사용해서 채널 정의를 중지하려면 STOP CHINIT를 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	7(중지)
오브젝트 유형	SYSTEM
이름	공백

시스템 기능 중지 패널이 표시됩니다. 패널 뒤에 있는 텍스트가 이 패널 사용 방법을 설명합니다.

```
Stop a System Function

Select function type, complete fields, then press Enter to stop system
function.

함수 유형........ 1. 채널 시작기
2. Channel listener
조치 큐 관리자.... : MQ25

채널 시작기
Restart shared channels  Y Y=Yes, N=No

Channel listener
인바운드 처리........ Q G=Group, Q=Qmgr
전송 유형...... _ L=LU6.2, T = TCP/IP

포트 번호 (TCP/IP)..... _____
IP 주소 (TCP/IP).... ________________________________________________



명령 == = > ________________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

그림 *110.* 기능 제어 중지

기능 유형 1(채널 시작기)를 선택하고 Enter를 누르십시오.

채널 시작기는 중지하기 전에 모든 실행 중인 채널이 일시정지 모드에서 중지하기를 기다립니다.

참고: 일부 채널이 실행 중이지만 활성이 아닌 수신자 또는 요청자 채널인 경우, 수신자 또는 송신자 채널 시작기로 발행된 중지 요청은 즉시 중지하게 만듭니다.

그러나 메시지가 흐르고 있는 경우에는 채널 시작기가 중지하기 전에 현재 메시지 배치가 완료되기를 기다립니다.

z/OS 채널 리스너 시작

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 리스너를 시작할 수 있습니다.

MQSC 명령을 사용해서 채널 리스너를 시작하려면 START LISTENER를 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	**값**
Action	6(시작)
오브젝트 유형	SYSTEM
이름	공백

시스템 기능 패널이 표시됩니다(666 페이지의 그림 109 참조).

기능 유형 2(채널 리스너)를 선택하십시오. 인바운드 속성 지정을 선택하십시오. 전송 유형을 선택하십시오. 전송 유형이 L이면 LU 이름을 선택하십시오. 전송 유형이 T인 경우에는 포트 번호 및 IP 주소(선택적으로)를 선택하십시오. Enter를 누르십시오.

참고: TCP/IP 리스너의 경우, 포트 및 IP 주소의 복수 조합을 시작할 수 있습니다.

z/OS 채널 리스너 중지

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 리스너를 중지할 수 있습니다.

MQSC 명령을 사용해서 채널 리스너를 중지하려면 STOP LISTENER를 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	7(중지)
오브젝트 유형	SYSTEM
이름	공백

시스템 기능 중지 패널이 표시됩니다(667 페이지의 그림 110 참조).

기능 유형 2(채널 리스너)를 선택하십시오. 인바운드 속성 지정을 선택하십시오. 전송 유형을 선택하십시오. 전송 유형이 'T'이면 포트 번호 및 IP 주소(선택적으로)를 선택하십시오. Enter를 누르십시오.

참고: TCP/IP 리스너의 경우에는 포트 및 IP 주소의 특정 조합, 또는 모든 조합을 중지시킬 수 있습니다.

z/OS 채널 시작

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널을 시작할 수 있습니다.

MQSC 명령을 사용해서 채널을 시작하려면 START CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	6(시작)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	CHANNEL.TO.USE
처리	오브젝트의 속성 지정

채널 시작 패널이 표시됩니다. 패널 뒤에 있는 텍스트가 패널 사용 방법을 설명합니다.

```
Start a Channel

Select disposition, then press Enter to start channel.


채널 이름....... : CHANNEL.TO.USE
채널 유형...... : 송신자
설명......... : CHANNEL.TO.USE 설명


처리.......... P   P=Private on MQ25
S=Shared on MQ25
A=Shared on any queue manager








명령 == = > _______________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

그림 *111.* 채널 시작

채널 인스턴스의 속성 지정 및 시작되는 큐 관리자를 선택하십시오.

채널을 시작하려면 Enter를 누르십시오.

z/OS 공유 채널 시작

공유 채널을 시작하고 지정된 채널 시작기에서 이를 유지하려면, 속성 지정 = S(START CHANNEL 명령에서 CHLDISP(FIXSHARED) 지정)를 사용하십시오.

한 번에 하나의 공유 채널 인스턴스만이 실행 중일 수 있습니다. 채널의 두 번째 인스턴스를 시작하려는 시도는 실패합니다.

이 방법으로 채널을 시작하려는 경우, 채널에 다음 규칙을 적용하십시오.

- 큐 공유 그룹의 모든 큐 관리자로부터 채널을 중지할 수 있습니다. 채널이 시작된 채널 시작기가 채널 중지 요청을 발행하는 순간에 실행 중이 아니어도 채널을 중지할 수 있습니다. 채널이 중지된 경우, 동일한 채널 또는 다른 채널 시작기에 속성 지정 = S(CHLDISP(FIXSHARED))를 지정하여 재시작할 수 있습니다. 또는 속성 지정 = A(CHLDISP(SHARED))를 지정하여 시작할 수도 있습니다.
- 채널을 시작하고 있는 중이거나 다시 시도하려면 동일 채널 또는 다른 채널 시작기에서 속성 지정 = S(CHLDISP(FIXSHARED))를 지정하여 재시작할 수 있습니다. 또는 속성 지정 = A(CHLDISP(SHARED))를 지정하여 시작할 수도 있습니다.
- 채널이 활성 상태에 있을 경우 채널은 시작된 트리거로 있어야 합니다. 시작된 트리거인 공유 채널은 항상 공유 속성 지정(CHLDISP(SHARED))을 가지고 있습니다.
- 채널이 활성 상태에 있을 경우 모든 채널 시작기에서 CHLDISP(FIXSHARED)로 시작되어야 합니다. 또는 속성 지정 = A(CHLDISP(SHARED))를 지정하여 시작할 수도 있습니다.
- 채널이 시작된 채널 시작기가 SHARED(RESTART)로 중지될 때 또는 채널 시작기가 비정상적으로 종료할 때 큐 공유 그룹의 다른 어떤 활성 채널 시작기도 채널을 복구하지 않습니다. 채널은 채널이 시작된 채널 시작기가 다음에 재시작될 때만 복구됩니다. 이는 워크로드에 추가하려는 공유 큐 공유 그룹에서 다른 채널 시작기에 전달된 실패한 채널 복구 시도를 중지합니다.

z/OS 채널 테스트

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널을 테스트할 수 있습니다.

MQSC 명령을 사용해서 채널을 테스트하려면 PING CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	5(수행)
오브젝트 유형	SENDER, SERVER 또는 CHANNEL
이름	CHANNEL.TO.USE
처리	채널 오브젝트의 속성 지정

채널 기능 수행 패널이 표시됩니다. 패널 뒤에 있는 텍스트가 패널 사용 방법을 설명합니다.

```
Perform a Channel Function

Select function type, complete fields, then press Enter.


함수 유형........ 1. 3재설정. 커미트로 해결하기
2. 핑 4. 백아웃으로 해결하기

채널 이름....... : CHANNEL.TO.USE
채널 유형...... : 송신자
설명......... : CHANNEL.TO.USE 설명


처리.......... P   P=Private on MQ25
S=Shared on MQ25
A=Shared on any queue manager

재설정을 위한 순서 번호... 1-999999999
핑에 대한 데이터 길이...... 16-32768



명령 == = > _______________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

그림 *112.* 채널 테스트

기능 유형 2(ping)를 선택하십시오.

테스트가 실행되는 채널의 속성 지정 및 테스트되는 큐 관리자를 선택하십시오.

데이터 길이는 초기에 16으로 설정됩니다. 원하는 경우 변경하고 Enter를 누르십시오.

z/OS 채널의 메시지 순서 번호 재설정

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널에 대한 메시지 순서 번호를 재설정할 수 있습니다.

MQSC 명령을 사용해서 채널 순서 번호를 재설정하려면 RESET CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	5(수행)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	CHANNEL.TO.USE
처리	채널 오브젝트의 속성 지정

채널 기능 수행 패널이 표시됩니다(670 페이지의 그림 112 참조).

기능 유형 1(재설정)을 선택하십시오.

재설정이 실행되는 채널의 속성 지정 및 실행되는 큐 관리자를 선택하십시오.

순서 번호 필드는 초기에 1로 설정됩니다. 원하는 경우 이 값을 변경하고 Enter를 누르십시오.

z/OS 채널상의 인다우트*(in-doubt)* 메시지 해결

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 인다운트 메시지를 해석할 수 있습니다.

MQSC 명령을 사용해서 채널상의 인다우트 메시지를 해결하려면 RESOLVE CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	5(수행)

필드	값
오브젝트 유형	SENDER, SERVER 또는 CHANNEL
이름	CHANNEL.TO.USE
처리	오브젝트의 속성 지정

채널 기능 수행 패널이 표시됩니다(670 페이지의 그림 112 참조).

기능 유형 3 또는 4(커미트 또는 백아웃으로 해석)를 선택하십시오. (자세한 정보는 165 페이지의 『인다우트(in-doubt) 채널』 참조)

해석이 실행되는 채널의 속성 지정 및 실행되는 큐 관리자를 선택하십시오. Enter를 누르십시오.

z/OS 채널 중지

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널을 중지할 수 있습니다.

MQSC 명령을 사용해서 채널을 중지하려면 STOP CHANNEL을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	7(중지)
오브젝트 유형	채널 유형(예: SENDER) 또는 CHANNEL
이름	CHANNEL.TO.USE
처리	오브젝트의 속성 지정

채널 중지 패널이 표시됩니다. 패널 뒤에 있는 텍스트가 패널 사용 방법을 설명합니다.

```
Stop a Channel

Complete fields, then press Enter to stop channel.


채널 이름....... : CHANNEL.TO.USE
채널 유형...... : 송신자
설명......... : CHANNEL.TO.USE 설명


처리.......... P   P=Private on MQ25
A=Shared on any queue manager

중단 모드.......... 도면 1 2. 강제 실행
중지 상태......... 도면 1 중지 2. 비활성

큐 관리자........ ________________________________________________
연결 이름........ ________________________________________________



명령 == = > ________________________________________________________________
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

그림 *113.* 채널 중지

중지가 실행되는 채널의 속성 지정 및 중지되는 큐 관리자를 선택하십시오.

필요한 중지 모드를 선택하십시오.

일시정지
: 채널은 배치 크기 값에 도달되지 않았고 이미 전송 큐에서 대기 중인 메시지가 있는 경우에도 현재 메시지가 완료되고 배치가 종료될 때 중지합니다. 새 배치는 시작되지 않습니다. 이 모드가 기본값입니다.

강제 실행
: 채널이 즉시 중지합니다. 메시지의 배치가 진행 중인 경우 '인다우트' 상황이 발생할 수 있습니다.

중지하려는 채널의 큐 관리자와 연결 이름을 선택하십시오.

필요한 상태를 선택하십시오.

중지됨
: 채널이 자동으로 재시작되지 않으며, 수동으로 재시작되어야 합니다. 큐 관리자나 연결 이름이 지정되지 않은 경우 이 모드가 기본값입니다. 이름이 지정된 경우에는 이것이 허용되지 않습니다.

비활성
: 채널은 필요할 때 자동으로 재시작됩니다. 큐 관리자나 연결 이름이 지정되는 경우 이 모드가 기본값입니다.

채널을 중지하려면 Enter를 누르십시오.

자세한 정보는 164 페이지의 『채널 중지 및 일시정지』의 내용을 참조하십시오. 중지된 채널 재시작에 대한 정보는 165 페이지의 『중지된 채널 재시작』의 내용을 참조하십시오.

참고: 공유 채널이 재시도 상태에 있으며 채널이 시작된 채널 시작기가 실행되지 않는 경우, 명령이 입력된 큐 관리자에서 채널에 대한 STOP 요청이 발행됩니다.

z/OS 채널 상태 표시

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 채널 상태를 표시할 수 있습니다.

MQSC 명령을 사용해서 채널 또는 채널 세트의 상태를 표시하려면 DISPLAY CHSTATUS를 사용하십시오.

참고: 많은 채널이 있는 경우 채널 상태 정보 표시는 약간의 시간이 걸릴 수 있습니다.

채널 목록 패널에서 조작 및 제어판을 사용하여(663 페이지의 그림 106 참조) 각 채널에 대한 채널 상태 요약이 다음과 같이 표시됩니다.

INACTIVE	활성인 연결이 없습니다.
상태	하나의 연결이 활성화되었습니다.
nnn status	두 개 이상의 연결이 현재 상태이며 모든 현재 연결이 동일한 상태를 가지고 있습니다.
nnn CURRENT	두 개 이상의 연결이 현재 상태이며 현재 연결이 모두 동일한 상태를 가지고 있지는 않습니다.
공백	IBM MQ가(예를 들어, 채널 시작기가 실행 중이 아니기 때문에) 활성인 연결 수를 판별할 수 없습니다. **참고:** 속성 지정이 GROUP인 채널 오브젝트의 경우에는 상태가 표시되지 않습니다.

여기서 *nnn*은 활성 연결의 수이고 *status*는 다음 중 하나입니다.

INIT	INITIALIZING
BIND	BINDING
START	STARTING
RUN	RUNNING
STOP	STOPPING 또는 STOPPED
RETRY	RETRYING
REQST	REQUESTING

채널 상태에 대한 자세한 정보를 표시하려면 목록 채널 또는 표시 또는 변경 채널 패널에서 상태 키(F11)를 눌러 목록 채널 - 현재 상태 패널을 표시하십시오(673 페이지의 그림 114 참조).

```
List Channels - Current Status - MQ25      Row 1 of 16

Type action codes, then press Enter. Press F11 to display saved status.
1=Display current status

Channel name     Connection name                  State
Start time      Messages Last message time  Type   Disposition
<> *                         CHANNEL   ALL    MQ25

_ RMA0.CIRCUIT.ACL.F  RMA1                         STOP
_  2005-03-21 10.22.36 557735  2005-03-24 09.51.11 SENDER  PRIVATE MQ25
_ RMA0.CIRCUIT.ACL.N  RMA1
_  2005-03-21 10.23.09 378675  2005-03-24 09.51.10 SENDER  PRIVATE MQ25
_ RMA0.CIRCUIT.CL.F  RMA2
_  2005-03-24 01.12.51 45544   2005-03-24 09.51.08 SENDER  PRIVATE MQ25
_ RMA0.CIRCUIT.CL.N  RMA2
_  2005-03-24 01.13.55 45560   2005-03-24 09.51.11 SENDER  PRIVATE MQ25
_ RMA1.CIRCUIT.CL.F  RMA1
_  2005-03-21 10.24.12 360757  2005-03-24 09.51.11 RECEIVER PRIVATE MQ25
_ RMA1.CIRCUIT.CL.N  RMA1
_  2005-03-21 10.23.40 302870  2005-03-24 09.51.09 RECEIVER PRIVATE MQ25
******** End of list ********
명령 == = > ________________________________________________________________
F1=Help   F2=Split   F3=Exit   F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd    F9=SwapNext F10=Messages F11=Saved  F12=Cancel
```

그림 *114.* 채널 연결 나열

상태 값은 다음과 같습니다.

INIT	INITIALIZING
BIND	BINDING
START	STARTING
RUN	RUNNING
STOP	STOPPING 또는 STOPPED
RETRY	RETRYING
REQST	REQUESTING
DOUBT	STOPPED 및 INDOUBT(YES)

자세한 정보는 156 페이지의 『채널 상태』의 내용을 참조하십시오.

F11 키를 눌러 저장된 상태와 함께 유사한 채널 연결 목록을 볼 수 있습니다. 현재 목록으로 다시 돌아가려면 F11 키를 누르십시오. 저장된 상태는 최소 하나의 메시지 배치가 채널에서 전송될 때까지 적용되지 않습니다.

조치 코드 1 또는 슬래시(/)를 사용하여 연결을 선택하고 Enter를 누르십시오. 채널 연결 현재 상태 표시 패널이 표시됩니다.

z/OS 클러스터 채널 표시

MQSC 명령을 사용하거나 조작 및 제어판을 사용하여 클러스터 채널을 표시할 수 있습니다.

정의된 모든 클러스터 채널(명시적으로 또는 자동 정의를 사용하여)을 표시하려면, MQSC 명령 DISPLAY CLUSQMGR을 사용하십시오.

조작 및 제어판을 사용하여 초기 패널부터 시작하여 다음 필드를 완료하고 Enter를 누르십시오.

필드	값
Action	1(목록 또는 표시)
오브젝트 유형	CLUSCHL
이름	*

각 클러스터 채널에 대한 정보가 세 줄을 차지하고 해당 채널, 클러스터 및 큐 관리자 이름을 포함하는 674 페이지의 그림 115 그림과 같은 패널이 표시됩니다. 클러스터-송신자 채널의 경우, 전반적인 상태가 표시됩니다.

```
List Cluster queue manager Channels - MQ25    Row 1 of 9

Type action codes, then press Enter. Press F11 to display connection status.
1=Display  5=Perform  6=Start  7=Stop

Channel name      Connection name                State
Type      Cluster name                    Suspended
Cluster queue manager name              Disposition
<>   *                        -    MQ25
_ TO.MQ90.T       HURSLEY.MACH90.COM(1590)
_  CLUSRCVR   VJH01T                     N
_   MQ90                        -    MQ25
_ TO.MQ95.T       HURSLEY.MACH95.COM(1595)             RUN
_  CLUSSDRA   VJH01T                     N
_   MQ95                        -    MQ25
_ TO.MQ96.T       HURSLEY.MACH96.COM(1596)             RUN
_  CLUSSDRB   VJH01T                     N
_   MQ96                        -    MQ25
******** End of list ********


명령 == = > _________________________________________________________________
F1=Help    F2=Split   F3=Exit   F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd     F9=SwapNext F10=Messages F11=Status  F12=Cancel
```

그림 *115.* 클러스터 채널 목록

하나 이상의 채널에 관한 전체 정보를 표시하려면 이름에 대해 조치 코드 1을 입력하고 Enter를 누르십시오. 조치 코드 5, 6 또는 7을 사용하여 기능(예: ping, 해결 및 재설정)을 수행하고, 클러스터 채널을 시작 또는 중지하십시오.

채널 상태에 대해 정보를 표시하려면 상태 키(F11)를 누르십시오.

z/OS *zEnterprise Data Compression Express* 기능을 사용하도록 *IBM MQ for z/OS* 준비

The zEnterprise® Data Compression (zEDC) Express facility is available for certain models of IBM Z® machines, starting from IBM zEC12 GA2, using a minimum z/OS level of z/OS 2.1.

자세한 정보는 zEnterprise 데이터 압축 (zEDC) 을 참조하십시오.

전제조건

IBM z15 이상의 경우, zEnterprise 데이터 압축 (zEDC) Express 기능이 하드웨어 시스템의 PCIe I/O 드로어에 있는 선택적 기능에서 zEDC용 Integrated Accelerator로 온칩으로 이동되었습니다. 이 변경으로 구성 전제조건이 업데이트되고 하드웨어 시스템에 따라 다릅니다.

IBM z15 이상

z/OS.

- z/OS 2.5: UJ00639
- z/OS 2.4: UJ00636
- z/OS 2.3: UJ00635
- z/OS 2.2: UJ00638

z15 이상 시스템에 대한 하드웨어 요구사항은 없습니다. 이 시스템에서 zEDC 솔루션을 위한 통합 가속기는 내장 데이터 가속 기능을 제공하므로 별도의 어댑터가 더 이상 필요하지 않습니다.

IBM zEC12 GA2 to IBM z14

또한 시스템에 다음과 같은 요구사항이 있어야 합니다.

- 하드웨어 시스템의 PCIe I/O 드로어에 설치된 zEDC Express ® 어댑터.
- IFAPRDxx parmlib 멤버에서 zEDC 소프트웨어 기능 (선택사항, 유료 기능) 이 사용 가능해야 합니다.

프로시저

IBM zEC12 GA2 to IBM z14

채널 시작기 사용자 ID가 RACF FACILITY CLASS의 FPZ.ACCELERATOR.COMPRESSION 프로파일 또는 엔터프라이즈가 사용하는 외부 보안 관리자(ESM)의 동등한 부분에 대한 READ 권한을 갖고 있는지 확인하십시오.

주의: IBM z15 이상에는 필요하지 않습니다.

IBM zEnterprise zEC12 GA2 이상

송신측 및 수신측 모두에서 COMPMSG(ZLIBFAST)를 갖는 채널을 구성하십시오. 구성된 후에는 zlib 압축이 채널을 통해 이동하는 메시지를 압축 및 압축 해제하는 데 사용됩니다.

압축할 데이터의 크기가 최소 임계값 이상인 경우 zEDC 에서 압축이 수행됩니다. 임계값은 사용 중인 IBM z 하드웨어에 따라 다릅니다.

- IBM zEC12 GA2 to IBM z14 has a minimum threshold of 4KB
- IBM z15 이상의 최소 임계값은 1KB 입니다.

임계값 미만의 메시지에 대해서는 소프트웨어에서 압축 또는 인플레이션이 수행됩니다.

z/OS z/OS에 대한 통신 설정

분산 큐잉 관리 채널이 시작되면 이 채널에서 채널 정의에 지정된 연결을 사용하려 합니다. 성공하려면 연결이 정의되고 사용 가능해야 합니다. 이 절에서는 연결을 정의하는 방법에 대해 설명합니다.

DQM은 IBM MQ의 리모트 큐잉 기능입니다. 통신 링크에 대한 인터페이스를 형성하는 큐 관리자에 대한 채널 제어 프로그램을 제공합니다. 이러한 링크는 시스템 운영자가 제어할 수 있습니다. 분산 큐잉 관리의 채널 정의에서 이들 연결을 사용합니다.

z/OS에 사용될 수 있는 통신 프로토콜의 두 양식 중 하나를 선택하십시오.

- 676 페이지의 『z/OS에서 TCP 연결 정의』
- 678 페이지의 『APPC/MVS를 사용하여 z/OS에 대한 LU 6.2 연결 정의』

각 채널 정의는 하나의 프로토콜만 전송 프로토콜(전송 유형) 속성으로 지정해야 합니다. 큐 관리자는 둘 이상의 프로토콜을 사용하여 통신할 수 있습니다.

예제 구성- IBM MQ for z/OS 을 참조하는 것이 도움이 될 수도 있습니다. 큐 공유 그룹을 사용하는 경우682 페이지의 『큐 공유 그룹을 사용하는 IBM MQ for z/OS에 대한 통신 설정』을 참조하십시오.

관련 개념

661 페이지의 『패널 및 명령 사용』
DQM을 관리하기 위해 MQSC 명령, PCF 명령 또는 조작 및 제어판을 사용할 수 있습니다.

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

597 페이지의 『IBM MQ for z/OS 설정』
이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

660 페이지의 『z/OS에서 채널 모니터링 및 제어』
DQM 명령과 패널을 사용하여 리모트 큐 관리자에 대한 채널을 작성, 모니터, 제어할 수 있습니다.

679 페이지의 『큐 공유 그룹으로 DQM을 위한 IBM MQ for z/OS 준비』
IBM MQ for z/OS에서 큐 공유 그룹을 갖는 분산 큐잉을 구성하려면 이 절의 지시사항을 사용하십시오.

682 페이지의 『큐 공유 그룹을 사용하는 IBM MQ for z/OS에 대한 통신 설정』

분산 큐잉 관리 채널이 시작되면, 채널 정의에서 지정된 연결을 사용하려고 합니다. 이 시도가 성공하려면 연결이 정의되고 사용 가능해야 합니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

z/OS *z/OS*에서 *TCP* 연결 정의

TCP 연결을 정의하려면 여러 설정을 구성해야 합니다.

TCP 시스템 매개변수 데이터 세트인 *tcpip*.TCPIP.DATA에서 TCP 주소 공간 이름이 지정되어야 합니다. 데이터 세트에 "TCPIPJOBNAME *TCPIP_proc*" 명령문이 포함되어야 합니다.

방화벽을 사용 중인 경우, 채널 시작기에서 채널의 주소로 및 원격 연결에서 큐 관리자로의 `allow` 연결을 구성해야 합니다.

일반적으로 방화벽에 대한 정의는 목적지 IP 주소 및 포트에 전송 IP 주소 및 포트를 구성합니다.

- z/OS 이미지는 둘 이상의 호스트 이름을 가질 수 있으며, 복수 호스트 주소를 소스 주소로 갖는 방화벽을 구성해야 할 수 있습니다.

  NETSTAT HOME 명령을 사용하여 이들 이름과 주소를 표시할 수 있습니다.
- 채널 시작기는 다른 포트에 복수 리스너를 가질 수 있으므로, 이들 포트를 구성해야 합니다.
- 큐 공유 그룹에 대해 공유 포트를 사용 중인 경우 공유 포트도 구성해야 합니다.

채널 시작기 주소 공간에 데이터 세트를 읽을 권한이 있어야 합니다. 사용 중인 TCP/IP 제품 및 인터페이스에 따라 다음 기술을 사용하여 TCPIP.DATA 데이터 세트에 액세스할 수 있습니다.

- 환경 변수 RESOLVER_CONFIG
- HFS 파일, /etc/resolv.conf
- //SYSTCPD DD문
- //SYSTCPDD DD문
- *jobname/userid*.TCPIP.DATA
- SYS1.TCPPARMS(TCPDATA)
- *zapname*.TCPIP.DATA

또한 TCP/IP에 대한 상위 레벨 규정자를 올바르게 지정하는 데 유의해야 합니다.

이름을 IP 주소로 변환 및 IP 주소를 이름으로 변환할 수 있는 적절하게 구성된 DNS(Domain Name System) 서버가 필요합니다.

참고: 분석기 구성에 대한 일부 변경은 그를 사용하는 애플리케이션(예: IBM MQ)의 재순환이 필요합니다.

자세한 정보는 다음 사항을 참고하십시오.

- 기본 TCP/IP 시스템
- z/OS UNIX 시스템 서비스.

시작 시 각 TCP 채널은 TCP 자원을 사용합니다. PROFILE.TCPIP 구성 데이터 세트에서 다음 매개변수를 조정해야 할 수 있습니다.

ACBPOOLSIZE
: 시작된 TCP 채널당 하나를 추가하고, 1을 더하십시오.

CCBPOOLSIZE
: 시작된 TCP 채널당 하나를 추가하고, DQM 디스패처당 1을 더하고, 1을 더하십시오.

DATABUFFERPOOLSIZE
: 시작된 TCP 채널당 둘을 추가하고, 1을 더하십시오.

MAXFILEPROC
: 채널 시작기의 각 디스패처가 핸들링할 수 있는 채널 수를 제어할 수 있습니다.

이 매개변수는 SYSI.PARMLIB의 BPXPRMxx 멤버에서 지정됩니다. 필요한 만큼 큰 값을 지정했는지 확인하십시오.

기본적으로 채널 시작기는 TCPNAME 큐 관리자 속성에 이름 지정된 스택과 연관된 IP 주소로만 바인딩할 수 있습니다. 채널 시작기가 시스템에서 추가 TCP/IP 스택을 사용하여 통신할 수 있도록 하려면 TCPSTACK 큐 관리자 속성을 MULTIPLE로 변경하십시오.

관련 개념

677 페이지의 『송신측』
TCP/IP 연결의 송신측에는 구성할 많은 설정이 있습니다.

677 페이지의 『TCP에서 수신』
TCP/IP 연결의 수신측에는 구성할 많은 설정이 있습니다.

677 페이지의 『TCP 리스너 백로그 옵션 사용』
TCP/IP에서 수신 시, 최대 미해결 연결 요청 수가 설정됩니다. 이런 미해결 요청은 리스너가 요청을 승인할 때까지 TCP/IP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

z/OS 송신측

TCP/IP 연결의 송신측에는 구성할 많은 설정이 있습니다.

채널 정의의 연결 이름(CONNAME) 필드는 대상의 호스트 이름(예: MVSHUR1) 또는 TCP 네트워크 주소로 설정되어야 합니다. TCP 네트워크 주소는 IPv4 점분리 십진수 양식(예: 127.0.0.1)이거나 IPv6 16진수 양식(예: 2001:DB8:0:0:0:0:0:0)일 수 있습니다. 연결 이름이 호스트 이름인 경우에는, 호스트 이름을 TCP 호스트 주소로 변환하기 위해 TCP 이름 서버가 필요합니다. (이 요구사항은 IBM MQ가 아니라 TCP의 기능입니다.)

연결을 시작하는 한쪽 끝(송신자, 요청자 및 서버 채널 유형)에서 연결에 대한 선택적 포트 번호를 제공할 수 있습니다. 예를 들어, 다음과 같습니다.

연결 이름
: 192.0.2.0(1555)

이 경우 시작하는 한쪽 끝은 포트 1555에서 대기하는 수신 프로그램에 연결하려고 시도합니다.

참고: 선택적 포트 번호가 지정되지 않으면 1414의 기본 포트 번호가 사용됩니다.

채널 시작기는 사용 가능한 모든 활성 TCP/IP 스택을 사용할 수 있습니다. 기본적으로, 채널 시작기는 아웃바운드 채널을 TCPNAME 큐 관리자 속성에서 이름 지정된 TCP/IP 스택에 대한 기본 IP 주소에 바인드합니다. 다른 스택을 통해 연결하려면 채널의 LOCLADDR 속성에서 스택의 호스트 이름이나 IP 주소 중 하나를 지정해야 합니다.

z/OS *TCP*에서 수신

TCP/IP 연결의 수신측에는 구성할 많은 설정이 있습니다.

수신 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다. 그렇게 하려면, 리스너 프로그램이 시작되어 수신되는 네트워크 요청을 감지하고 관련 채널을 시작해야 합니다. START LISTENER 명령을 사용하거나 조작 및 제어판을 사용하여 이 리스너 프로그램을 시작합니다.

기본적으로,

- TCP 리스너 프로그램은 포트 1414를 사용하고 TCP 스택에 사용 가능한 모든 주소에서 대기합니다.
- TCP/IP 리스너는 TCPNAME 큐 관리자 속성에서 이름 지정되는 TCP/IP 스택과 연관된 주소에만 바인드할 수 있습니다.

다른 주소 또는 모든 사용 가능한 TCP 스택에 대한 리스너를 시작하려면 TCPSTACK 큐 관리자 속성을 'MULTIPLE'로 설정하십시오.

START LISTENER 명령에서 IPADDR을 지정하여 특정 주소나 호스트 이름에서만 대기하기 위해 TCP 리스너 프로그램을 시작할 수 있습니다. 자세한 정보는 리스너의 내용을 참조하십시오.

z/OS *TCP* 리스너 백로그 옵션 사용

TCP/IP에서 수신 시, 최대 미해결 연결 요청 수가 설정됩니다. 이런 미해결 요청은 리스너가 요청을 승인할 때까지 TCP/IP 포트에서 대기 중인 요청의 백로그로 간주될 수 있습니다.

z/OS의 기본 리스너 백로그 값은 10000입니다. 백로그가 이 값에 도달하면 TCP/IP 연결은 거부되고 채널은 시작할 수 없습니다.

MCA 채널의 경우에는, 채널이 RETRY 상태가 되고 나중에 연결을 재시도하게 됩니다.

클라이언트 연결의 경우, 클라이언트가 MQCONN에서 MQRC_Q_MGR_NOT_AVAILABLE 이유 코드를 수신하며 나중에 연결을 재시도할 수 있습니다.

z/OS *APPC/MVS*를 사용하여 *z/OS*에 대한 *LU 6.2* 연결 정의

LU6.2 연결을 정의하려면 여러 설정을 구성해야 합니다.

APPC/MVS 설정

채널 시작기의 각 인스턴스에는 다음 예에서처럼 SYS1.PARMLIB의 APPCPMxx 멤버에서 APPC/MVS로 정의된 사용할 LU 이름이 필요합니다.

```
LUADD ACBNAME( luname ) NOSCHED TPDATA(CSQ.APPCTP)
```

*luname*은 사용될 논리 장치의 이름입니다. NOSCHED는 필수이며, TPDATA는 사용되지 않습니다. ASCHPMxx 멤버나 APPC/MVS TP 프로파일 데이터 세트에 어떠한 추가사항도 필요하지 않습니다.

부가 정보 데이터 세트가 DQM에서 사용된 연결을 정의하기 위해 확장되어야 합니다. APPC 유틸리티 프로그램 ATBSDFMU를 사용하여 이를 수행하는 방법에 대한 자세한 내용은 제공된 샘플 CSQ4SIDE를 참조하십시오. 사용할 TPNAME 값의 자세한 내용은 다음 테이블을 참조하십시오.

표 *50.* 리모트 큐 관리자 플랫폼에 대한 로컬 *z/OS* 시스템의 설정

원격 플랫폼	TPNAME
z/OS 또는 MVS	리모트 큐 관리자에 관한 해당 부가 정보와 동일한 TPNAME입니다.
IBM i	IBM i 시스템의 라우팅 입력 항목에 있는 비교 값과 동일합니다.
UNIX and Linux 시스템	리모트 큐 관리자에 관한 해당 부가 정보와 동일한 TPNAME입니다.
Windows	Windows 리스너 실행 명령 또는 Windows에서 TpSetup을 사용하여 정의된 호출 가능 트랜잭션 프로그램에 지정되었습니다.

동일한 시스템에 둘 이상의 큐 관리자가 있다면, 반드시 채널 정의에서 TPname이 고유한지 확인하십시오.

필요할 수 있는 VTAM 정의에 대한 정보도 *Multiplatform APPC Configuration Guide*를 참조하십시오.

큐 관리자가 동일한 또는 다른 z/OS 시스템의 큐 관리자와 APPC를 사용하여 통신하는 환경에서는, 통신 LU에 대한 VTAM 정의가 SECACPT(ALREADYV)를 지정하는지 또는 CONVSEC(ALREADYV)를 지정하는 LU 사이의 연결에 대한 RACF APPCLU 프로파일이 있는지 확인하십시오.

z/OS 명령 VARY ACTIVE가 인바운드 또는 아웃바운드 통신을 시작하기 전에 기본 및 리스너 LU 모두에 대해 실행되어야 합니다.

주의: APPC 설정뿐만 아니라, 다음 명령을 실행해야 합니다.

```
ALTER QMGR LUNAME(luname)
```

그리고 채널 시작기를 다시 시작하십시오.

자세한 정보는 LUNAME의 내용을 참조하십시오.

관련 개념

679 페이지의 『LU 6.2에 연결』
LU 6.2에 연결하기 위해서는 구성할 많은 설정이 있습니다.

679 페이지의 『LU 6.2에서 수신』
LU 6.2에서 수신하려면 구성할 많은 설정이 있습니다.

z/OS *LU 6.2*에 연결

LU 6.2에 연결하기 위해서는 구성할 많은 설정이 있습니다.

채널 정의의 연결 이름(CONNAME) 필드는 APPC/MVS에 대한 부가 정보 데이터 세트에서 지정된 대로 기호 목적지 이름으로 설정되어야 합니다.

사용할 LU 이름(이전에 설명한 대로 APPC/MVS로 정의됨)도 채널 시작기 매개변수에서 지정되어야 합니다. 리스너가 수신에 사용하는 동일한 LU로 설정되어야 합니다.

채널 시작기는 "SECURITY(SAME)" APPC/MVS 옵션을 사용하므로, 아웃바운드 전송에 사용되고 수신자에게 제시되는 채널 시작기 주소 공간의 사용자 ID입니다.

z/OS *LU 6.2*에서 수신

LU 6.2에서 수신하려면 구성할 많은 설정이 있습니다.

수신 MCA는 송신 채널로부터 들어온 시동 요청에 응답하여 시작됩니다. 그렇게 하려면, 리스너 프로그램이 시작되어 수신되는 네트워크 요청을 감지하고 관련 채널을 시작해야 합니다. 리스너 프로그램은 APPC/MVS 서버입니다. START LISTENER 명령으로 또는 조작 및 제어판을 사용하여 이를 시작합니다. 부가 정보 데이터 세트에서 정의된 기호 목적지 이름과 함께 사용할 LU 이름을 지정해야 합니다. 그렇게 식별된 로컬 LU는 채널 시작기 매개변수에서 설정된 대로 아웃바운드 전송에 사용되는 것과 동일해야 합니다.

z/OS 큐 공유 그룹으로 DQM을 위한 IBM MQ for z/OS 준비

IBM MQ for z/OS에서 큐 공유 그룹을 갖는 분산 큐잉을 구성하려면 이 절의 지시사항을 사용하십시오.

큐 공유 그룹을 사용한 구성 예에 대해서는 구성 예 - 큐 공유 그룹을 사용하는 IBM MQ for z/OS의 내용을 참조하십시오. 큐 공유 그룹을 사용한 메시지 채널 계획 예에 대해서는 큐 공유 그룹을 사용하는 z/OS의 메시지 채널 계획 예를 참조하십시오.

큐 공유 그룹을 포함하여 분산 큐잉을 사용할 수 있도록 다음 컴포넌트를 작성하고 구성해야 합니다.

- LU 6.2 및 TCP/IP 리스너
- 전송 큐 및 트리거링
- 메시지 채널 에이전트
- 동기화 큐

통신을 설정하는 데 필요한 구성요소를 작성한 후에는 682 페이지의 『큐 공유 그룹을 사용하는 IBM MQ for z/OS에 대한 통신 설정』의 내용을 참조하십시오.

큐 공유 그룹 사용 시 채널을 모니터하고 제어하는 방법에 대한 정보는 660 페이지의 『z/OS에서 채널 모니터링 및 제어』의 내용을 참조하십시오.

큐 공유 그룹 개념 및 이점은 다음 절을 참조하십시오.

서비스 클래스

공유 큐는 상이한 서비스 클래스를 제공하는 로컬 큐의 유형입니다. 공유 큐의 메시지는 결합 기능(CF)에 저장되며, 큐 공유 그룹의 모든 큐 관리자가 이 메시지를 액세스할 수 있습니다. 공유 큐의 메시지는 100MB 미만의 길이를 갖는 메시지여야 합니다.

일반 인터페이스

큐 공유 그룹에는 네트워크가 그룹을 단일 엔티티로 볼 수 있는 일반 인터페이스가 있습니다. 이 보기는 그룹 내의 임의의 큐 관리자에 연결하는 데 사용될 수 있는 단일 일반 주소를 가져서 달성됩니다.

큐 공유 그룹의 각 큐 관리자는 일반 주소와 논리적으로 관련된 주소의 인바운드 세션 요청을 인식합니다. 자세한 정보는 680 페이지의 『큐 공유 그룹에 대한 LU 6.2 및 TCP/IP 리스너』의 내용을 참조하십시오.

로드 밸런싱된 채널 시작

공유 전송 큐는 큐 공유 그룹의 임의의 채널 시작기에서 실행 중인 아웃바운드 채널에서 제공될 수 있습니다. 로드 밸런싱된 채널 시작이 채널 시작 명령의 목표로 지정된 위치를 판별합니다. 필요한 통신 서브시스템에 대한 액

세스 권한이 있는 적절한 채널 시작기가 선택됩니다. 예를 들어, TRPTYPE(LU6.2)으로 정의된 채널은 TCP/IP 서브시스템에 대한 액세스만 있는 채널 시작기에서 시작될 수 없습니다.

채널 시작기 선택은 채널 시작기의 헤드룸 및 채널 로드에 종속적입니다. 채널 로드는 채널 시작기 매개변수에서 정의된 대로 허용된 최대 활성 채널 수의 백분율로 표시되는 활성 채널 수입니다. 헤드룸은 활성 채널 수와 허용된 최대 수 사이의 차이입니다.

인바운드 공유 채널은 680 페이지의 『큐 공유 그룹에 대한 LU 6.2 및 TCP/IP 리스너』에 설명된 대로 일반 주소를 사용하여 큐 공유 그룹 간에 로드 밸런스를 조정할 수 있습니다.

공유 채널 복구

다음 테이블은 공유 채널 실패 유형 및 각 유형의 핸들링 방법을 표시합니다.

실패의 유형:	발생하는 상황:
채널 시작기 통신 서브시스템 실패	통신 서브시스템에 종속적인 채널은 채널 재시도를 입력하며 로드 밸런스 시작 명령으로 적절한 큐 공유 그룹 채널 시작기에서 다시 시작됩니다.
채널 시작기 실패	채널 시작기가 실패해도 연관된 큐 관리자는 활성 상태로 남습니다. 큐 관리자는 실패를 모니터하며 복구 처리를 시작합니다.
큐 관리자 실패	큐 관리자가 실패합니다(연관된 채널 시작기 실패). 큐 공유 그룹의 다른 큐 관리자가 이벤트를 모니터하고 피어 복구를 시작합니다.
공유 상태 실패	채널 상태 정보가 Db2에 저장되므로, 채널 상태 변경 발생 시 Db2에 대한 연결이 손실되면 실패가 발생합니다. 실행 중인 채널은 이 자원에 대한 액세스 없이 실행을 계속할 수 있습니다. Db2에 대한 액세스가 실패하면 채널이 재시도를 입력합니다.

공유 채널 상태 검색을 위해 복구를 관리하는 시스템에서 사용 가능하도록 실패한 시스템 대신 공유 채널 복구 처리가 Db2에 대한 연결을 요구합니다.

클라이언트 채널

클라이언트 연결 채널은 특정 큐 관리자에 연결되는 대신 일반 인터페이스에 연결되는 큐 공유 그룹의 메시지 고가용성을 활용할 수 있습니다. 자세한 정보는 클라이언트 연결 채널을 참조하십시오.

관련 개념

597 페이지의 『IBM MQ for z/OS 설정』
이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

657 페이지의 『다른 큐 관리자와의 통신 설정』
이 절에서는 분산 큐잉을 사용하기 시작하기 전에 작성해야 하는 IBM MQ for z/OS 준비를 설명합니다.

682 페이지의 『클러스터 및 큐 공유 그룹』
단일 정의로 클러스터가 공유 큐를 사용 가능하도록 할 수 있습니다. 그렇게 하려면 공유 큐를 정의할 때 클러스터의 이름을 지정합니다.

682 페이지의 『채널 및 직렬화』
공유 큐 피어 복구 동안 공유 큐의 메시지를 처리하는 메시지 채널 에이전트는 큐에 대한 액세스를 직렬화합니다.

관련 태스크

128 페이지의 『분산 큐잉 구성』
이 절은 큐 정의, 채널 정의, 트리거, 동기점 프로시저 등 IBM MQ 설치 간 상호통신에 대한 자세한 정보를 제공합니다.

관련 정보

공유 큐 및 큐 공유 그룹
그룹 내 큐잉

z/OS 큐 공유 그룹에 대한 *LU 6.2* 및 *TCP/IP* 리스너

그룹 LU 6.2 및 TCP/IP 리스너는 공통 주소와 논리적으로 연결된 주소에서 대기합니다.

LU 6.2 리스너의 경우 지정된 LUGROUP이 큐 공유 그룹과 연관된 VTAM 일반 자원에 맵핑됩니다.이 기술을 설정하는 예는 678 페이지의 『APPC/MVS를 사용하여 z/OS에 대한 LU 6.2 연결 정의』의 내용을 참조하십시오.

TCP/IP 리스너의 경우 지정된 포트가 다음 방법 중 하나로 일반 주소에 연결될 수 있습니다.

- IBM Network Dispatcher 같은 프론트 엔드 라우터의 경우, 인바운드 연결 요청이 라우터에서 큐 공유 그룹의 멤버로 전달됩니다.
- TCP/IP Sysplex Distributor의 경우, 분산 DVIPA로 설정된 특정 주소에서 실행 중이고 대기 중인 각 리스너에 수신 요청의 일부가 할당됩니다. 이 기술 설정 예는 Sysplex Distributor 사용을 참조하십시오.

z/OS 큐 공유 그룹에 대한 전송 큐 및 트리거

공유 전송 큐를 사용하면 큐 공유 그룹에서 목적지로 이동되기 전에 메시지를 저장할 수 있습니다.

이는 공유 큐이며, 큐 공유 그룹의 모든 큐 관리자가 접근할 수 있습니다.

트리거링

트리거된 공유 큐는 충족된 트리거 조건에 대해 하나 이상의 트리거 메시지를 생성할 수 있습니다. 트리거된 공유 큐와 연관된 큐 공유 그룹의 큐 관리자에서 정의된 각각의 로컬 이니시에이션 큐에 대해 하나의 트리거 메시지가 생성됩니다.

분산 큐잉의 경우 각 채널 시작기가 충족된 공유 전송 큐 트리거 조건에 대한 트리거 메시지를 수신합니다. 그러나 단 하나의 채널 시작기가 실제로 트리거된 시작을 처리하고, 다른 것은 안전하게 실패합니다. 그런 다음 트리거된 채널이 채널 `QSG.TO.QM2`를 시작하기 위해 트리거되는 로드 밸런싱된 시작(679 페이지의 『큐 공유 그룹으로 DQM을 위한 IBM MQ for z/OS 준비』 참조)으로 시작됩니다. 공유 전송 큐를 작성하려면 다음 예에 표시된 대로 IBM MQ 명령(MQSC)을 사용하십시오.

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') +
USAGE(XMITQ) QSGDISP(SHARED) +
CFSTRUCT(APPLICATION1) INITQ(SYSTEM.CHANNEL.INITQ) +
TRIGGER TRIGDATA(QSG.TO.QM2)
```

z/OS 큐 공유 그룹에 대한 메시지 채널 에이전트

채널이 해당 이름을 지닌 채널의 채널 정의에 액세스를 하는 경우에는 채널 시작기에서만 시작될 수 있습니다.

메시지 채널 에이전트는 메시지의 송신 및 수신을 제어하는 IBM MQ 프로그램입니다. 메시지 채널 에이전트는 한 큐 관리자에서 다른 큐 관리자로 메시지를 이동합니다. 채널의 각 끝에 하나의 메시지 채널 에이전트가 있습니다.

채널 정의를 큐 관리자 전용으로 정의하거나, 또는 공유 저장소에 저장함으로써 모든 위치(그룹 정의)에서 사용할 수도 있습니다. 이는 그룹 정의 채널을 큐 공유 그룹의 모든 채널 시작기에서 사용할 수 있음을 의미합니다.

참고: 그룹 정의의 개별 사본은 변경이나 삭제가 가능합니다.

그룹 채널 정의를 작성하려면 다음 예에 표시된 대로 IBM MQ 명령(MQSC)을 사용하십시오.

```
DEFINE CHL(QSG.TO.QM2) CHLTYPE(SDR) +
TRPTYPE(TCP) CONNAME(QM2.MACH.IBM.COM) +
XMITQ(QM2) QSGDISP(GROUP)
```

```
DEFINE CHL(QM2.TO.QSG) CHLTYPE(RCVR) TRPTYPE(TCP) +
QSGDISP(GROUP)
```

큐 공유 그룹을 사용한 분산 큐잉에 사용하는 메시지 채널 에이전트를 보는 관점에는 두 가지가 존재합니다.

인바운드

인바운드 채널이 그룹 리스너를 통해 큐 관리자에 연결되는 경우, 인바운드 채널은 공유 채널입니다. 일반 인터페이스를 통해 큐 공유 그룹에 연결된 후 그룹 내의 큐 관리자로 전달되거나 특정 큐 관리자의 그룹 포트 또는 그룹 리스너가 사용하는 luname을 대상으로 합니다.

아웃바운드

아웃바운드 채널이 공유 전송 큐로부터 메시지를 이동하는 경우, 아웃바운드 채널은 공유 채널입니다. 명령 예에서, 송신자 채널 `QSG.TO.QM2`는 전송 큐 QM2가 QSGDISP(SHARED)로 정의되기 때문에 공유 채널입니다.

z/OS 큐 공유 그룹을 위한 동기화 큐

공유 채널은 SYSTEM.QSG.CHANNEL.SYNCQ라는 자체 공유 동기화 큐를 갖습니다.

이 동기화 큐는 큐 공유 그룹의 모든 멤버에 액세스할 수 있습니다. (전용 채널은 계속해서 전용 동기화 큐를 사용합니다. 658 페이지의 『IBM MQ 오브젝트 정의』의 내용을 참조하십시오.) 이것은 채널이 통신 서브시스템, 채널 시작기 또는 큐 관리자의 장애 시에 큐 공유 그룹 내의 다른 큐 관리자 및 채널 시작기 인스턴스에서 재시작될 수 있음을 의미합니다. 추가 정보는 679 페이지의 『큐 공유 그룹으로 DQM을 위한 IBM MQ for z/OS 준비』의 내용을 참조하십시오.

큐 공유 그룹을 사용하는 DQM의 경우에는 이름이 SYSTEM.QSG.CHANNEL.SYNCQ인 공유 큐를 사용할 수 있어야 합니다. 이 큐를 사용할 수 있어야 그룹 리스너를 시작할 수 있습니다.

큐가 사용 불가능하기 때문에 그룹 리스너가 실패하는 경우, 채널 시작기를 재순환하지 않고 큐가 정의되고 리스너가 재시작될 수 있습니다. 비공유 채널은 영향을 받지 않습니다.

반드시 INDXTYPE(MSGID)을 사용하여 이들 큐를 정의하십시오. 이 정의는 큐의 메시지에 액세스할 수 있는 속도를 개선합니다.

z/OS 클러스터 및 큐 공유 그룹

단일 정의로 클러스터가 공유 큐를 사용 가능하도록 할 수 있습니다. 그렇게 하려면 공유 큐를 정의할 때 클러스터의 이름을 지정합니다.

네트워크의 사용자는 공유 큐를 큐 공유 그룹 내에서 각 큐 관리자가 호스팅하는 것으로 봅니다. (공유 큐는 큐 공유 그룹에 의해 호스팅되는 것으로 광고되지 않습니다.) 클라이언트는 큐 공유 그룹의 모든 멤버와 함께 세션을 시작하여 메시지를 동일한 공유 큐에 넣을 수 있습니다.

자세한 정보는 216 페이지의 『큐 관리자 클러스터 구성』의 내용을 참조하십시오.

z/OS 채널 및 직렬화

공유 큐 피어 복구 동안 공유 큐의 메시지를 처리하는 메시지 채널 에이전트는 큐에 대한 액세스를 직렬화합니다.

메시지 채널 에이전트가 하나 이상의 공유 큐에서 미확약 메시지를 처리하는 중에 큐 공유 그룹의 큐 관리자가 실패하면, 채널 및 연관된 채널 시작기가 종료되고 큐 관리자에 대해 공유 큐 피어 복구가 수행됩니다.

공유 큐 피어 복구가 비동기 활동이므로, 공유 큐 피어 복구가 완료되기 전에 피어 채널 복구시 큐 공유 그룹의 다른 부분에서 채널을 동시에 재시작하려고 할 수 있습니다. 이 이벤트가 발생하는 경우, 커미트된 메시지가 아직 복구되고 있는 메시지에 앞서 처리될 수 있습니다. 이러한 방식으로 메시지가 순서를 벗어나서 처리되지 않도록 하기 위해 공유 큐에서 메시지를 처리하는 메시지 채널 에이전트가 이 큐에 대한 액세스를 직렬화합니다.

공유 큐 피어 복구가 여전히 진행 중인 채널의 시작 시도에 실패할 수 있습니다. 복구가 진행 중임을 나타내는 오류 메시지가 발행되었고 채널이 재시도 상태에 놓였습니다. 큐 관리자 피어 복구가 완료되면, 다음에 재시도할 때 채널을 재시작할 수 있습니다.

같은 이유로, 채널의 RESOLVE, PING 또는 DELETE 시도에 실패할 수 있습니다.

z/OS 큐 공유 그룹을 사용하는 *IBM MQ for z/OS*에 대한 통신 설정

분산 큐잉 관리 채널이 시작되면, 채널 정의에서 지정된 연결을 사용하려고 합니다. 이 시도가 성공하려면 연결이 정의되고 사용 가능해야 합니다.

사용될 수 있는 통신 프로토콜의 두 형식 중 하나를 선택하십시오.

- TCP
- APPC/MVS를 통한 LU 6.2

구성 예 - 큐 공유 그룹을 사용하는 IBM MQ for z/OS의 내용을 참조하는 것이 유용함을 발견할 수 있습니다.

z/OS 큐 공유 그룹에 대한 *TCP* 연결 정의

큐 공유 그룹에 대한 TCP 연결을 정의하려면, 송신측 및 수신측에서 특정 속성을 구성해야 합니다.

TCP 설정에 대한 정보는 676 페이지의 『z/OS에서 TCP 연결 정의』의 내용을 참조하십시오.

송신측

큐 공유 그룹에 연결하기 위한 채널 정의의 연결 이름(CONNAME) 필드는 큐 공유 그룹의 일반 인터페이스로 설정되어야 합니다(큐 공유 그룹 참조). 자세한 내용은 Sysplex Distributor 사용을 참조하십시오.

큐 공유 그룹을 사용하는 TCP에서의 수신

수신 공유 채널 프로그램은 송신 채널의 시동 요청에 대한 응답으로 시작됩니다. 그렇게 하려면, 리스너가 시작되어 수신되는 네트워크 요청을 감지하고 관련 채널을 시작해야 합니다. START LISTENER 명령으로, 그룹의 인바운드 속성 지정을 사용하거나 조작 및 제어판을 사용하여 이 리스너 프로그램을 시작합니다.

큐 공유 그룹에 있는 모든 그룹 리스너는 같은 포트에서 대기해야 합니다. 단일 MVS 이미지에서 실행 중인 두 개 이상의 채널 시작기가 있는 경우, 가상 IP 주소를 정의하고 START LISTENER 명령에 IPADDR을 지정하여 특정 주소 또는 호스트 이름만 대기하도록 TCP 리스너 프로그램을 시작할 수 있습니다. (자세한 정보는 START LISTENER를 참조하십시오.)

z/OS *z/OS*에서 *LU 6.2* 연결 정의
큐 공유 그룹에 대한 LU 6.2 연결을 정의하려면, 송신측 및 수신측에서 특정 속성을 구성해야 합니다.

APPC/MVS 설정에 대한 정보는 z/OS의 통신 설정을 참조하십시오.

APPC/MVS(LU 6.2)로의 연결

큐 공유 그룹에 연결할 채널 정의의 연결 이름(CONNAME) 필드는 APPC/MVS에 대한 부가 정보 데이터 세트에서 지정된 대로 기호 목적지 이름으로 설정되어야 합니다. 이 기호 목적지에서 지정되는 참여자 LU는 일반 자원 이름이어야 합니다. 자세한 내용은 일반 자원을 사용하여 네트워크에 자신을 정의를 참조하십시오.

일반 인터페이스를 사용하는 LU 6.2에서의 수신

수신 공유 MCA는 송신 채널의 시동 요청에 대한 응답으로 시작됩니다. 그렇게 하려면, 그룹 프로그램 리스너가 시작되어 수신되는 네트워크 요청을 감지하고 관련 채널을 시작해야 합니다. 리스너 프로그램은 APPC/MVS 서버입니다. START LISTENER 명령으로, 인바운드 속성 지정 그룹을 사용하거나 조작 및 제어판을 사용하여 이 리스너 프로그램을 시작합니다. 부가 정보 데이터 세트에서 정의된 기호 목적지 이름을 사용할 LU 이름을 지정해야 합니다. 자세한 내용은 일반 자원을 사용하여 네트워크에 자신을 정의를 참조하십시오.

z/OS IMS와 함께 IBM MQ 사용

IBM MQ -IMS 어댑터 및 IBM MQ - IMS 브릿지는 IBM MQ가 IMS와 상호작용할 수 있도록 허용하는 두 컴포넌트입니다.

함께 작동하도록 IBM MQ 및 IMS를 구성하려면 다음 태스크를 완료해야 합니다.

- 684 페이지의 『IMS 어댑터 설정』
- 690 페이지의 『IMS 브릿지 설정』

관련 개념

691 페이지의 『CICS와 함께 IBM MQ 사용』
CICS와 함께 IBM MQ를 사용하려면 IBM MQ CICS 어댑터 및 IBM MQ CICS bridge 컴포넌트를 구성해야 합니다.

693 페이지의 『IMS에서 OTMA 엑시트 사용』
IBM MQ for z/OS와 함께 IMS OTMA(Open Transaction Manager Access) 엑시트를 사용하려는 경우 이 토픽을 사용하십시오.

관련 태스크

592 페이지의 『z/OS에서 큐 관리자 구성』
IBM MQ for z/OS에서 큐 관리자를 구성하려면 이들 지시사항을 사용하십시오.

관련 참조

691 페이지의 『Language Environment 또는 z/OS 호출 가능 서비스 업그레이드 및 서비스 적용』
수행해야 하는 조치는 CALLLIBS 또는 LINK 중 사용하는 항목과 SMP/E의 버전에 따라 달라집니다.

관련 정보

IBM MQ 및 IMS
IBM MQ for z/OS의 IMS 및 IMS 브릿지

z/OS IMS 어댑터 설정

IMS 내에서 IBM MQ를 사용하려면 IBM MQ - IMS 어댑터(일반적으로 IMS 어댑터로 부름)가 필요합니다.

이 주제에서는 IMS 어댑터가 IMS 서브시스템에서 사용 가능하도록 설정하는 방법을 설명합니다. IMS 서브시스템을 사용자 조정하는 데 익숙하지 않은 경우, *IBM Documentation*의 *IMS* 정보를 참조하십시오.

IMS 어댑터를 IMS 애플리케이션에 사용 가능하게 설정하려면 다음 단계를 수행하십시오.

1. IMS 외부 서브시스템 연결 기능(ESAF)을 사용하여 외부 서브시스템으로 IMS에 대해 IBM MQ를 정의하십시오.

   685 페이지의 『IMS에 IBM MQ 정의』를 참조하십시오.
2. LPA 또는 링크 목록에 없는 경우 IBM MQ에 연결하는 모든 종속된 리젼 및 IMS 제어 리젼에 대해 JCL의 JOBLIB 또는 STEPLIB 연결에 IBM MQ 로드 라이브러리 thlqual.SCSQAUTH를 포함하십시오. JOBLIB 또는 STEPLIB에 권한이 부여되지 않은 경우 IMS 모듈을 포함하는 라이브러리(보통 IMS RESLIB) 이후에 DFSESL 연결에도 포함하십시오.

   또한 thlqual.SCSQANLx를 포함하십시오(여기서 x는 언어 문자임).

   DFSESL이 존재하는 경우 SCSQAUTH 및 SCSQANLx가 연결에 포함되거나 LNKLIST에 추가되어야 합니다. JCL에서 STEPLIB 또는 JOBLIB 연결에 추가하는 것은 충분하지 않습니다.
3. thlqual.SCSQASMS에서 사용자 라이브러리로 IBM MQ 어셈블러 프로그램 CSQQDEFV를 복사하십시오.
4. 제공된 프로그램, CSQQDEFV는 기본적으로 MQM1의 IMS LIT(Language Interface Token)로 식별된 하나의 서브시스템 이름 CSQ1을 포함합니다. 테스트 및 설치 확인을 위해 이 이름을 유지할 수 있습니다.

   프로덕션 서브시스템의 경우, NAME=CSQ1을 사용자의 서브시스템 이름으로 변경하거나 CSQ1을 사용합니다. 필요에 따라 추가 서브시스템 정의를 추가할 수 있습니다. LIT에 대한 자세한 정보는 688 페이지의 『IMS 어댑터에 IBM MQ 큐 관리자 정의』을(를) 참조하십시오.
5. 프로그램을 어셈블 및 링크 편집하여 CSQQDEFV 로드 모듈을 생성하십시오. 어셈블리의 경우, SYSLIB 연결에 라이브러리 thlqual.SCSQMACS를 포함하십시오. 링크 편집 매개변수 RENT를 사용하십시오. 이는 thlqual.SCSQPROC(CSQ4DEFV)의 샘플 JCL에서 표시됩니다.
6. IBM MQ에 연결된 모든 종속된 리젼에 대해 JCL의 JOBLIB 또는 STEPLIB 연결에서 작성한 모듈 CSQQDEFV를 포함하는 사용자 라이브러리를 포함하십시오. SCSQAUTH는 기본 로드 모듈을 가지므로 SCSQAUTH 앞에 이 라이브러리를 넣으십시오. 이를 수행하지 않으면 IMS에서 사용자 3041 이상종료를 수신합니다.
7. IMS 어댑터가 예기치 않은 IBM MQ 오류를 감지하면 z/OS SNAP 덤프를 DD 이름 CSQSNAP로 발행하고 애플리케이션에 MQRC_UNEXPECTED_ERROR로 이유 코드를 발행합니다. CSQSNAP DD문이 IMS 종속된 리젼 JCL에 없는 경우 덤프는 생성되지 않습니다. 이 경우 JCL에서 CSQSNAP DD 문을 포함하고 애플리케이션을 다시 실행할 수 있습니다. 그러나 일부 문제점은 일시적이므로 실패가 발생한 시점에서 실패 이유를 캡처하도록 CSQSNAP DD 문을 포함하는 것이 좋습니다.
8. 동적 IBM MQ 호출(IBM MQ 스텁을 동적으로 호출에 설명됨)을 사용하려면 685 페이지의 그림 116에 표시된 대로 동적 스텁을 빌드하십시오.
9. IMS 트리거 모니터를 사용하려는 경우 IMS 트리거 모니터 애플리케이션 CSQQTRMN을 정의하고 PSBGEN 및 ACBGEN을 수행하십시오. 689 페이지의 『IMS트리거 모니터 설정』를 참조하십시오.
10. RACF을(를) 사용하여 OPERCMDS 클래스에서 자원을 보호하는 경우, IBM MQ 큐 관리자 주소 공간과 연관된 사용자 ID가 연결될 수 있는 IMS 시스템에 MODIFY 명령을 실행할 권한이 있는지 확인하십시오.

```
//DYNSTUB EXEC PGM=IEWL,PARM='RENT,REUS,MAP,XREF'
//SYSPRINT DD  SYSOUT=*
//ACSQMOD  DD  DISP=SHR,DSN=thlqual.SCSQLOAD
//IMSLIB   DD  DISP=SHR,DSN=ims.reslib
//SYSLMOD  DD  DISP=SHR,DSN=private.load[1]
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,1)
//SYSLIN   DD  *
  INCLUDE ACSQMOD(CSQQSTUB)
  INCLUDE IMSLIB(DFSLI000)
  ALIAS MQCONN,MQCONNX,MQDISC             MQI entry points
  ALIAS MQGET,MQPUT,MQPUT1                MQI entry points
  ALIAS MQOPEN,MQCLOSE                    MQI entry points
  ALIAS MQBACK,MQCMIT                     MQI entry points
  ALIAS CSQBBAK,CSQBCMT                   MQI entry points
  ALIAS MQINQ,MQSET                       MQI entry points
  ALIAS DFSPLI,PLITDLI                    IMS entry points
  ALIAS DFSCOBOL,CBLTDLI                  IMS entry points
  ALIAS DFSFOR,FORTDLI                    IMS entry points
  ALIAS DFSASM,ASMTDLI                    IMS entry points
  ALIAS DFSPASCL,PASTDLI                  IMS entry points
  ALIAS DFHEI01,DFHEI1                    IMS entry points
  ALIAS DFSAIBLI,AIBTDLI                  IMS entry points
  ALIAS DFSESS,DSNWLI,DSNHLI              IMS entry points
  ALIAS MQCRTMH,MQDLTMH,MQDLTMP           IMS entry points
  ALIAS MQINQMP,MQSETMP,MQMHBUF,MQBUFMH   IMS entry points
  MODE AMODE(31),RMODE(24)                Note RMODE setting
  NAME CSQQDYNS(R)
/*

[1]Specify the name of a library accessible to IMS applications that
want to make dynamic calls to IBM MQ.
```

그림 *116.* 동적 호출 스텁을 링크-편집하는 샘플 *JCL*

관련 개념

690 페이지의 『IMS 브릿지 설정』

IBM MQ - IMS 브릿지는 IBM MQ가 IBM MQ 가능하지 않은 기존 프로그램 및 트랜잭션으로 입출력할 수 있게 하는 선택적 컴포넌트입니다.

관련 정보

IBM MQ 및 IMS

IBM MQ for z/OS의 IMS 및 IMS 브릿지

z/OS *IMS*에 *IBM MQ* 정의

IBM MQ가 IMS 제어 영역 및 해당 IBM MQ 큐 관리자에 액세스하는 각 종속 영역에 정의되어야 합니다. 이를 수행하려면 IMS.PROCLIB 라이브러리에 서브시스템 멤버(SSM)를 작성하고 SSM을 적용 가능한 IMS 영역에 식별해야 합니다.

IMS.PROCLIB에 서브시스템 멤버 항목을 배치합니다.

IMS.PROCLIB의 각 SSM 입력 항목은 IMS 영역에서 다른 큐 관리자로의 연결을 정의합니다.

SSM의 이름을 지정하려면 IMS IMSCTRL 매크로의 IMSID 필드의 값(1 - 4자의 영숫자 문자)를 사용자의 사이트에서 정의하는 임의의 이름(1 - 4자의 영숫자 문자)과 연결하십시오.

하나의 SSM이 모든 IMS 영역에 의해 공유될 수 있거나, 특정 멤버가 각 영역에 대해 정의될 수 있습니다. 외부 서브시스템에 대한 연결이 있으므로 이 멤버에는 많은 입력 항목이 들어 있습니다. 각 입력 항목은 80자 레코드입니다.

위치지정 매개변수

이 입력 항목의 필드는 다음과 같습니다.

```
SSN,LIT,ESMT,RTT,REO,CRC
```

설명:

주민등록번호
IBM MQ 큐 관리자 이름을 지정합니다. 필수이며 1 - 4자를 포함해야 합니다.

LIT
IMS에 제공된 언어 인터페이스 토큰(LIT)을 지정합니다. 이 필드는 필수이며, 그의 값이 CSQQDEFV 모듈의 하나와 일치해야 합니다.

ESMT
외부 서브시스템 모듈 테이블(ESMT)을 지정합니다. 이 테이블은 IMS가 로드해야 하는 첨부 모듈을 지정합니다. CSQQESMT가 이 필드의 필수 값입니다.

RTT
이 옵션은 IBM MQ에서 지원되지 않습니다.

REO
IMS 애플리케이션이 동작하지 않는 외부 서브시스템을 참조하는 경우 또는 자원이 스레드 작성 시에 사용 불가능한 경우에 사용될 영역 오류 옵션(REO)을 지정합니다. 이 필드는 선택사항이며 다음일 수 있는 단일 문자를 포함합니다.

R
IBM MQ 서비스에 대한 요청이 실패했음을 표시하는 리턴 코드를 애플리케이션으로 전달합니다.

Q
이상종료 코드 U3051로 애플리케이션을 종료하고, 활동을 마지막 커미트 지점으로 백아웃하고, 트랜잭션의 PSTOP을 수행하고, 입력 메시지를 다시 큐에 넣습니다. 이 옵션은 IMS 애플리케이션이 동작하지 않는 외부 서브시스템을 참조하려고 시도할 때 또는 자원이 스레드 작성 시에 사용 불가능한 경우에만 적용됩니다.

IBM MQ이 요청을 처리 중인 동안, 즉, 어댑터가 요청을 IBM MQ로 전달한 후 IBM MQ 문제점이 발생하는 경우 IBM MQ 완료 및 이유 코드가 애플리케이션으로 리턴됩니다.

A
이상종료 코드 U3047과 함께 애플리케이션을 종료하고 입력 메시지를 제거합니다. 이 옵션은 IMS 애플리케이션이 동작하지 않는 외부 서브시스템을 참조할 때 또는 자원이 스레드 작성 시에 사용 불가능한 경우에만 적용됩니다.

IBM MQ이 요청을 처리 중인 동안, 즉, 어댑터가 요청을 IBM MQ로 전달한 후 IBM MQ 문제점이 발생하는 경우 IBM MQ 완료 및 이유 코드가 애플리케이션으로 리턴됩니다.

CRC
이 옵션은 지정될 수 있지만 IBM MQ에서 사용되지 않습니다.

참고: 모든 위치 매개변수에 대한 전체 세부사항은 외부 서브시스템을 IMS에 지정하는 방법을 참조하십시오.

SSM 입력 항목 예는 다음과 같습니다.

```
CSQ1,MQM1,CSQQESMT,,R,
```

설명:

CSQ1	IBM MQ과 함께 제공되는 기본 서브시스템 이름입니다. 사용자 설치에 맞게 이것을 변경할 수 있습니다.
MQM1	CSQQDEFV에서 제공되는 기본 LIT입니다.
CSQQESMT	외부 서브시스템 모듈 이름입니다. 이 값을 사용해야 합니다.

R	REO 옵션.

키워드 매개변수

키워드 형식으로 IBM MQ 매개변수를 지정할 수 있습니다. SST 매개변수의 값은 DB2 또는 MQ입니다. MQ 값에 대한 지원은 IMS 14에 추가되었습니다. MQ를 사용하면 명확성이 높아지고 IMS 서브시스템 명령에 SST 값이 포함되지만, 그렇지 않으면 큰 효과가 없습니다. 필요한 경우 여전히 DB2 값을 사용할 수 있습니다. 기타 매개변수는 위치 지정 매개변수에 설명된 대로이며 다음 예에 표시됩니다.

```
SST=MQ,SSN=SYS3,LIT=MQM3,ESMT=CSQQESMT
```

설명:

SYS3	서브시스템 이름
MQM3	CSQQDEFV에서 제공되는 LIT
CSQQESMT	외부 서브시스템 모듈 이름

SSM EXEC 매개변수 지정

IMS 제어 영역의 시동 프로시저에서 SSM EXEC 매개변수를 지정하십시오. 이 매개변수는 1 - 4자의 서브시스템 멤버 이름(SSM)을 지정합니다.

IMS 제어 영역에 대한 SSM을 지정하는 경우, 제어 영역 아래에서 실행하는 모든 종속 영역은 SSM 매개변수에 의해 지정되는 IMS.PROCLIB 멤버에서 이름 지정되는 IBM MQ 큐 관리자에 접속할 수 있습니다. IMS.PROCLIB 멤버 이름은 SSM EXEC 매개변수에 지정된 1-4개의 문자로 연결된 IMS ID(IMSID= *xxxx*)입니다. IMS ID는 IMSCTRL 생성 매크로의 IMSID 매개변수입니다.

IMS에서는 필요한 만큼 많은 외부 서브시스템 연결을 정의합니다. 둘 이상의 연결이 서로 다른 IBM MQ 큐 관리자에 대해 정의될 수 있습니다. 모든 IBM MQ 연결은 동일한 z/OS 시스템 안에 있어야 합니다. 종속 영역의 경우 종속 영역 SSM을 지정하거나 제어 영역에 대해 지정된 것을 사용할 수 있습니다. 종속 영역 SSM 및 제어 영역 SSM에서 상이한 영역 오류 옵션(REO)을 지정할 수 있습니다. 687 페이지의 표 51에서는 SSM 스펙의 서로 다른 가능성을 보여줍니다.

표 *51. SSM* 스펙 옵션

제어 영역에 대한 SSM	종속 영역에 대한 SSM	Action	주석
아니오	아니오	없음	연결할 수 있는 외부 서브시스템이 없습니다.
아니오	예	없음	연결할 수 있는 외부 서브시스템이 없습니다.
예	아니오	제어 영역 SSM 사용	영역에서 스케줄된 애플리케이션은 제어 영역 SSM에서 식별되는 외부 서브시스템에 액세스할 수 있습니다. 각 첨부에 대한 엑시트 및 제어 블록은 제어 영역 및 종속 영역 주소 공간에 로드됩니다.
예	예(비어 있음)	종속 영역에 SSM이 사용되지 않음	이 영역에서 스케줄된 애플리케이션은 DL/I 데이터베이스에만 액세스할 수 있습니다. 각 첨부에 대한 엑시트 및 제어 블록은 제어 영역 주소 공간에 로드됩니다.
예	예(비어있지 않음)	제어 영역 SSM으로 종속 영역 SSM 검사	이 영역에서 스케줄된 애플리케이션은 두 SSM 모두에서 식별되는 외부 서브시스템에만 액세스할 수 있습니다. 각 첨부에 대한 엑시트 및 제어 블록은 제어 영역 및 종속 영역 주소 공간에 로드됩니다.

SSM 스펙 가능성의 최대 수를 제어할 특정 매개변수가 없습니다.

IMS 어댑터 사전로드

IMS 어댑터의 성능은 IMS에 의해 사전로드되는 경우 개선될 수 있습니다. 사전로드는 IMS.PROCLIB의 DFSMPLxx 멤버에 의해 제어됩니다. 자세한 정보는 "IMS Administration Guide: System"을 참조하십시오. 지정할 IBM MQ 모듈 이름은 다음과 같습니다.

CSQACLST	CSQAMLST	CSQAPRH	CSQAVICM	CSQFSALM	CSQQDEFV
CSQQCONN	CSQQDISC	CSQQTERM	CSQQINIT	CSQQBACK	CSQQCMMT
CSQQESMT	CSQQPREP	CSQQTTHD	CSQQWAIT	CSQQNORM	CSQQSSOF
CSQQSSON	CSQFSTAB	CSQQRESV	CSQQSNOP	CSQQCMND	CSQQCVER
CSQQTMID	CSQQTRGI	CSQQCON2	CSQBPAPI	CSQBCRMH	CSQBAPPL

IBM MQ classes for JMS의 사용에 대한 자세한 정보는 IMS에서 IBM MQ classes for JMS 사용을 참조하십시오.

IMS의 현재 릴리스는 MPP, BMP, IFP, JMP 및 JBP 영역에만 있는 PDS-E 형식 라이브러리에서 IBM MQ 모듈 사전로드를 지원합니다. IMS 영역의 다른 모든 유형은 PDS-E 라이브러리에서의 사전로드를 지원하지 않습니다. 임의의 다른 영역 유형에 대해 사전로드가 필요한 경우, 제공되는 IBM MQ 모듈이 PDS 형식 라이브러리로 복사되어야 합니다.

z/OS *IMS* 어댑터에 *IBM MQ* 큐 관리자 정의

IBM MQ 큐 관리자의 이름 및 대응하는 언어 인터페이스 토큰(LIT)이 큐 관리자 정의 테이블에서 정의되어야 합니다.

제공된 CSQQDEFX 매크로를 사용하여 CSQQDEFV 로드 모듈을 작성합니다. 688 페이지의 그림 117은(는) 이 어셈블러 매크로의 구문을 표시합니다.

```
CSQQDEFX TYPE=ENTRY|DEFAULT,NAME=qmgr-name,LIT=token
or
CSQQDEFX TYPE=END
```

그림 *117. CSQQDEFX* 매크로 구문

매개변수

TYPE=ENTRY|DEFAULT
다음과 같이 TYPE=ENTRY 또는 TYPE=DEFAULT를 지정하십시오.

TYPE=ENTRY
IMS 애플리케이션에 사용 가능한 IBM MQ 큐 관리자를 설명하는 테이블 입력 항목이 생성되도록 지정합니다. 이것이 첫 번째 입력 항목인 경우, CSQQDEFV CSECT 명령문을 포함한 테이블 헤더도 생성됩니다.

TYPE=DEFAULT
TYPE=ENTRY의 경우와 같습니다. 지정된 큐 관리자가 MQCONN 또는 MQCONNX가 모두 공백인 이름을 지정할 때 사용될 기본 큐 관리자입니다. 테이블에 그런 입력 항목이 하나만 있어야 합니다.

NAME=*qmgr-name*
MQCONN 또는 **MQCONNX**로 지정되는 대로 큐 관리자의 이름을 지정합니다.

LIT= token
IMS가 큐 관리자를 식별하기 위해 사용하는 언어 인터페이스 토큰(LIT)의 이름을 지정합니다.

MQCONN 또는 MQCONNX 호출은 *name* 입력 매개변수 및 *hconn* 출력 매개변수를 이름 레이블에, 따라서 CSQQDEFV 입력 항목의 LIT에 연관시킵니다. *hconn* 매개변수를 전달하는 추가 IBM MQ 호출은 MQCONN 또는 MQCONNX 호출에서 식별되는 CSQQDEFV 입력 항목의 LIT를 사용하여 해당하는 동일 LIT를 갖는 IMS SSM PROCLIB 멤버에서 정의되는 IBM MQ 큐 관리자로 전달합니다.

요약하면, MQCONN 또는 MQCONNX 호출의 **name** 매개변수가 CSQQDEFV의 CSQQDEFV를 식별하며 SSM 멤버의 동일한 CSQQDEFV가 IBM MQ 큐 관리자를 식별합니다. (MQCONN 호출에 대한 정보는 MQCONN - 연결 큐 매니저 참조). MQCONNX 호출에 대한 정보는 MQCONNX - 연결 큐 관리자(확장)를 참조하십시오.)

TYPE=END
테이블이 완료되었음을 지정합니다. 이 매개변수가 생략되면 TYPE=ENTRY가 가정됩니다.

CSQQDEFX 매크로 사용

689 페이지의 그림 118은(는) 큐 관리자 정의 테이블의 일반 레이아웃을 표시합니다.

```
CSQQDEFX NAME=subsystem1,LIT=token1
CSQQDEFX NAME=subsystem2,LIT=token2,TYPE=DEFAULT
CSQQDEFX NAME=subsystem3,LIT=token3
...
CSQQDEFX NAME=subsystemN,LIT=tokenN
CSQQDEFX TYPE=END
END
```

그림 *118.* 큐 관리자 정의 테이블의 레이아웃

z/OS *IMS*트리거 모니터 설정

IBM MQ 이니시에이션 큐를 모니터할 IMS 배치 지향 프로그램을 설정할 수 있습니다.

thlqual.SCSQPROC 라이브러리에서 모델 CSQQTAPL을 사용하여 IMS에 애플리케이션을 정의하십시오 (CSQQTRMN에 대한 트랜잭션 정의 예 참조).

thlqual.SCSQPROC 라이브러리에서 모델 CSQQTPSB를 사용하여 PSB 및 ACB를 생성하십시오(CSQQTRMN에 대한 PSB 정의 예 참조).

```
*  This is the application definition *
*  for the IMS Trigger Monitor BMP    *

APPLCTN PSB=CSQQTRMN,
PGMTYPE=BATCH,
SCHDTYP=PARALLEL
```

그림 *119. CSQQTRMN*에 대한 트랜잭션 정의 예

```
PCB  TYPE=TP,       ALTPCB for transaction messages
MODIFY=YES,         To "triggered" IMS transaction
PCBNAME=CSQQTRMN
PCB  TYPE=TP,       ALTPCB for diagnostic messages
MODIFY=YES,         To LTERM specified or "MASTER"
PCBNAME=CSQQTRMG,
EXPRESS=YES
PSBGEN LANG=ASSEM,
PSBNAME=CSQQTRMN,   Runs program CSQQTRMN
CMPAT=YES
```

그림 *120. CSQQTRMN*에 대한 *PSB* 정의 예

IMS 트리거 모니터 시작 및 중지에 대한 추가 정보는 IMS 트리거 모니터 제어를 참조하십시오.

z/OS IMS 브릿지 설정

IBM MQ - IMS 브릿지는 IBM MQ가 IBM MQ 가능하지 않은 기존 프로그램 및 트랜잭션으로 입출력할 수 있게 하는 선택적 컴포넌트입니다.

이 토픽은 IBM MQ - IMS 브릿지를 사용자 정의하기 위해 수행해야 하는 사항을 설명합니다.

IBM MQ에 대한 XCF 및 OTMA 매개변수를 정의하십시오.
이 단계는 IBM MQ 시스템에 대한 XCF 그룹 및 멤버 이름과 기타 OTMA 매개변수를 정의합니다. IBM MQ 및 IMS는 동일한 XCF 그룹에 속해야 합니다. 시스템 매개변수 로드 모듈에서 이러한 매개변수를 조정하려면 CSQ6SYSP 매크로의 OTMACON 키워드를 사용하십시오.

자세한 정보는 CSQ6SYSP 사용을 참조하십시오.

IMS에 XCF 및 OTMA 매개변수를 정의하십시오.
이 단계는 IMS 시스템을 위한 XCF 그룹 및 멤버 이름을 정의합니다. IMS 및 IBM MQ는 동일한 XCF 그룹에 속해야 합니다.

JCL 또는 IMS PROCLIB의 DFSPBxxx 멤버에 있는 IMS 매개변수 목록에 다음 매개변수를 추가하십시오.

OTMA=Y
이것은 IMS가 시작될 때 자동으로 OTMA를 시작합니다. (선택사항입니다. OTMA=N을 지정하는 경우 IMS 명령 /START OTMA를 실행하여 OTMA도 시작할 수 있습니다.)

GRNAME=
이 매개변수는 XCF 그룹 이름을 제공합니다.

스토리지 클래스 정의(다음 단계 참조) 및 CSQ6SYSP 매크로에서 OTMACON 키워드의 **Group** 매개변수에 지정된 그룹 이름과 동일합니다.

OTMANM=
이 매개변수는 IMS 시스템의 XCF 멤버 이름을 제공합니다.

이는 스토리지 클래스 정의에 지정된 멤버 이름과 동일합니다(다음 단계 참조).

IBM MQ에 IMS 시스템의 XCF 그룹 및 멤버 이름을 지시하십시오.
이것은 큐의 스토리지 클래스에서 지정됩니다. IBM MQ - IMS 브릿지를 통해 메시지를 보내려는 경우, 큐에 대한 스토리지 클래스를 정의할 때 이것을 지정해야 합니다. 스토리지 클래스에서 대상 IMS 시스템의 XCF 그룹 및 멤버 이름을 정의해야 합니다. 이를 수행하려면 IBM MQ 운영 및 제어판을 사용하거나 프로그램 가능 명령 형식 소개에서 설명하는 대로 IBM MQ 명령을 사용하십시오.

필요한 보안을 설정하십시오.
/SECURE OTMA IMS 명령은 OTMA를 통해 IMS 에 연결하는 **모든** IBM MQ 큐 관리자에 적용할 보안 레벨을 결정합니다. 자세한 정보는 IMS에서 IBM MQ 사용에 대한 보안 고려사항을 참조하십시오.

동일한 큐 관리자에 대한 추가 IMS 연결 추가

동일한 큐 관리자에 대한 IMS 연결을 추가하려면 다음을 수행해야 합니다.

- 새 IMS를 가리키도록 보조 스토리지 클래스 STGCLASS를 정의하십시오. 자세한 정보는 DEFINE STGCLASS를 참조하십시오.
- 보조 스토리지 클래스를 가리키도록 새 로컬 큐를 추가하십시오.

중요사항:

- 하나의 로컬 큐는 두 스토리지 클래스를 가리킬 수 없습니다.
- 하나의 스토리지 클래스는 두 IMS 브릿지를 가리킬 수 없습니다.
- IBM MQ 및 IMS는 동일한 XCF 그룹에 속해야 합니다. 시스템 매개변수 로드 모듈에서 이러한 매개변수를 조정하려면 CSQ6SYSP 매크로의 OTMACON 키워드를 사용하십시오.

  자세한 정보는 CSQ6SYSP 사용을 참조하십시오.

관련 개념

684 페이지의 『IMS 어댑터 설정』
IMS 내에서 IBM MQ를 사용하려면 IBM MQ - IMS 어댑터(일반적으로 IMS 어댑터로 부름)가 필요합니다.

관련 정보

IBM MQ 및 IMS

IBM MQ for z/OS의 IMS 및 IMS 브릿지

z/OS CICS와 함께 IBM MQ 사용

CICS와 함께 IBM MQ를 사용하려면 IBM MQ CICS 어댑터 및 IBM MQ CICS bridge 컴포넌트를 구성해야 합니다.

IBM MQ CICS 어댑터 및 IBM MQ CICS bridge 구성요소 구성에 대한 자세한 정보는 CICS 문서의 MQ 에 대한 연결 구성 섹션을 참조하십시오.

관련 개념

683 페이지의 『IMS와 함께 IBM MQ 사용』

IBM MQ -IMS 어댑터 및 IBM MQ - IMS 브릿지는 IBM MQ가 IMS와 상호작용할 수 있도록 허용하는 두 컴포넌트입니다.

관련 참조

691 페이지의 『Language Environment 또는 z/OS 호출 가능 서비스 업그레이드 및 서비스 적용』

수행해야 하는 조치는 CALLLIBS 또는 LINK 중 사용하는 항목과 SMP/E의 버전에 따라 달라집니다.

관련 정보

IBM MQ 및 CICS

z/OS Language Environment 또는 z/OS 호출 가능 서비스 업그레이드 및 서비스 적용

수행해야 하는 조치는 CALLLIBS 또는 LINK 중 사용하는 항목과 SMP/E의 버전에 따라 달라집니다.

다음 표는 다음 제품의 레벨을 업그레이드하거나 서비스를 적용하는 경우 IBM MQ for z/OS에 수행해야 하는 사항을 보여줍니다.

- Language Environment
- z/OS 호출 가능 서비스(예: APPC 및 RRS)

표 *52.* 서비스가 적용되거나 제품이 새 릴리스로 업그레이드됨

제품	CALLLIBS를 사용하고 SMP/E V3r2 이상인 경우 조치 **참고: Language Environment 및 호출 가능 서비스에 대해 별도의 작업을 실행할 필요가 없습니다. 한 작업이 충분합니다.**	LINK 사용 시 조치
Language Environment	1. SMP/E 작업의 경계를 대상 영역으로 설정하십시오. 2. SMPCNTL 카드에서 LINK LMODS CALLLIBS를 지정하십시오. 또한 CHECK, RETRY(YES) 및 RC와 같은 기타 매개변수를 지정할 수도 있습니다. 추가 정보는 *z/OS*의 *SMP/E:* 명령을 참조하십시오. 3. SMP/E 작업을 실행하십시오.	SMP/E 영역이 자동 다시 링크에 대해 설정되어 있고 CSQ8SLDQ 작업이 실행되는 경우 조치는 필요하지 않습니다.

표 52. 서비스가 적용되거나 제품이 새 릴리스로 업그레이드됨 *(계속)*		
제품	**CALLLIBS를 사용하고 SMP/E V3r2 이상인 경우 조치** **참고: Language Environment 및 호출 가능 서비스에 대해 별도의 작업을 실행할 필요가 없습니다. 한 작업이 충분합니다.**	**LINK 사용 시 조치**
호출 가능 서비스	1. SMP/E 작업의 경계를 대상 영역으로 설정하십시오. 2. SMPCNTL 카드에서 LINK LMODS CALLLIBS를 지정하십시오. 또한 CHECK, RETRY(YES) 및 RC와 같은 기타 매개변수를 지정할 수도 있습니다. 추가 정보는 *z/OS*의 *SMP/E:* 명령을 참조하십시오. 3. SMP/E 작업을 실행하십시오.	SMP/E 영역이 자동 다시 링크에 대해 설정되어 있고 CSQ8SLDQ 작업이 실행되는 경우 조치는 필요하지 않습니다.

표 53. 새 *SMP/E* 환경 및 라이브러리에서 제품 중 하나를 새 릴리스로 업데이트함		
제품	**CALLLIBS를 사용하고 SMP/E V3r2 이상인 경우 조치** **참고: Language Environment 및 호출 가능 서비스에 대해 세 개의 별도 작업을 실행할 필요가 없습니다. 두 제품에 대해 하나의 작업으로 충분합니다.**	**LINK 사용 시 조치**
Language Environment	1. 새 라이브러리를 가리키도록 SCEELKED 및 SCEESPC에 대한 DDDEF를 변경하십시오. 2. SMP/E 작업의 경계를 대상 영역으로 설정하십시오. 3. SMPCNTL 카드에서 LINK LMODS CALLLIBS를 지정하십시오. 또한 CHECK, RETRY(YES) 및 RC와 같은 기타 매개변수를 지정할 수도 있습니다. 추가 정보는 *z/OS*의 *SMP/E:* 명령을 참조하십시오. 4. SMP/E 작업을 실행하십시오.	1. IBM MQ for z/OS 대상 구역에서 다음 LMOD 입력 항목에 대한 XZMOD 하위 항목을 삭제하십시오. CMQXDCST, CMQXRCTL, CMQXSUPR, CSQCBE00, CSQCBE30, CSQCBP00, CSQCBP10, CSQCBR00, CSQUCVX, CSQUDLQH, CSQVXPCB, CSQVXSPT, CSQXDCST, CSQXRCTL, CSQXSUPR, CSQXTCMI, CSQXTCP, CSQXTNSV, CSQ7DRPS, IMQB23IC, IMQB23IM, IMQB23IR, IMQS23IC, IMQS23IM, IMQS23IR 2. IBM MQ 구역과 Language Environment 구역 사이에 적합한 ZONEINDEX를 설정하십시오. 3. LINK 명령의 FROMZONE 매개변수에서 새 영역을 참조하도록 CSQ8SLDQ를 조정하십시오. CSQ8SLDQ는 SCSQINST 라이브러리에서 찾을 수 있습니다. 4. CSQ8SLDQ를 실행하십시오.

표 53. 새 *SMP/E* 환경 및 라이브러리에서 제품 중 하나를 새 릴리스로 업데이트함 *(*계속*)*

제품	CALLLIBS를 사용하고 SMP/E V3r2 이상인 경우 조치 참고: Language Environment 및 호출 가능 서비스에 대해 세 개의 별도 작업을 실행할 필요가 없습니다. 두 제품에 대해 하나의 작업으로 충분합니다.	LINK 사용 시 조치
호출 가능 서비스	1. 새 라이브러리를 가리키도록 CSSLIB에 대한 DDDEF를 변경하십시오. 2. SMP/E 작업의 경계를 대상 영역으로 설정하십시오. 3. SMPCNTL 카드에서 LINK LMODS CALLLIBS를 지정하십시오. 또한 CHECK, RETRY(YES) 및 RC와 같은 기타 매개변수를 지정할 수도 있습니다. 추가 정보는 *z/OS*의 *SMP/E:* 명령을 참조하십시오. 4. SMP/E 작업을 실행하십시오.	1. IBM MQ for z/OS 대상 구역에서 다음 LMOD 입력 항목에 대한 XZMOD 하위 항목을 삭제하십시오. CMQXRCTL, CMQXSUPR, CSQBSRV, CSQILPLM, CSQXJST, CSQXRCTL, CSQXSUPR, CSQ3AMGP, CSQ3EPX, CSQ3REPL 2. IBM MQ 구역 및 호출 가능한 서비스 구역 사이에서 적절한 ZONEINDEX를 설정하십시오. 3. LINK 명령의 FROMZONE 매개변수에서 새 영역을 참조하도록 CSQ8SLDQ를 조정하십시오. CSQ8SLDQ는 SCSQINST 라이브러리에서 찾을 수 있습니다. 4. CSQ8SLDQ를 실행하십시오.

CALLLIBS를 사용하는 경우 모듈을 다시 링크하는 작업의 예는 693 페이지의 『LINK CALLLIBS 작업 실행』의 내용을 참조하십시오.

z/OS LINK CALLLIBS 작업 실행

CALLLIBS를 사용할 때 모듈을 재링크하기 위한 작업 예입니다.

다음은 SMP/E V3r2 시스템에서 CALLLIBS를 사용할 때 모듈을 재링크하기 위한 작업의 예입니다. JOBCARD 및 IBM MQ for z/OS를 포함하는 SMP/E CSI의 데이터 세트 이름을 제공해야 합니다.

```
//********************************************************************
//* RUN LINK CALLLIBS.
//********************************************************************
//CALLLIBS EXEC PGM=GIMSMP,REGION=4096K
//SMPCSI   DD DSN=your.csi
//         DISP=SHR
//SYSPRINT DD SYSOUT=*
//SMPCNTL DD *
SET BDY(TZONE).
LINK LMODS CALLLIBS .
/*
```

그림 *121. SMP/E LINK CALLLIBS* 작업 예

z/OS IMS에서 OTMA 엑시트 사용

IBM MQ for z/OS와 함께 IMS OTMA(Open Transaction Manager Access) 엑시트를 사용하려는 경우 이 토픽을 사용하십시오.

IMS 트랜잭션의 출력을 IBM MQ로 보내려고 하고 해당 트랜잭션이 IBM MQ에서 시작하지 않은 경우 하나 이상의 IMS OTMA 엑시트를 코딩해야 합니다.

비슷하게 출력을 비OTMA 목적지로 보내기 원하고 트랜잭션이 IBM MQ에서 시작하지 않은 경우, 하나 이사의 IMS OTMA 엑시트로 코딩해야 합니다.

다음 엑시트는 IMS와 IBM MQ 사이에서 사용자 정의 처리를 사용할 수 있도록 하기 위해 IMS에서 사용 가능합니다.

- OTMA 사전 라우팅 엑시트
- 목적지 분석 사용자(DRU) 엑시트

OTMA 엑시트 이름

사전 라우팅 엑시트를 DFSYPRXO로 이름을 지정해야 합니다. 이미 IMS에 있는 모듈 이름과 충돌하지 않는 한, DRU 엑시트를 임의의 이름으로 지정할 수 있습니다.

목적지 분석 사용자 엑시트 이름 지정

IMS에 의해 실행될 OTMA DRU 엑시트의 이름을 지정하려면 CSQ6SYSP 매크로의 OTMACON 키워드의 *Druexit* 매개변수를 사용할 수 있습니다.

오브젝트 ID를 단순화하려면 DRU0xxxx의 이름 지정 규칙 채택을 고려하십시오. 여기서 xxxx는 IBM MQ 큐 관리자의 이름입니다.

OTMACON 매개변수에서 DRU 엑시트 이름을 지정하지 않은 경우 기본값은 DFSYDRU0입니다. 이 모듈의 샘플은 IMS에서 제공됩니다. 이에 관한 정보는 *IMS/ESA®* 사용자 정의 안내서를 참조하십시오.

IMS 목적지에 대한 이름 지정 규칙

IMS 프로그램의 출력을 보내는 목적지에 대한 이름 지정 규칙이 필요합니다. 이것은 IMS 애플리케이션의 CHNG 호출에서 설정되거나 IMS PSB에 존재하는 목적지입니다.

OTMA 엑시트의 샘플 시나리오

IMS에 대한 사전 라우팅 엑시트 및 목적지 라우팅 엑시트의 예에 대해서는 다음 토픽을 사용하십시오.

- 694 페이지의 『사전 라우팅 엑시트 DFSYPRX0』
- 695 페이지의 『목적지 해석 사용자 엑시트』

식별을 단순화하려면 OTMA 목적지 이름을 IBM MQ 큐 관리자 이름(예: 반복되는 IBM MQ 큐 관리자 이름)에 비슷하게 만드십시오. 이 경우에 IBM MQ 큐 관리자 이름이 " **VCPE** "인 경우, CHNG 호출이 설정하는 목적지는 " **VCPEVCPE** "입니다.

관련 개념

683 페이지의 『IMS와 함께 IBM MQ 사용』

IBM MQ -IMS 어댑터 및 IBM MQ - IMS 브릿지는 IBM MQ가 IMS와 상호작용할 수 있도록 허용하는 두 컴포넌트입니다.

관련 정보

IBM MQ 및 IMS
IBM MQ for z/OS의 IMS 및 IMS 브릿지

z/OS 사전 라우팅 엑시트 DFSYPRX0

이 토픽에는 IMS에서 OTMA에 대한 샘플 사전 라우팅 엑시트가 들어 있습니다.

먼저 사전 라우팅 엑시트 DFSYPRX0를 코딩해야 합니다. IMS에 의해 이 루틴으로 전달되는 매개변수는 *IMS/ESA* 사용자 정의 안내서에 문서화되어 있습니다.

이 엑시트는 메시지가 알려진 OTMA 목적지(이 예에서는 VCPEVCPE)를 위한 것인지 여부를 테스트합니다. 맞는 경우 엑시트는 메시지를 송신하는 트랜잭션이 OTMA에서 시작했는지 여부를 확인해야 합니다. 메시지가 OTMA에서 시작한 경우 OTMA 헤더를 가지므로, 0으로 설정된 레지스터 15와 함께 DFSYPRX0에서 엑시트해야 합니다.

- 메시지를 송신하는 트랜잭션이 OTMA에서 시작하지 않은 경우 클라이언트 이름을 올바른 OTMA 클라이언트인 것으로 설정해야 합니다. 이것은 메시지를 송신하려는 IBM MQ 큐 관리자의 XCF 멤버 이름입니다. *IMS/ESA* 사용자 정의 안내서에서 이 값을 설정할 위치를 알려줍니다. 클라이언트 이름(CSQ6SYSP 매크로의

OTMACON 매개변수에서)이 큐 관리자 이름으로 설정할 것을 제안합니다. 이는 기본값입니다. 그런 다음 레지스터 15를 4로 설정하고 DFSYPRX0에서 엑시트해야 합니다.

- 메시지를 전송하는 트랜잭션이 OTMA에서 시작했고 목적지가 OTMA가 아닌 경우 레지스터 15를 8로 설정하고 엑시트해야 합니다.
- 다른 모든 경우에는 레지스터 15를 0으로 설정해야 합니다.

OTMA 클라이언트 이름을 IMS에 알려지지 않은 이름으로 설정하는 경우 애플리케이션 CHNG 또는 ISRT 호출이 A1 상태 코드를 리턴합니다.

둘 이상의 IBM MQ 큐 관리자와 통신하는 IMS 시스템의 경우 각 IBM MQ 큐 관리자에 대해 논리를 반복해야 합니다.

샘플 어셈블러 코드는 695 페이지의 그림 122에 표시되어 있습니다.

```
TITLE 'DFSYPRX0: OTMA PRE-ROUTING USER EXIT'
DFSYPRX0 CSECT
DFSYPRX0 AMODE 31
DFSYPRX0 RMODE ANY
*
SAVE (14,12),,DFSYPRX0&SYSDATE&SYSTIME
SPACE 2
LR  R12,R15        MODULE ADDRESSABILITY
USING DFSYPRX0,R12
*
L   R2,12(,R1)      R2 -> OTMA PREROUTE PARMS
*
LA  R3,48(,R2)      R3 AT ORIGINAL OTMA CLIENT (IF ANY)
CLC  0(16,R3),=XL16'00'  OTMA ORIG?
BNE  OTMAIN        YES, GO TO THAT CODE
*
NOOTMAIN DS 0H            NOT OTMA INPUT
LA  R5,8(,R2)       R5 IS AT THE DESTINATION NAME
CLC  0(8,R5),=C'VCPEVCPE' IS IT THE OTMA UNSOLICITED DEST?
BNE  EXIT0         NO, NORMAL PROCESSING
*
L   R4,80(,R2)      R4 AT ADDR OF OTMA CLIENT
MVC  0(16,R4),=CL16'VCPE' CLIENT OVERRIDE
B   EXIT4         AND EXIT
*
OTMAIN  DS 0H            OTMA INPUT
LA  R5,8(,R2)       R5 IS AT THE DESTINATION NAME
CLC  0(8,R5),=C'VCPEVCPE' IS IT THE OTMA UNSOLICITED DEST?
BNE  EXIT8         NO, NORMAL PROCESSING
```

```
*
EXIT0  DS 0H
LA  R15,0         RC = 0
B   BYEBYE
*
EXIT4  DS 0H
LA  R15,4         RC = 4
B   BYEBYE
*
EXIT8  DS 0H
LA  R15,8         RC = 8
B   BYEBYE
*
BYEBYE  DS 0H
RETURN (14,12),,RC=(15)   RETURN WITH RETURN CODE IN R15
SPACE 2
REQUATE
SPACE 2
END
```

그림 *122. OTMA* 사전 라우팅 엑시트 어셈블러 샘플

z/OS 목적지 해석 사용자 엑시트

이 토픽에는 IMS에 대한 목적지 해석 사용자 엑시트 샘플이 들어 있습니다.

DFSYPRX0에서 레지스터 15를 4를 설정한 경우 또는 트랜잭션의 소스가 OTMA이고 **또한** 레지스터 15를 0으로 설정한 경우 DRU 엑시트가 호출됩니다. 이 예에서 DRU 엑시트 이름은 DRU0VCPE입니다.

DRU 엑시트는 목적지가 VCPEVCPE인지 확인합니다. 맞는 경우 OTMA 사용자 데이터(OTMA에 접두부에 있음)를 다음과 같이 설정합니다.

오프셋
: **OTMA 사용자 데이터**

(10진수)

0
: OTMA 사용자 데이터 길이(이 예에서는 334)

2
: MQMD

326
: 형식에 대한 응답

이들 오프셋은 IBM MQ - IMS 브릿지가 이 정보를 찾을 것으로 예상하는 장소입니다.

DRU 엑시트는 가능한 단순할 것을 제안합니다. 따라서 이 샘플에서는 특정 IBM MQ 큐 관리자에 대해 IMS에서 시작되는 모든 메시지를 동일한 IBM MQ 큐에 넣습니다.

메시지가 지속적이어야 하는 경우 IMS는 동기화된 트랜잭션 파이프를 사용해야 합니다. 이를 수행하려면 DRU 엑시트가 OUTPUT 플래그를 설정해야 합니다. 더 자세한 내용은 *IMS/ESA* 사용자 정의 안내서를 참조하십시오.

이 큐를 처리할 IBM MQ 애플리케이션을 작성하고, MQMD 구조, MQIIH 구조(존재하는 경우) 또는 사용자 데이터의 정보를 사용하여 각 메시지를 그의 목적지로 라우트하십시오.

샘플 어셈블러 DRU 종료는 697 페이지의 그림 123에 표시됩니다.

```
TITLE 'DRU0VCPE: OTMA DESTINATION RESOLUTION USER EXIT'
DRU0VCPE CSECT
DRU0VCPE AMODE 31
DRU0VCPE RMODE ANY
*
SAVE (14,12),,DRU0VCPE&SYSDATE&SYSTIME
SPACE 2
LR  R12,R15            MODULE ADDRESSABILITY
USING DRU0VCPE,R12
*
L   R2,12(,R1)           R2 -> OTMA DRU PARMS
*
L   R5,88(,R2)           R5 ADDR OF OTMA USERDATA
LA  R6,2(,R5)           R6 ADDR OF MQMD
USING MQMD,R6            AS A BASE
*
LA  R4,MQMD_LENGTH+10       SET THE OTMA USERDATA LEN
STH  R4,0(,R5)           = LL + MQMD + 8
*                       CLEAR REST OF USERDATA
MVI  0(R6),X'00'          ...NULL FIRST BYTE
MVC  1(255,R6),0(R6)        ...AND PROPAGATE IT
MVC  256(MQMD_LENGTH-256+8,R6),255(R6) ...AND PROPAGATE IT
*
VCPE   DS  0H
CLC  44(16,R2),=CL16'VCPE'     IS DESTINATION VCPE?
BNE  EXIT4             NO, THEN DEST IS NON-OTMA
MVC  MQMD_REPLYTOQ,=CL48'IMS.BRIDGE.UNSOLICITED.QUEUE'
MVC  MQMD_REPLYTOQMGR,=CL48'VCPE'  SET QNAME AND QMGRNAME
MVC  MQMD_FORMAT,MQFMT_IMS     SET MQMD FORMAT NAME
MVC  MQMD_LENGTH(8,R6),MQFMT_IMS_VAR_STRING
*                       SET REPLYTO FORMAT NAME
B   EXIT0
*
EXIT0  DS  0H
LA  R15,0             SET RC TO OTMA PROCESS
B   BYEBYE             AND EXIT
*
EXIT4  DS  0H
LA  R15,4             SET RC TO NON-OTMA
B   BYEBYE             AND EXIT
*
BYEBYE  DS  0H
RETURN (14,12),,RC=(15)       RETURN CODE IN R15
SPACE 2
REQUATE
SPACE 2
CMQA  EQUONLY=NO
CMQMDA DSECT=YES
SPACE 2
END
```

그림 *123.* 샘플 어셈블러 *DRU* 엑시트

V 9.0.1 z/OS IBM z/OSMF를 사용하여 IBM MQ 자동화

IBM z/OS Management Facility(z/OSMF)에서는 메인프레임 z/OS 시스템의 일상적인 조작 및 관리를 쉽게 관리할 수 있도록 통합된 사용자 지원을 통해 태스크 중심의 웹 브라우저 기반 사용자 인터페이스에서 시스템 관리 기능을 제공합니다.

기존 태스크를 단순화하고 다른 태스크를 자동화하여 z/OSMF는 z/OS 시스템 관리의 일부 영역을 단순화하는 데 도움을 줄 수 있습니다.

자원은 사용자 제공 포털에서 단추를 한 번 클릭하여 프로비저닝하거나 디프로비저닝할 수 있습니다. z/OSMF에서는 이 태스크를 지원할 REST API를 제공합니다.

z/OSMF와 함께 제공된 샘플 마켓플레이스 포털을 사용하여 자원을 프로비저닝 및 디프로비저닝할 수 있습니다. 또는 고급 사용자는 z/OSMF WUI(웹 사용자 인터페이스)를 사용할 수 있습니다.

이 절에서는 z/OSMF을(를) 이해하는 것으로 가정하지만 z/OSMF에 익숙하지 않은 경우 z/OSMF 시작하기를 참조하십시오. 또는 z/OSMF WUI 온라인 도움말에서 이 섹션에 액세스할 수도 있습니다.

z/OS 클라우드 구성에 익숙해야 합니다.

- 클라우드 프로비저닝 - 자원 관리 및 소프트웨어 서비스
- 구성 - 구성 지원 프로그램 및 성능 - 워크로드 관리
- 성능 - 워크로드 관리

시작하기 학습서 - 클라우드와 함께 세부 정보는 이 릴리스의 새로운 기능 절에 있습니다.

z/OSMF 2.2에서는 역할 기반 활동 및 태스크를 소개합니다. 따라서 다음 개념을 이해하는 것이 중요합니다.

도메인
관리자
승인자
테넌트
템플리트
인스턴스
워크플로우

등. *z/OSMF* 프로그래밍 안내서 또는 z/OSMF WUI 도움말의 클라우드 프로비저닝을 참조하십시오.

샘플 IBM MQ z/OSMF 워크플로우 및 연관된 파일이 제공되고, IBM MQ for z/OS UNIX System Services Components 기능의 일부로 설치 가능합니다. 이 기능의 설치 프로세스와 디렉토리 및 파일 구조는 IBM MQ for z/OS 프로그램 디렉토리(IBM Publications Center에서 다운로드할 수 있음)에서 설명됩니다.

샘플 워크플로우는 XML로 작성되며, IBM MQ 큐 관리자, 채널 시작기, 로컬 큐의 프로비저닝(작성) 또는 디프로비저닝(삭제)을 자동화하는 방법과 프로비저닝된 IBM MQ 자원에서 조치를 수행하는 방법을 보여줍니다. 워크플로우내 단계에서 작업을 제출하거나(JCL), REXX execs를 실행하거나, 쉘 스크립트를 처리하거나, REST API 호출을 발행합니다.

샘플은 z/OSMF를 사용하여 달성할 수 있는 기능 유형을 보여주도록 설계되었습니다. 이 경우 z/OSMF 워크플로우는 일반적으로 자원을 프로비저닝하는 데 사용되며, 메시지의 넣기 또는 가져오기와 같은 조치는 특히 IBM MQ 애플리케이션을 사용하여 수행된다고 예상됩니다.

워크플로우 변수 특성이 설정된 경우(다음 절에서 논의한 대로) 제공된 대로 동일한 워크플로우를 실행하거나 필요한 경우 이를 사용자 정의할 수 있습니다. 추가 기능을 수행하는 고유 워크플로우를 작성할 수 있습니다. 샘플 워크플로우를 실행하기 전에 다음을 참조하십시오.

- 698 페이지의 『전제조건』
- 700 페이지의 『보안 설정』
- 702 페이지의 『제한사항』

샘플 워크플로우 애플리케이션은 다음을 위해 제공됩니다.

- 703 페이지의 『IBM MQ 큐 관리자의 프로비저닝 또는 디프로비저닝을 자동화하고 프로비저닝된 큐 관리자에서 조치 수행』
- 704 페이지의 『IBM MQ 로컬 큐의 프로비저닝 또는 디프로비저닝을 자동화하고 프로비저닝된 큐에서 조치 수행』.

관련 개념

597 페이지의 『IBM MQ for z/OS 설정』
이 주제를 IBM MQ for z/OS 시스템 사용자 정의를 위한 단계별 안내서로 사용하십시오.

V 9.0.1 z/OS 전제조건

IBM z/OS Management Facility(z/OSMF)를 IBM MQ와 함께 실행하는 데 필요한 필수조건입니다.

IBM MQ 9.0.1 에서 제공되는 IBM MQ 워크플로우는 z/OSMF의 새 기능을 이용합니다. 이 기능은 z/OS 2.1 및 2.2의 APAR을 통해 제공됩니다. 자세한 내용은 다음 텍스트에 제공되어 있습니다.

1. IBM z/OS Management Facility 2.2를 올바르게 설치하고 구성했습니다. 보안을 사용하는 상태로 실행할 경우 z/OSMF에서 설명하는대로 모든 보안 설정을 구성해야 합니다.
2. 다음에 대해 다음과 같은 APAR을 설치했습니다.

z/OS 2.1

- PI71068
- PI71079
- PI71082
- PI71084
- OA50130

z/OS 2.2

- PI70526
- PI70521
- PI70527
- PI67839
- PI70767
- PI46315
- OA49081
- OA49802
- OA50130

3. z/OSMF Angel(필수) 및 서버 프로세스가 구성되어 있어야 합니다.
4. z/OS 클라우드 환경이 위에서 설명한 것과 z/OSMF에서 설명하는 대로 구성되어 있어야 합니다.
5. IBM MQ for z/OS 9.0.1가 설치되어 있고 제품 로드 라이브러리가 사용 가능합니다.
6. 다음 IBM MQ 큐 관리자 사용자 정의 태스크를 수행했습니다.

태스크	설명
1	z/OS 시스템 매개변수 식별
2	APF에서 IBM MQ 로드 라이브러리에 권한 부여
3	z/OS 링크 목록 및 LPA 업데이트
4	z/OS 프로그램 특성 테이블 업데이트

7. 샘플 워크플로우 및 연관된 파일은 USS(z/OS)에 대해 적합한 UNIX System Services 디렉토리에 설치됩니다.
8. **'/tmp'** USS 디렉토리를 사용할 수 있습니다. provision.xml 워크플로우에서 이 디렉토리에 임시 파일을 작성할 수 있기 때문입니다. 파일이 작성된 경우 일반적으로 워크플로우에서 사용 후 파일을 삭제합니다.
9. `deprovision.xml` 파일에는 CSQ4ZWS1.rexx 및 CSQ4ZWS2.rexx REXX exec을 호출하는 단계가 있습니다. 이 exec는 큐 관리자 및 채널 시작기 서브시스템이 중지되기를 기다립니다. exec는 시스템 호출로 USS 'SLEEP' 명령을 호출합니다.

   USS 구성에 따라 'SLEEP' 명령이 코딩한 대로 작동하지 않을 수 있습니다. 처리 중 'SLEEP' 명령을 찾을 수 없음을 나타내는 오류가 발생하면 exec CSQ4ZWS1.rexx 및 CSQ4ZWS2.rexx에서 다음 행을 바꿔 볼 수 있습니다.

```
CALL SYSCALLS('ON')              /* Enable USS calls    */
ADDRESS SYSCALL
"SLEEP" 10                            /* Sleep for 10 seconds */
CALL SYSCALLS 'OFF'              /* Disable USS calls    */
```

   다음 컴파일 명령으로 바꿉니다.

```
'sleep' 10
```

그런 다음, OMVS(Open MVS) **env** 명령을 실행하여 PATH 환경 변수 설정을 확인하십시오. **sleep** 명령을 포함하는 디렉토리가 PATH에 정의되어 있는지 확인하십시오. **sleep** 명령은 일반적으로 /bin 디렉토리에 있습니다.

10. z/OSMF가 시작되었는지 확인하십시오.

    Angel 및 서버 z/OSMF 프로세스가 모두 시작되어야 하고 z/OSMF WUI(웹 사용자 인터페이스)가 가동 및 실행 중이어야 합니다. 자세한 정보는 Liberty 프로파일: z/OS의 프로세스 유형을 참조하십시오.

    REST API을(를) 사용하여 워크플로우를 구동하려는 경우에도 z/OSMF WUI를 시작해야 합니다. z/OSMF WUI는 워크플로우 작성 및 처리를 모니터링하는 데 유용할 수 있습니다.

관련 개념

697 페이지의 『IBM z/OSMF를 사용하여 IBM MQ 자동화』
IBM z/OS Management Facility(z/OSMF)에서는 메인프레임 z/OS 시스템의 일상적인 조작 및 관리를 쉽게 관리할 수 있도록 통합된 사용자 지원을 통해 태스크 중심의 웹 브라우저 기반 사용자 인터페이스에서 시스템 관리 기능을 제공합니다.

V 9.0.1 z/OS 보안 설정

z/OSMF를 실행하는 데 필요한 보안 설정입니다.

다음 사용자 ID 변수 특성이 특성 파일에 정의됩니다. 자세한 정보는 706 페이지의 『워크플로우 실행』의 내용을 참조하십시오.

사용자 ID 특성	설명
CSQ_USERID	워크플로우 단계를 실행하는 데 사용되는 사용자 ID입니다. 그러나 선택된 단계(일반적으로 상승된 수준의 권한이 필요함)는 다음 텍스트에 나열된 **CSQ_ADMIN_*** 사용자 ID를 기반으로 다른 사용자 ID를 사용하여 실행됩니다. 사용 중인 사용자 ID는 워크플로우의 해당 단계에서 **runAsUser** 특성으로 식별됩니다.
CSQ_ADMIN_APF_USERID	APF가 큐 관리자 시스템 매개변수 모듈을 포함하는 로드 라이브러리에 권한을 부여할 때 사용할 사용자 ID입니.
CSQ_APF_APPROVAL_ID	사용자가 데이터 세트 APF 권한 부여 단계를 사용자 CSQ_ADMIN_APF_USERID로 실행할 수 있게 허용하는 데 사용되는 승인 ID입니다.
CSQ_ADMIN_CONSOLE_USERID	z/OS 콘솔 명령을 발행하는 실행에서 단계를 실행할 때 사용되는 사용자 ID입니다. ⚠ **주의:** 이 사용자 ID는 시작된 태스크 프로파일 (MVS.START.STC. *) 에 대한 UPDATE 액세스를 허용해야 합니다. "OPERCMDS" 클래스에 있다. 자세한 내용은 IBM Documentation의 IBM z/OS 정보에 있는 *z/OS* 섹션을 참조하십시오.
CSQ_CONSOLE_APPROVAL_ID	사용자 CSQ_ADMIN_CONSOLE_USERID로 z/OS 콘솔 명령을 실행하는 단계를 사용자가 실행할 수 있도록 허용하는 데 사용된 승인 ID입니다.
CSQ_ADMIN_SAF_USERID	SAF 명령을 실행할 때 사용할 사용자 ID입니다.
CSQ_SAF_APPROVAL_ID	사용자 CSQ_ADMIN_SAF_USERID로 SAF 명령을 실행하는 단계를 사용자가 실행할 수 있도록 허용하는 데 사용된 승인 ID입니다.
CSQ_ADMIN_SSI_USERID	z/OS에 프로비저닝될 서브시스템을 식별하기 위해 SETSSI 명령을 실행할 때 사용할 사용자 ID입니다.
CSQ_SSI_APPROVAL_ID	사용자 CSQ_ADMIN_SSI_USERID로 SETSSI 명령을 실행하는 단계를 사용자가 실행할 수 있도록 허용하는 데 사용된 승인 ID입니다.

참고: 워크플로우 프로비저닝과 디프로비저닝을 실행하는 데 사용할 사용자 ID에는 충분한 권한이 있어야 합니다.

1. 큐 관리자 프로비저닝과 디프로비저닝 워크플로우는 APF 권한 부여 데이터 세트에 SETPROG 명령을 사용합니다. 사용자 ID가 CSQ_ADMIN_APF_USERID 특성에 설정되었거나 워크플로우를 실행하는 데 사용할 사용자 ID가 이 명령을 실행하도록 허용해야 합니다. 다음 명령을 실행하여 이를 달성할 수 있습니다.

```
PERMIT MVS.SETPROG CLASS(OPERCMDS) ID(value of CSQ_ADMIN_APF_USERID) ACCESS(UPDATE)
```

   참고: SETPROG 명령은 z/OS 시스템의 IPL에서 지속되지 않을 수 있습니다. 따라서 IPL 후에 다음 SETPROG 명령을 수동으로 실행해야 할 수도 있습니다.

```
SETPROG APF,ADD,DSN=value of CSQ_AUTH_LIB_HLQ.value of CSQ_SSID.APF.LOAD,SMS
```

   SETPROG 명령에 대한 자세한 내용은 RACF를 사용하여 APF 목록 제어의 내용을 참조하십시오.

   또한 APF 권한이 부여된 라이브러리를 제어하도록 FACILITY 클래스를 사용할 수 있으므로, 다음 명령을 실행해야 할 수도 있습니다.

```
PERMIT CSVAPF.libname CLASS(FACILITY) ID(value of CSQ_ADMIN_APF_USERID)
       ACCESS(UPDATE)
```

2. 큐 관리자 프로비전 워크플로우의 단계에서는 z/OS에 대한 IBM MQ 서브시스템을 식별하기 위해 SETSSI 명령을 실행합니다. CSQ_ADMIN_SSI_USERID 특성에 설정된 사용자 ID는 이 명령을 사용하도록 허용되어야 합니다. 다음 명령을 실행하여 이를 달성할 수 있습니다.

```
PERMIT MVS.SETSSI.ADD CLASS(OPERCMDS) ID(value of CSQ_ADMIN_SSI_USERID)
        ACCESS(CONTROL)
```

   참고: SETSSI 명령을 통해 z/OS에 식별된 서브시스템은 z/OS 시스템의 IPL에서 지속되지 않습니다. IPL 후에 다음 SETSSI 명령을 수동으로 실행해야 할 수도 있습니다.

```
SETSSI  ADD,S='value of CSQ_SSID',I=CSQ3INI,
       P='CSQ3EPX,value of CSQ_CMD_PFX,S'
```

   SETSSI 명령에 대한 자세한 내용은 SETSSI 명령을 참조하십시오.

3. 워크플로우는 큐 관리자 명령을 실행합니다. 따라서 보안을 사용하려는 경우 특성 CSQ_ADMIN_RACF_USERID에 설정된 사용자 ID(또는 워크플로우를 실행하는 데 사용되는 사용자 ID)에 대해 사용하는 클래스에 따라 MQADMIN 또는 MXADMIN 클래스에 대한 CLAUTH(클라이언트 인증) 권한을 부여해야 합니다. 이는 사용자 ID가 이 클래스에 대한 보안 프로파일을 정의할 수 있도록 허용하기 위함입니다. 다음 명령을 실행하여 이를 달성할 수 있습니다.

```
ALTUSR value of CSQ_ADMIN_RACF_USERID CLAUTH(MQADMIN)
```

   CLAUTH에 대한 자세한 내용은 CLAUTH(클래스 권한) 속성의 내용을 참조하십시오.

4. deprovision.xml 워크플로우는 z/OS 명령(예: DISPLAY ACTIVE 작업, CANCEL 또는 FORCE 서브시스템)을 실행하므로, CSQ_ADMIN_CONSOLE_USERID 특성에 설정된 사용자 ID(워크플로우를 실행하는 데 사용되는 사용자 ID)에는 이러한 명령을 실행할 적합한 권한이 있어야 합니다.
5. 소프트웨어 서비스 태스크의 템플리트 테이블을 사용하여 큐 관리자 인스턴스를 요청하는 사용자에게 z/OSMF에서 정의한 대로 z/OSMF 및 구성 지원자에 대한 액세스 권한이 있어야 합니다.
6. 큐 관리자를 제공하는 사용자의 사용자 ID에는 CSQ_PROC_LIB 변수로 정의된 PROCLIB 데이터 세트에서 멤버를 추가 및 삭제할 수 있는 권한이 필요합니다.
7. 큐 프로비저닝 전에 큐 관리자를 먼저 프로비저닝해야 합니다.
8. `queueLoad.xml` 및 `queueOffload.xml` 워크플로우를 사용하려면 사용되는 데이터 세트를 미리 정의해야 합니다. 또한 이러한 워크플로우를 실행하는 데 사용되는 사용자 ID에는 데이터 세트에 대한 UPDATE 권한이 부여되어야 합니다.
9. 큐 관리자 `provision.xml` 워크플로우의 단계에서 현재 서브시스템 보안을 비활성화합니다. IBM MQ 자원을 보호하기 위해 적절한 보안 명령을 추가하여 서브시스템 보안을 사용 가능하게 하도록 `csq4znse.jcl`

작업을 수정할 수 있습니다. 그러나 명령을 추가하려면 deprovision.xml 워크플로우에서 제출한 `csq4dse.jcl`의 보안 권한을 삭제하는 명령을 추가해야 합니다.

참고: 이 단계에서 RACF 보안 명령을 발행합니다. 대체 보안 제품을 사용하는 경우 보안 제품에 해당하는 명령을 발행하도록 이 단계를 수정해야 합니다.

네트워크 요구사항

큐 관리자 템플리트 및 해당 템플리트의 자원을 추가할 때 **네트워크 자원 풀 작성**을 클릭하십시오. 이 템플리트에 대한 네트워크 자원이 포함된 자원 풀이 작성됩니다.

구성 지원을 사용할 경우 이 템플리트에 할당되는 포트 수에 대한 한도를 정의하여 네트워크 관리자가 이 네트워크 자원 풀 정의를 수행해야 합니다.

템플리트 인스턴스마다 `provision.xml` 워크플로우가 범위내 포트를 할당하고, 해당 포트를 수신하기 위한 리스너를 시작합니다.

IBM 워크로드 관리자를 사용한 분류

큐 관리자 및 채널 시작기 주소 공간을 WLM을 사용하여 분류하려면 큐 관리자를 프로비저닝하기 위한 템플리트를 추가할 때 이를 지정해야 합니다.

분류할지 여부는 `workflow_variables.properties`파일에 설정된 플래그 **`CSQ_DEFINE_MSTR_WLM_RULE`** 및 **`CSQ_DEFINE_CHIN_WLM_RULE`**에 의해 제어됩니다.

WLM을 사용한 분류에 대한 자세한 정보는 *z/OSMF* 구성 안내서를 참조하십시오.

관련 개념

698 페이지의 『전제조건』
IBM z/OS Management Facility(z/OSMF)를 IBM MQ와 함께 실행하는 데 필요한 필수조건입니다.

V 9.0.1 z/OS 제한사항

z/OSMF를 IBM MQ와 같이 사용할 경우의 제한사항입니다.

1. `provision.xml` 워크플로우는 현재 다음과 같은 강조표시된 큐 관리자 사용자 정의 태스크를 자동화합니다.

태스크	설명
1	z/OS 시스템 매개변수 식별
2	APF에서 IBM MQ 로드 라이브러리에 권한 부여**(provision.xml은 일부 라이브러리에 대해 APF 권한 부여를 수행함)**
3	z/OS 링크 목록 및 LPA 업데이트
4	z/OS 프로그램 특성 테이블 업데이트
5	**z/OS에 대해 IBM MQ 서브시스템 정의**
6	**IBM MQ 큐 관리자용 프로시저 작성**
7	**채널 시작기용 프로시저 작성**
8	**z/OS WLM 서비스 클래스에 대해 IBM MQ 서브시스템 정의**
9	커플링 기능 오프로드 스토리지 환경 선택 및 설정
10	커플링 기능 설정
11	ESM 보안 제어 구현
12	SYS1.PARMLIB 멤버 업데이트
13	**초기화 입력 데이터 세트 사용자 정의**

태스크	설명
14	**부트스트랩 및 로그 데이터 세트 작성**
15	**페이지 세트 정의**
16	Db2 데이터 공유 그룹에 IBM MQ 입력 항목 추가
17	**시스템 매개변수 모듈 일부 조정**
18	**채널 시작기 매개변수 일부 조정**
19	배치, TSO, RRS 어댑터 설정
20	조작 및 제어판 설정
21	IBM MQ 덤프 형식화 멤버 포함
22	정보 메시지 억제
23	Advanced Message Security에 대한 시스템 DIAG 멤버 업데이트
24	Advanced Message Security용 프로시저 작성
25GB	시작된 태스크 사용자 고급 메시지 보안 설정
26	Advanced Message Security의 보안 관리자에게 RACDCERT 권한 부여
27	사용자에게 Advanced Message Security의 자원 권한 부여

2. 굵은 텍스트로 강조표시되지 않은 사용자 정의 태스크는 필요한 경우 수동으로 수행해야 합니다.
3. 샘플 INP1 및 INP2 멤버는 현재 그대로 사용됩니다. 필요한 경우, 이 멤버에서 정의하는 자원을 제어하도록 추가 특성을 정의할 수 있습니다.
4. 특성 파일에 나열된 특정 특성과 관련된 주석은 해당 특성 사용 시 제한사항을 나타냅니다. 자세한 정보는 706 페이지의 『워크플로우 실행』의 내용을 참조하십시오.

관련 개념

700 페이지의 『보안 설정』
z/OSMF를 실행하는 데 필요한 보안 설정입니다.

V 9.0.1 z/OS IBM MQ 오브젝트의 프로비저닝 자동화

큐 관리자 및 로컬 큐의 프로비저닝을 자동화하는 샘플이 제공됩니다.

IBM MQ 큐 관리자의 프로비저닝 또는 디프로비저닝을 자동화하고 프로비저닝된 큐 관리자에서 조치 수행

다음과 같이 큐 관리자에 특정한 샘플 z/OSMF 워크플로우가 제공됩니다.

워크플로우 이름	설명
provision.xml	IBM MQ for z/OS 큐 관리자 프로비저닝 이 샘플 워크플로우에서는 다음을 수행합니다. • 큐 관리자의 필수 시스템 자원을 프로비저닝합니다. • 채널 시작기의 필수 시스템 자원을 프로비저닝합니다. • 큐 관리자 시작(이때 채널 시작기 및 TCP/IP 리스너도 시작됨) • 샘플 큐 관리자 설치 확인 프로그램을 실행합니다. 환경 특성은 다른 특성의 큐 관리자 프로비저닝을 제어하도록 설정할 수 있습니다. 추가 정보는 706 페이지의 『워크플로우 실행』의 내용을 참조하십시오.

워크플로우 이름	설명
	참고: 이 워크플로우에 대한 템플리트 추가를 지원하기 위해 Manifest 파일 (`provision.mf`)이 제공됩니다. 이 파일에는 추가 정보를 포함하는 **qaas_readme.pdf** 파일에 대한 참조가 포함되어 있습니다. 템플리트가 추가되면 링크를 통해 파일에 액세스할 수 있습니다.
deprovision.xml	IBM MQ for z/OS 큐 관리자 디프로비저닝 이 샘플 워크플로우에서는 다음을 수행합니다. • 채널 시작기를 중지하고(TCP/IP 리스너도 중지) 큐 관리자를 중지합니다. • 서브시스템이 중지하기를 기다립니다. • 모든 채널 시작기 및 큐 관리자 시스템 자원을 디프로비저닝합니다.
startQMgr.xml	IBM MQ for z/OS 큐 관리자 시작 이 샘플 워크플로우는 큐 관리자를 시작합니다(이때 채널 시작기 및 TCP/IP 리스너도 시작됨).
stopQMgr.xml	IBM MQ for z/OS 큐 관리자 중지 이 샘플 워크플로우는 채널 시작기를 중지하고(TCP/IP 리스너도 중지) 큐 관리자를 중지합니다.

각 워크플로우는 하나 이상의 단계를 수행합니다. 워크플로우에서 주석은 각 단계에서 수행하는 기능을 설명합니다. 일부 단계는 데이터 입력만 요청하지만, 일부 단계는 JCL을 제출하고, REXX execs, 쉘 스크립트를 호출하거나 REST API 호출을 실행하여 명시된 기능을 수행합니다.

JCL 또는 REXX exec 파일의 정확한 이름은 각 단계를 참조하십시오. 워크플로우 및 연관된 JCL 또는 REXX exec 파일은 하나 이상의 변수 XML 파일에서 선언된 변수를 참조합니다. 자세한 정보는 706 페이지의 『워크플로우 변수 선언 파일』의 내용을 참조하십시오.

deprovision, **startQMgr** 및 **stopQMgr**을 프로비저닝된 IBM MQ for z/OS 큐 관리자에 대한 조치로 수행할 수 있습니다.

IBM MQ 로컬 큐의 프로비저닝 또는 디프로비저닝을 자동화하고 프로비저닝된 큐에서 조치 수행

다음과 같이 큐에 특정한 샘플 z/OSMF 워크플로우가 제공됩니다.

워크플로우 이름	설명
defineQueue.xml	로컬 큐 정의 이 샘플 워크플로우는 z/OSMF 워크플로우를 사용하여 특성 설정에 따라 크기 조정된 큐(작게, 보통 또는 크게)를 정의하는 방법을 보여줍니다. **참고:** 이 워크플로우에 대한 템플리트 추가를 지원하기 위해 Manifest 파일 (`provision.mf`)이 제공됩니다. 이 파일에는 추가 정보를 포함하는 **qaas_readme.pdf** 파일에 대한 참조가 포함되어 있습니다. 템플리트가 추가되면 링크를 통해 파일에 액세스할 수 있습니다.
displayQueue.xml	로컬 큐의 선택한 속성 표시 이 샘플 워크플로우는 로컬 큐의 선택한 속성을 표시합니다. 속성은 z/OSMF 변수에서 리턴되며(변수 이름에 대해서는 워크플로우의 단계 참조) 이어서 표시됩니다. 필요한 경우 변수의 컨텐츠는 REST API를 사용하여 액세스할 수 있습니다. 자세한 정보는 z/OSMF 프로그래밍 안내서의 *REST APIs for Cloud Provisioning* 및 z/OSMF 워크플로우 서비스를 참조하십시오.

워크플로우 이름	설명
deleteQueue.xml	로컬 큐 삭제 이 샘플 워크플로우는 지정된 큐 관리자에서 로컬 큐를 삭제합니다.
putQueue.xml	로컬 큐에 하나 이상의 메시지 넣기 이 샘플 워크플로우는 로컬 큐에 하나 이상의 메시지를 넣습니다. 메시지 텍스트를 지정할 수 있지만, 로컬 큐에도 둘 이상의 메시지를 동시에 넣는 경우 동일한 메시지 텍스트를 사용합니다.
getQueue.xml	로컬 큐에서 하나 이상의 메시지 가져오기 이 샘플 워크플로우는 로컬 큐에서 하나 이상의 메시지를 가져옵니다. 메시지는 z/OSMF 변수에서 리턴되며(변수 이름에 대해서는 워크플로우의 단계 참조) 이어서 표시됩니다. 필요한 경우 REST API를 사용하여 변수의 컨텐츠에 액세스할 수 있습니다. 자세한 정보는 클라우드 프로비저닝 REST API를 참조하고 z/OSMF 워크플로우 서비스도 참조하십시오.
loadQueue.xml	데이터 세트에서 로컬 큐로 메시지 로드 이 샘플 워크플로우는 데이터 세트에서 로컬 큐로 메시지를 로드합니다. 데이터 세트의 기본 이름은 특성을 설정하여 지정합니다. 자세한 정보는 706 페이지의 『워크플로우 실행』의 내용을 참조하십시오.
offloadQueue.xml	로컬 큐에서 데이터 세트로 메시지를 오프로드합니다. 이 샘플 워크플로우는 로컬 큐에서 데이터 세트로 메시지를 오프로드합니다. 데이터 세트의 기본 이름은 특성을 설정하여 지정합니다. 자세한 정보는 706 페이지의 『워크플로우 실행』의 내용을 참조하십시오.
clearQueue.xml	로컬 큐에서 메시지 지우기 이 샘플 워크플로우는 로컬 큐에 있는 모든 메시지를 지웁니다(삭제).

참고사항:

1. **Put Queue** 조치를 사용하면 일부 메시지 데이터를 입력하고 하나 이상의 메시지를 큐에 저장할 수 있습니다. 지정한 요청 중에 둘 이상의 메시지를 큐에 보관할 경우 동일한 메시지 데이터가 사용됩니다.
2. loadQueue.xml 및 offloadQueue.xml 워크플로우는 본질적으로 IBM MQ for Multiplatforms에서 사용 가능한 **dmpmqmsg** 유틸리티인 IBM MQ for z/OS QLOAD 유틸리티를 호출합니다. 따라서 데이터 세트에서 큐 또는 큐에서 데이터 세트로 로드되는 메시지는 **dmpmqmsg** 형식으로 예상됩니다.

   loadQueue 및 offloadQueue 조치를 실행하는 가장 쉬운 방법은 다음을 수행하는 것입니다.

   a. **putQueue**를 여러 번 실행하여 일부 메시지를 큐에 넣습니다.
   b. **offloadQueue**를 사용하여 큐에서 데이터 세트로 메시지를 오프로드합니다.
   c. 필요한 경우 **clearQueue**를 발행하여 큐에서 모든 메시지를 제거합니다.
   d. **loadQueue**를 사용하여 데이터 세트에서 동일한 큐 또는 다른 큐로 메시지를 로드합니다.

   dmpmqmsg 형식에 관심이 있는 경우 오프로드 요청을 발행한 후 데이터 세트의 컨텐츠를 찾아볼 수 있습니다.
3. **displayQueue**, **deleteQueue**, **putQueue**, **getQueue**, **loadQueue**, **offloadQueue** 및 **clearQueue**를 프로비저닝된 IBM MQ for z/OS 로컬 큐에 대한 조치로 수행할 수 있습니다. 조치 및 조치 파일에 대한 자세한 정보는 *z/OSMF* 프로그래밍 안내서를 참조하십시오.
4. 기본적으로 모든 조치 관련 워크플로우가 삭제됩니다. 이에 대한 이유는 사용자가 워크플로우를 정리할 필요성을 최소화하기 위함입니다.

그러나 조치로 인해 일부 출력에 문제가 있을 수 있습니다. 예를 들어, **displayQueue** 및 **getQueue** 조치 모두가 출력을 생성합니다.

조치를 수행하는 즉시 관련 워크플로우가 삭제되기 때문에 출력을 볼 수 없습니다. 따라서 z/OS WUI에서 워크플로우 조치를 수행하는 경우 출력을 보려는 각 조치에 대해 **<workflow>** 태그에서 **cleanAfterComplete** 플래그를 잘못된 로 설정해야 합니다.

예를 들어, **displayQueue**의 출력을 표시하려면 플래그를 다음과 같이 설정합니다.

```
<action name="displayQueue">
  <workflow cleanAfterComplete="false">
  ...

  ...
  </workflow>
</action>
```

그러나 이는 조치 관련 워크플로우를 수동으로 정리해야 함을 의미합니다.

각 샘플 z/OSMF 워크플로우는 하나 이상의 단계를 수행합니다. 워크플로우에서 주석은 각 단계에서 수행하는 기능을 설명합니다. 일부 단계는 데이터 입력만 요청하지만, 일부 단계는 JCL을 제출하고 다른 단계는 REXX exec을 호출하여 명시된 기능을 수행합니다.

JCL 또는 REXX exec 파일의 정확한 이름은 각 단계를 참조하십시오. 워크플로우 및 연관된 JCL 또는 REXX exec 파일은 하나 이상의 706 페이지의 『워크플로우 변수 선언 파일』에서 선언된 변수를 참조합니다.

관련 개념

702 페이지의 『제한사항』
z/OSMF를 IBM MQ와 같이 사용할 경우의 제한사항입니다.

V 9.0.1 z/OS 워크플로우 실행

샘플 z/OSMF 워크플로우에서 참조하는 파일에 대한 설명과 워크플로우를 실행하는 방법입니다.

워크플로우 변수 선언 파일

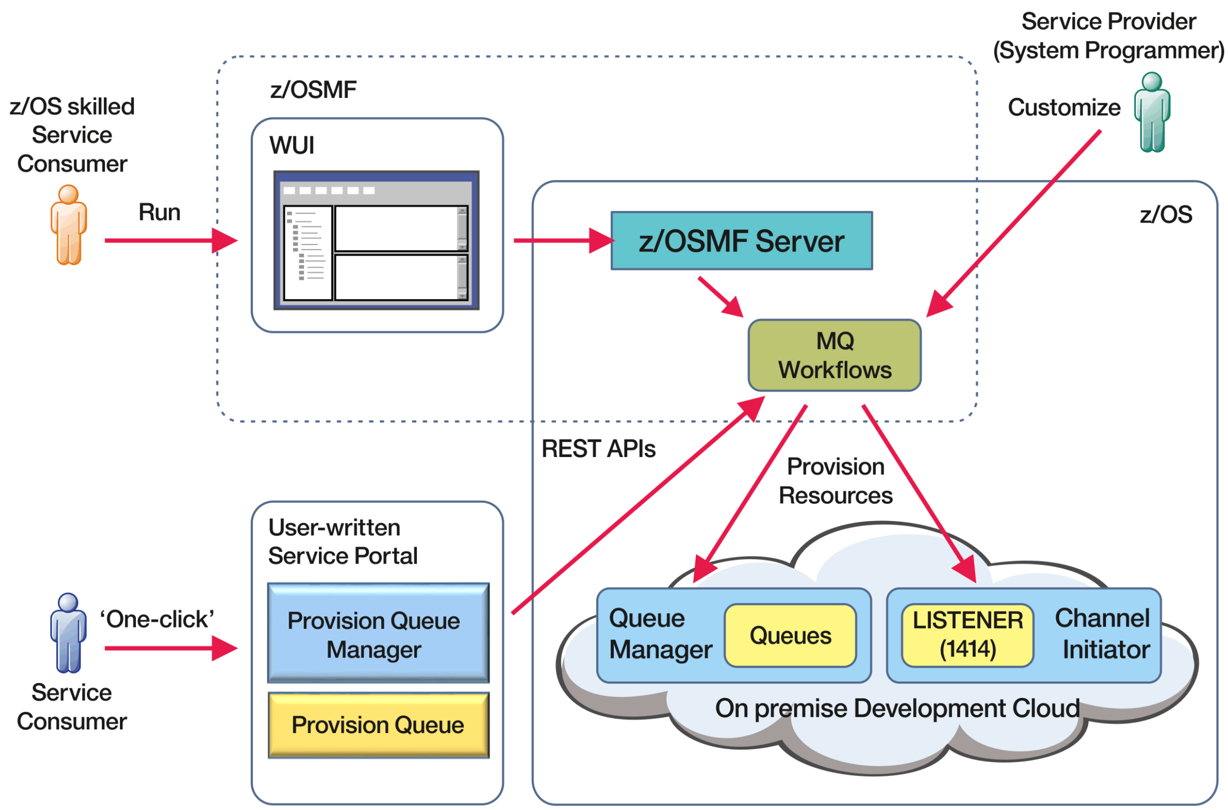
다음 파일은 z/OSMF 워크플로우 및 연관된 JCL 또는 REXX exec 파일에서 참조하는 변수를 선언합니다.

워크플로우 변수 선언 파일 이름	설명
common_variables.xml	큐 관리자(및 채널 시작기)와 큐 워크플로우 모두에 공통되는 변수.
qmgr_variables.xml	큐 관리자(및 채널 시작기) 워크플로우에 특정한 변수.
queue_variables.xml	큐 워크플로우에 특정한 변수
tcpip_variables.xml	큐 관리자 (및 채널 시작기) 워크플로우에 특정하고, TCP/IP 자원을 식별하는 데 사용되는 변수.

참고: 속성의 기본 표시는 *private*입니다. z/OSMF REST API를 사용하여 변수를 조회하고 선택된 변수를 *public*으로 표시하려면 다음을 수행하십시오. 그러나 예를 들어 다음과 같은 경우 기본값을 변경할 수 있습니다.

워크플로우 실행

그림 *124. IBM MQ for z/OS* 자원의 '원클릭' 프로비저닝



워크플로우를 실행하려면 다음 파일에서 일부 특성을 설정해야 합니다.

워크플로우 변수 특성 파일 이름	설명
workflow_variables.properties	워크플로우 변수에 대한 초기 특성. 파일에서 주석은 각 특성의 목적을 보여줍니다. • 대괄호(< >) 내에 있는 특성은 사용자 특정 값으로 설정해야 합니다. • 환경 특성은 개발(DEV), 테스트(TEST), 품질 보증(QA) 또는 프로덕션(PROD) 환경에 대해 큐 관리자를 프로비저닝하기 위해 설정할 수 있습니다. 추가 특성 설정은 각 환경에서 프로비저닝할 큐 관리자의 특성을 제어합니다. 예를 들어, 환경 유형마다 활성 로그 수, 페이지세트 수를 바꿀 수 있습니다. • 기타 특성은 IBM MQ 기본값으로 설정되지만, 필요한 경우 로컬 규칙을 만족하도록 수정할 수 있습니다.

일반적으로 특성을 설정하면 워크플로우를 그대로 실행할 수 있습니다. 그러나 필요한 경우 기존 단계를 수정 또는 제거하거나 새 단계를 추가하도록 워크플로우를 사용자 정의할 수 있습니다.

워크플로우를 실행할 수 있습니다.

• z/OSMF WUI에서

  WUI의 클라우드 프로비저닝 -> 소프트웨어 서비스에서 워크플로우는 자동 또는 수동 모드에서 실행될 수 있습니다. 수동 모드는 테스트할 때 유용하며, 두 모드에서 워크플로우의 각 단계 진행 상황을 모니터할 수 있습니다.

  자세한 정보는 z/OSMF WUI 도움말에서 클라우드 프로비저닝을 참조하고 워크플로우 작성도 참조하십시오.

• z/OSMF REST 워크플로우 서비스 사용.

  REST 워크플로우 서비스를 사용하여 REST API를 통해 워크플로우를 실행할 수 있습니다. 이 모드는 사용자가 작성한 포털에서 원클릭 조작을 작성할 때 유용합니다.

자세한 정보는 클라우드 프로비저닝 REST API를 참조하고 z/OSMF 워크플로우 서비스도 참조하십시오.

- z/OSMF와 함께 제공된 샘플 마켓플레이스 포털 사용.

관련 개념

703 페이지의 『IBM MQ 오브젝트의 프로비저닝 자동화』
큐 관리자 및 로컬 큐의 프로비저닝을 자동화하는 샘플이 제공됩니다.

z/OS MQ Adv.VUE V 9.0.3 IBM MQ Advanced for z/OS VUE 구성

이 정보를 사용하여 IBM MQ Advanced for z/OS VUE 인타이틀먼트로 사용 가능한 기능을 구성할 수 있습니다.

이 태스크 정보

From IBM MQ 9.0.3, you can use the features provided in the IBM MQ Advanced for z/OS VUE Connector Pack to simplify MFT topology on z/OS and make use of the connectivity from IBM MQ Advanced for z/OS, Value Unit Edition queue managers to the IBM Blockchain service in IBM Cloud (formerly Bluemix).

V 9.0.4 From IBM MQ 9.0.4, you can connect an IBM MQ classes for JMS, or IBM MQ classes for Java, application to a queue manager on z/OS, that has the **ADVCAP**(ENABLED) attribute, by using a client connection.

프로시저

1. IBM MQ Advanced for z/OS, Value Unit Edition과의 Managed File Transfer 에이전트 원격 연결을 사용으로 설정하십시오.
2. IBM Cloud에서 IBM Blockchain 서비스와 함께 사용하도록 IBM MQ Advanced for z/OS VUE 를 구성하십시오.

z/OS MQ Adv.VUE V 9.0.3 IBM Cloud (formerly Bluemix) 에서 IBM Cloud Product Insights 서비스와 함께 사용하도록 IBM MQ Advanced for z/OS VUE 구성

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

z/OS MQ Adv.VUE V 9.0.3 IBM Cloud (formerly Bluemix)에서 IBM Cloud Product Insights 서비스 인스턴스 작성

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

z/OS MQ Adv.VUE V 9.0.3 IBM Cloud (formerly Bluemix) 에서 IBM Cloud Product Insights 서비스 인스턴스와 함께 사용할 z/OS 큐 관리자 구성

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

V 9.0.4 z/OS MQ Adv.VUE HTTP 프록시를 통해 IBM Cloud에서 IBM Cloud Product Insights에 연결

IBM Cloud Product Insights 서비스는 더 이상 사용할 수 없습니다. 자세한 정보는 다음 블로그 포스트를 참조하십시오. Service Deprecation: IBM Cloud Product Insights.

z/OS MQ Adv.VUE V 9.0.3 리모트 z/OS 큐 관리자에 MFT 에이전트 연결

IBM MQ Advanced for z/OS VUE의 제품 ID(PID)를 사용하여 실행 중인 z/OS의 Managed File Transfer 에이전트는 클라이언트 연결을 사용하여 z/OS의 리모트 큐 관리자에 연결할 수 있습니다.

자세한 정보는 IBM MQ Advanced for z/OS, Value Unit Edition과의 에이전트 원격 연결만 사용을 참조하십시오.

z/OS MQ Adv. VUE V 9.0.3 블록체인과 함께 사용하도록 IBM MQ Advanced for z/OS VUE 구성

IBM MQ Bridge to blockchain을(를) 설정하고 실행하여 z/OS 큐 관리자 및 IBM Blockchain의 IBM MQ을(를) 안전하게 연결하십시오. 브릿지를 사용하면 IBM MQ Advanced 큐 관리자에 연결되는 메시징 애플리케이션을 통해 블록체인의 자원에 비동기적으로 연결하고 자원의 상태를 조회 및 업데이트할 수 있습니다.

시작하기 전에

- IBM MQ Bridge to blockchain은 IBM MQ Advanced for z/OS Value Unit Edition 9.0.4에 있는 커넥터 팩의 일부로 사용할 수 있습니다. 동일한 명령 레벨에서 실행 중인 z/OS 큐 관리자에 연결할 수 있습니다.
- IBM MQ Bridge to blockchain은 Hyperledger Fabric 1.0 architecture 기반의 블록체인 네트워크에서 사용할 수 있도록 지원됩니다.
- IBM MQ Bridge to blockchain은 다음이 설치된 x86 Linux 환경에서 설치, 구성 및 실행되어야 합니다.
  - IBM MQ 9.0.3 재배포 가능 Java 클라이언트.
  - IBM Java runtime environment 버전 8.

  IBM MQ 9.0.4 재배포 가능 Java 클라이언트 및 IBM Java runtime environment 버전 8이 이미 설치되어 있는 경우, 710 페이지의 『4』 및 711 페이지의 『5』단계를 완료하지 않아도 됩니다.

이 태스크 정보

블록체인은 네트워크에서 피어 사이의 트랜잭션에 동의하는 블록 체인으로 구성되는 공유 및 분배식 디지털 원장입니다. 체인 내의 각 블록은 이전 블록에 링크되며 이런 방식으로 계속되어 첫 번째 트랜잭션에 링크됩니다.

IBM Blockchain은 Hyperledger Fabric을 기반으로 빌드되며, Docker를 사용하거나 IBM Cloud (formerly Bluemix)의 컨테이너 클러스터에서 로컬로 개발할 수 있습니다. IBM Blockchain 네트워크를 프로덕션 중에 활성화하고 사용하여 높은 수준의 보안, 개인정보 보호 및 성능으로 비즈니스 네트워크를 빌드하고 관리할 수도 있습니다. 자세한 정보는 IBM Blockchain Platform을 참조하십시오.

Hyperledger Fabric is an open source, enterprise blockchain framework that is developed collaboratively by members of the Hyperledger Project, including IBM as the initial code contributor. Hyperledger Project, 또는 Hyperledger는 교차 산업 블록체인 기술을 발전시키기 위한 Linux Foundation 오픈 소스, 글로벌, 협업 이니셔티브입니다. 자세한 정보는 IBM Blockchain, Hyperledger 프로젝트 및 Hyperledger Fabric을 참조하십시오.

IBM MQ Advanced for z/OS VUE 및 IBM Blockchain을 이미 사용 중인 경우, IBM MQ Bridge to blockchain을 사용하여 간단한 조회와 업데이트를 송신하고 블록체인 네트워크에서 응답을 수신할 수 있습니다. 이러한 방식으로 온프레미스 IBM 소프트웨어를 클라우드 블록체인 서비스와 통합할 수 있습니다.

브릿지 운영 프로세스에 대한 간략한 개요는 그림 1에서 볼 수 있습니다. 사용자 애플리케이션은 z/OS 큐 관리자의 입력/요청 큐에 JSON 형식의 메시지를 넣습니다. 브릿지는 큐 관리자에 연결하여 입력/요청 큐에서 메시지를 가져오고 JSON이 올바르게 형식화되었는지 확인한 후 조회 또는 업데이트를 블록체인에 발행합니다. 블록체인에서 리턴하는 데이터는 원래 IBM MQ 요청 메시지에 정의된 것과 같이, 브릿지를 통해 구문 분석된 후 응답 큐에 배치됩니다. 사용자 애플리케이션은 큐 관리자에 연결하여 응답 큐의 응답 메시지를 가져오며 정보를 사용할 수 있습니다.

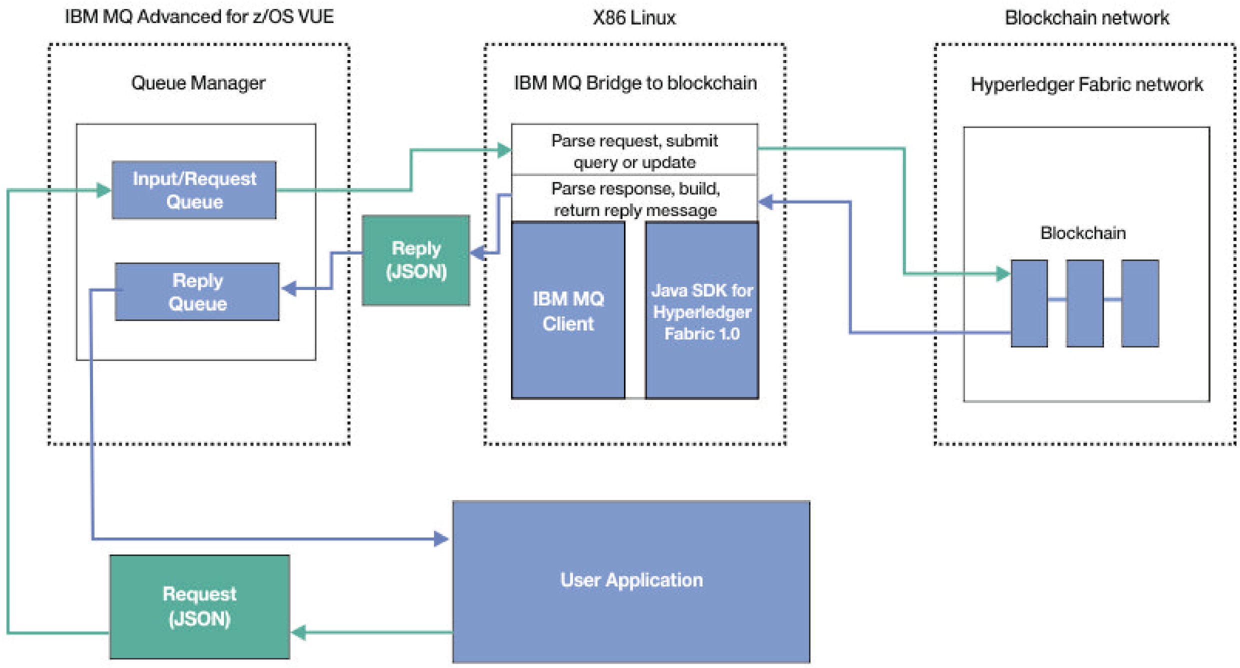


그림 *125. IBM MQ Bridge to blockchain*

IBM MQ Bridge to blockchain이 블록체인 네트워크에 참가자 또는 피어로 연결되도록 구성할 수 있습니다. 브릿지가 실행 중인 경우, 메시징 애플리케이션은 자원의 상태를 조회 또는 업데이트하여 그 결과를 메시징 애플리케이션에 응답으로 리턴하는 체인코드 루틴을 구동하도록 브릿지에 요청합니다.

프로시저

1. `csq4bcbq.jcl`에 정의된 브릿지용 오브젝트를 작성하십시오.

   샘플 브릿지 큐 정의는 사용자 신임 정보 및 브릿지에 대한 메시지 입력에 사용되는 이름 지정된 기본 큐 (`SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE` 및 `SYSTEM.BLOCKCHAIN.INPUT.QUEUE`)에 제공됩니다.

   a) `csq4bcbq.jcl`을 z/OS 데이터 세트로 복사하십시오.

   b) `csq4bcbq.jcl` 를 편집하여 z/OS 큐 관리자를 사용자 정의하십시오. 큐 관리자 이름과 IBM MQ 제품 라이브러리에 대한 상위 레벨 규정자를 제공해야 합니다. **APPL1** 브릿지 큐 예를 수정하거나 추가 애플리케이션에 대해 INPUT 및 REPLY 큐를 추가할 수 있습니다.

   c) `csq4bcbq.jcl`을 제출하여 정의한 오브젝트를 작성하십시오.

2. 원하는 방법을 사용하여 `x86download.tar.gz`를 `x86download` 디렉토리에서 x86 Linux 환경으로 전송하십시오.

   파일이 2진 모드로 전송되었는지 확인하십시오.

3. x86 Linux에서 `x86download.tar.gz`의 압축을 해제하십시오.

   ```
   tar -xvzf x86download.tar.gz
   ```

   압축 해제한 4개의 디렉토리는 `bin`, `lib`, `prereqs`, `samp`입니다.

4. IBM Java runtime environment 버전 8을 x86 Linux 환경으로 다운로드하십시오.

   a) IBM Java SDK Developer Centre Java 8 Downloads 페이지의 **Linux on AMD64/EMT64T** 섹션에서 **Installable package(InstallAnywhere as root)** 링크(파일 이름: `ibm-java-x86_64-jre-8.0-4.6.bin`)를 클릭하십시오.

      IBM SDK, Java Technology Edition, Version 8 라이센스 페이지가 표시됩니다.

   b) 계속하려면 라이센스에 동의하십시오.

      다운로드를 시작하려면 다운로드 창에서 **파일 저장**을 선택하십시오.

c) ibm-java-x86_64-jre-8.0-4.6.bin 파일을 실행하여 x86 Linux 환경에 설치하십시오.

기본 설치 위치는 /opt/ibm/ 디렉토리입니다.

```
./ibm-java-x86_64-jre-8.0-4.6.bin
```

d) 경로를 IBM 8 JRE로 설정하십시오.

```
export PATH=/opt/ibm/java-x86_64-80/jre/bin:$PATH
```

5. Fix Central에서 IBM MQ 9.0.4 Redistributable Java 클라이언트를 다운로드하십시오.
   a) 9.0.4.0-IBM-MQC-Redist LinuxX64 링크를 클릭하십시오.
   b) **브라우저를 사용하여 다운로드(HTTPS)**를 선택하십시오.
      계속을 클릭하십시오.
   c) 라이센스 이용 약관에 동의하십시오.
   d) 9.0.4.0-IBM-MQC-Redist-LinuxX64.tar.gz 링크를 클릭하고 **파일 저장**을 선택하여 다운로드하십시오.
   e) 9.0.4.0-IBM-MQC-Redist-LinuxX64.tar.gz를 x86 Linux 환경의 디렉토리로 압축 해제하십시오.
   f) 재배포 가능 Java 클라이언트를 압축 해제한 디렉토리로 경로를 설정하십시오.

```
export MQ_JAVA_INSTALL_PATH=/unpack_location/java
```

결과

You transferred the IBM MQ Bridge to blockchain from your z/OS to your x86 Linux environment, installed the IBM JRE 8, and the IBM MQ 9.0.4 Redistributable Java client.

다음에 수행할 작업

z/OS 큐 관리자 정보 및 블록체인 네트워크의 신임 정보를 사용하여 IBM MQ Bridge to blockchain용 구성 파일을 작성하십시오.

z/OS MQ Adv. VUE V 9.0.3 IBM MQ Bridge to blockchain용 구성 파일 작성

IBM MQ Bridge to blockchain용 구성 파일을 작성하여 IBM MQ 및 IBM Blockchain 네트워크에 연결하려면 큐 관리자와 블록체인 네트워크 매개변수를 입력하십시오.

시작하기 전에

- 블록체인 네트워크를 작성하고 구성했습니다.
- 블록체인 네트워크의 신임 정보 파일이 있습니다.
- IBM MQ Bridge to blockchain을 x86 Linux 환경에 설치했습니다.
- x86 Linux에 IBM MQ Bridge to blockchain, IBM MQ 9.0.4 Redistributable Java 클라이언트 및 IBM Java runtime environment 버전 8이 있습니다.

이 태스크 정보

이 태스크는 IBM MQ Bridge to blockchain 구성 파일을 작성하고 IBM Blockchain 및 IBM MQ 네트워크에 성공적으로 연결하는 데 필요한 최소 설정을 안내합니다.

브릿지를 사용하여 Hyperledger Fabric 1.0 architecture를 기반으로 하는 블록체인 네트워크에 연결할 수 있습니다. 브릿지를 사용하려면 블록체인 네트워크의 구성 정보가 필요합니다. 이 태스크의 각 단계에서는 서로 다르게 구성된 다음 두 블록체인 네트워크를 기반으로 하는 구성 세부사항 예를 찾을 수 있습니다.

- Docker에서 실행되는 Hyperledger Fabric 네트워크. 자세한 정보는 Hyperledger Fabric 시작하기, 첫 번째 애플리케이션 작성 및 585 페이지의 『Hyperledger Fabric 네트워크 신임 정보 파일 예』의 내용을 참조하십시오.

- IBM Cloud (formerly Bluemix)의 Kubernetes 클러스터에서 실행되는 Hyperledger Fabric 네트워크. 자세한 정보는 IBM Blockchain Platform에서 클라우드 샌드박스로 개발 및 587 페이지의 『Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예』의 내용을 참조하십시오.

모든 IBM MQ Bridge to blockchain 매개변수의 의미와 옵션에 대한 자세한 정보는 runmqbcb 명령을 참조하십시오. 자체 보안 요구사항을 고려하고 배치에 적합하게 매개변수를 사용자 정의해야 합니다.

프로시저

1. 브릿지를 실행하여 구성 파일을 작성하십시오.

   블록체인 네트워크 신임 정보 파일 및 z/OS 큐 관리자에서 제공하는 매개변수가 필요합니다. 709 페이지의 『블록체인과 함께 사용하도록 IBM MQ Advanced for z/OS VUE 구성』 태스크의 z/OS 환경에서 브릿지를 이동할 때 브릿지를 언팩한 위치의 bin 디렉토리에서 브릿지 스크립트를 실행하십시오.

   ```
   ./runmqbcb -o config_file_name.cfg
   ```

   다음 예에 표시된 것과 같이, 기존 값은 대괄호 안에 표시됩니다. 기존 값을 승인하려면 `Enter`를 누르고, 값을 지우려면 `Space`를 누른 후 `Enter`를 누르며, 새 값을 추가하려면 대괄호 안에 값을 입력한 후 `Enter`를 누르십시오. 값 목록(예: 피어)은 쉼표로 구분하거나 각 값을 줄바꿈하여 구분할 수 있습니다. 빈 행이 있으면 목록이 종료됩니다.

   참고: 기존 값은 편집할 수 없습니다. 대신 기존 값을 유지, 바꾸기 또는 지울 수는 있습니다.

2. z/OS 큐 관리자에 대한 연결 값을 입력하십시오.

   연결에 필요한 최소값은 큐 관리자 이름, 브릿지 입력 이름 및 정의한 ID 큐입니다. 리모트 큐 관리자에 대한 연결의 경우, **`MQ Channel`** 및 **`MQ Conname`** (큐 관리자가 실행 중인 호스트 주소 및 포트) 도 필요합니다. To use TLS for connecting to IBM MQ in step 713 페이지의 『6』, you must use JNDI or CCDT and specify **`MQ CCDT URL`** or **`JNDI implementation class`** and **`JNDI provider URL`** accordingly.

   ```
   Connection to Queue Manager
   ---------------------------
   Queue Manager                        : [z/OS_qmgr_name]
   Bridge Input Queue                   : [APPL1.BLOCKCHAIN.INPUT.QUEUE]
   Bridge User Identity Queue           : [SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE]
   MQ Channel                           : [SYSTEM.DEF.SVRCONN]
   MQ Conname                           : [host1.example.com(3714)]
   MQ CCDT URL                          : []
   JNDI implementation class            : []
   JNDI provider URL                    : []
   MQ Userid                            : []
   MQ Password                          : []
   ```

3. 블록체인 네트워크의 인증 기관에 대한 로그인 세부사항을 입력하십시오.

   로컬 Hyperledger Fabric 및 Kubernetes 클러스터 예제의 기본값은 **`Userid`** 의 경우 *admin* 이고 **`Enrollment Secret`**의 경우 *adminpw* 입니다. 블록체인 네트워크의 이 값을 변경한 경우 올바른 값을 사용하여 브릿지를 구성했는지 확인하십시오.

   ```
   Blockchain - User Identification
   --------------------------------
   Blockchain Userid                    : []admin
   Enrollment Secret                    : []******
   ```

4. 블록체인 네트워크에 대한 멤버십 및 ID 규칙을 제어하는 멤버십 서비스 제공자 ID(**MSPid**)를 입력하십시오.

   신임 정보 파일에서 **`Organisation Name`** 및 **`Organisation MSPId`**에 대한 **msp_id** 매개변수를 제공하십시오. 585 페이지의 『Hyperledger Fabric 네트워크 신임 정보 파일 예』에서 파일의 피어 섹션에서 **CORE_PEER_LOCALMSPID** 값을 사용하십시오. 587 페이지의 『Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예』에서는 **mspID** 값을 사용하십시오.

   ```
   Blockchain - Organisation Identification
   ----------------------------------------
   Organisation Name                    : []Org1MSP
   Organisation MSPId                   : []Org1MSP
   ```

5. 블록체인 네트워크 서버 위치 값을 입력하십시오.

585 페이지의 『Hyperledger Fabric 네트워크 신임 정보 파일 예』에서 인증서 권한, 피어 및 orderer 요소의 이름 및 서버: 포트 위치를 제공하십시오.

```
Blockchain server locations
---------------------------
Certificate Authority servers         : [ca.example.com Docker_container_host:7054] (for
example ca.example.com localhost:7054)
Peer servers                          : [peer0 localhost:7051]
Orderer servers                       : [orderer0 localhost:7050]
Peer Event servers                    : [peer0 localhost:7053]
Location of PEM file for Blockchain certificate : []
```

587 페이지의 『Kubernetes 컨테이너 클러스터 네트워크 구성 파일 예』에서 인증서 권한, 피어 및 orderer 요소의 이름 및 서버: 포트 위치를 제공하십시오.

```
Blockchain server locations
---------------------------
Certificate Authority servers         : [CA1
your_blockchain_network_public_ip_address:30000] (for example CA1 123.456.789.10:30000)
Peer servers                          : [blockchain-org1peer1
your_blockchain_network_public_ip_address:30110]
Orderer servers                       : [blockchain-orderer
your_blockchain_network_public_ip_address:31010]
Peer Event servers                    : [blockchain-org1peer1
your_blockchain_network_public_ip_address:30111]
Location of PEM file for Blockchain certificate : []
```

6. TLS 연결에 대한 인증 저장소 값을 입력하십시오.

브릿지는 큐 관리자에 연결되는 IBM MQ Java 클라이언트로 사용됩니다. 즉, TLS 보안을 사용하여 다른 IBM MQ Java 클라이언트와 동일한 방식으로 안전하게 연결하도록 구성할 수 있습니다. TLS 연결 구성 세부사항은 712 페이지의 『2』 단계에서 JNDI 또는 CCDT 정보를 지정한 경우에만 표시됩니다.

```
Certificate stores for TLS connections
-------------------------------------
Personal keystore                     : []
Keystore password                     : []
Trusted store for signer certs        : []
Trusted store password                : []
Use TLS for MQ connection             : [N]
Timeout for Blockchain operations     : [12]
```

7. IBM MQ Bridge to blockchain 로그 파일의 위치를 입력하십시오.

로그 파일 이름과 위치는 구성 파일 또는 명령행에 지정해야 합니다.

```
Behavior of bridge program
--------------------------
Runtime logfile for copy of stdout/stderr : [/var/mqm/errors/runmqbcb.log]
Done.
```

결과

IBM MQ Bridge to blockchain에서 IBM Blockchain 네트워크 및 IBM MQ z/OS 큐 관리자에 연결할 때 사용할 구성 파일을 작성했습니다.

다음에 수행할 작업

715 페이지의 『IBM MQ Bridge to blockchain 실행』을 위한 단계를 수행하십시오.

관련 정보

runmqbcb(IBM MQ Bridge to blockchain 실행)

z/OS V 9.0.3 IBM MQ Bridge to blockchain에서 사용할 큐에 대한 보안

IBM MQ Bridge to blockchain에서 사용하도록 정의된 z/OS 큐의 보안 구성을 위한 고려사항

다음 예에서는 IBM MQ Bridge to blockchain에 대한 큐의 보안을 설정하는 한 가지 방법을 보여주는 RACF 프로파일을 보여줍니다.

RESLEVEL

IBM MQ Bridge to blockchain은 **SVRCONN** 채널을 통해 **CHINIT**에 연결됩니다. 이때 브릿지 사용자가 사용하는 유효한 z/OS 사용자 ID에 대해 특정 보안 검사가 필요하다고 가정합니다. 즉, **CHINIT** 태스크에 대해 사용자 ID를 검사해야 합니다. **RESLEVEL** 프로파일에 대한 권한에 따라 단 한 개의 사용자 ID(채널 사용자 ID)만 검사할지 아니면 두 개의 사용자 ID(**channel** 사용자 ID와 **CHINIT** 사용자 ID)를 검사할지 여부가 결정됩니다. 예를 들면, 다음과 같습니다.

- 다음 코드는 **RESLEVEL** 프로파일에서 **CHINIT**에 **READ** 권한을 부여합니다. 따라서 **channel** 사용자 ID만 검사합니다.

  ```
  PERMIT RESLEVEL CLASS(MQADMIN) ID(CHINIT) ACCESS(READ)
  ```

- 다음 코드는 **RESLEVEL** 프로파일에서 **CHINIT**에 권한을 부여하지 않습니다. 따라서 두 개의 사용자 ID를 검사하며, **CHINIT** 사용자 ID에 추가 권한을 부여해야 합니다.

  ```
  PERMIT RESLEVEL CLASS(MQADMIN) ID(CHINIT) ACCESS(NONE)
  ```

다음 절에서는 추가 권한을 부여하는 코드 행이 강조표시되어 있습니다.

자세한 정보는 클라이언트 MQI 요청을 참조하십시오.

큐 자원 권한

ID 큐를 잠그고 브릿지 ID가 입력 및 출력에 이 큐를 사용하도록 허용하십시오.

```
RDEFINE MQQUEUE SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE UACC(NONE)
PERMIT SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE CLASS(MQQUEUE) ID(MQBBCART) ACCESS(UPDATE)
PERMIT SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE CLASS(MQQUEUE) ID(CHINIT) ACCESS(UPDATE)
```

브릿지 ID는 입력 큐를 열 수 있습니다.

```
DEF QL(CARTAX.BLOCKCHAIN.INPUT.QUEUE) LIKE(SYSTEM.BLOCKCHAIN.INPUT.QUEUE)
RDEFINE MQQUEUE CARTAX.BLOCKCHAIN.INPUT.QUEUE UACC(NONE)
PERMIT CARTAX.BLOCKCHAIN.INPUT.QUEUE CLASS(MQQUEUE) ID(MQBBCART) ACCESS(UPDATE)
PERMIT APPL1.BLOCKCHAIN.INPUT.QUEUE CLASS(MQQUEUE) ID(CHINIT) ACCESS(UPDATE)
```

APPCART 그룹의 애플리케이션 ID는 출력의 요청 큐를 열 수 있습니다.

```
PERMIT CARTAX.BLOCKCHAIN.INPUT.QUEUE CLASS(MQQUEUE) ID(APPCART) ACCESS(UPDATE)
```

애플리케이션 응답 큐를 포함할 프로파일

```
RDEFINE MQQUEUE CARTAX.APP.REPLY.** UACC(NONE)
```

APPCART 그룹의 애플리케이션 ID는 입력의 응답 큐를 열 수 있습니다.

```
RDEFINE MQADMIN CONTEXT.CARTAX.APP.REPLY.** UACC(NONE)
PERMIT CARTAX.APP.REPLY.** CLASS(MQQUEUE) ID(APPCART) ACCESS(UPDATE)
```

브릿지 ID는 출력의 응답 큐를 열고 **set_identity_context**를 사용하여 넣을 수 있습니다.

```
PERMIT CARTAX.APP.REPLY.** CLASS(MQQUEUE) ID(MQBBCART) ACCESS(UPDATE)
PERMIT CONTEXT.CARTAX.APP.REPLY.** CLASS(MQADMIN) ID(MQBBCART) ACCESS(UPDATE)
PERMIT CARTAX.APP.REPLY.** CLASS(MQQUEUE) ID(CHINIT) ACCESS(UPDATE)
PERMIT CONTEXT.CARTAX.APP.REPLY.** CLASS(MQADMIN) ID(CHINIT) ACCESS(UPDATE)
```

관련 태스크

718 페이지의 『IBM MQ Bridge to blockchain 클라이언트 샘플 실행』
IBM MQ Bridge to blockchain에서 제공하는 JMS 클라이언트 샘플을 사용하여 블록체인 브릿지가 검사하는 메시지를 입력 큐에 넣고 수신되는 응답을 확인할 수 있습니다.

관련 정보
큐 보안을 위한 프로파일
API 자원 보안 액세스 빠른 참조

z/OS MQ Adv. VUE V 9.0.3 IBM MQ Bridge to blockchain 실행

IBM MQ Bridge to blockchain를 실행하여 IBM Blockchain 및 IBM MQ에 연결합니다. 연결되면 브릿지가 조회 메시지를 처리하고, 이 메시지를 블록체인 네트워크로 송신하며, 응답을 수신하고 처리할 수 있습니다.

이 태스크 정보

이전의 태스크에서 작성한 구성 파일을 사용하여 IBM MQ Bridge to blockchain을 실행하십시오.

프로시저

1. 브릿지와 함께 사용할 z/OS 큐 관리자를 시작하십시오.
2. IBM MQ Bridge to blockchain을 시작하여 블록체인 네트워크와 z/OS 큐 관리자에 연결하십시오.
   709 페이지의 『블록체인과 함께 사용하도록 IBM MQ Advanced for z/OS VUE 구성』 태스크의 z/OS 환경에서 브릿지를 이동할 때 브릿지를 언팩한 위치의 bin 디렉토리에서 브릿지 스크립트를 실행하십시오.

   ```
   ./runmqbcb -f /config_file_location/config_file_name.cfg -r /log_file_location/logFile.log
   ```

   브릿지가 연결되면 다음과 유사한 출력이 리턴됩니다.

   ```
   Fri Oct 06 06:32:11 PDT 2017 IBM MQ Bridge to Blockchain
   5724-H72 (C) Copyright IBM Corp. 2017, 2023.

   Fri Oct 06 06:32:17 PDT 2017 Ready to process input messages.
   ```

3. 옵션: 브릿지를 실행한 후 리턴되는 메시지가 연결에 실패했음을 나타내는 경우 z/OS 큐 관리자 및 블록체인 네트워크에 대한 연결 문제점을 해결하십시오.
   a) 디버그 옵션 1을 사용하여 디버그 모드에서 명령을 실행하십시오.

   ```
   ./runmqbcb -f /config_file_location/config_file_name.cfg -r /log_file_location/
   logFile.log -d 1
   ```

   브릿지는 연결 설정을 스텝 스루하고 간결 모드로 처리 메시지를 표시합니다.
   b) 디버그 옵션 2를 사용하여 디버그 모드에서 명령을 실행하십시오.

   ```
   ./runmqbcb -f /config_file_location/config_file_name.cfg -r /log_file_location/
   logFile.log -d 2
   ```

   브릿지는 연결 설정을 스텝 스루하고 상세 모드로 처리 메시지를 표시합니다. 전체 출력이 로그 파일에 기록됩니다.

결과

IBM MQ Bridge to blockchain을 시작하고 큐 관리자와 블록체인 네트워크에 연결했습니다.

다음에 수행할 작업

- 718 페이지의 『IBM MQ Bridge to blockchain 클라이언트 샘플 실행』에 설명된 단계에 따라 조회 또는 업데이트 메시지를 형식화한 후 블록체인 네트워크로 송신하십시오.
- *MQBCB_EXTRA_JAVA_OPTIONS* 변수로 JVM 특성을 전달하여 IBM MQ 추적을 사용으로 설정하십시오. 자세한 정보는 IBM MQ Bridge to blockchain 추적을 참조하십시오.

z/OS MQ Adv. VUE V 9.0.3 *IBM MQ Bridge to blockchain*에 대한 메시지 형식

IBM MQ Bridge to blockchain에서 송신하고 수신한 메시지의 형식에 대한 정보입니다.

애플리케이션은 IBM MQ Bridge to blockchain이 블록체인에 보유된 정보의 조회 또는 업데이트를 수행하도록 요청합니다. 애플리케이션은 이를 위해 요청 메시지를 브릿지 요청 큐에 배치합니다. 조회 또는 업데이트 결과는 브릿지를 통해 응답 메시지로 형식화됩니다. 브릿지는 요청 메시지의 MQMD에서 **ReplyToQ** 및 **ReplyToQMgr** 필드에 포함된 정보를 응답 메시지의 목적지로 사용합니다.

브릿지에서 이용하고 생성하는 메시지는 JSON 형식의 텍스트(MQSTR) 메시지입니다. 입력 메시지는 단순 JSON이므로 프로그램에서는 문자열 연결을 사용하여 이를 생성할 수 있습니다. **args**를 제외한 모든 필드는 필수이므로, 해당 필드에 대한 인수 목록을 사용하려면 저장된 체인코드의 함수를 알고 있어야 합니다.

요청 메시지 형식

입력 메시지 형식:

```
{ "function": functionName,
    "channel" : chainName,
    "chaincodeName" : codeName,
    "args" : [ argument list]
}
```

Fabcar 샘플이 작동 중인 로컬 Hyperledger 네트워크 예

- 블록체인에 보유된 자동차 세부사항을 나타내는 JSON 오브젝트 목록을 리턴하는 Fabcar 체인코드에 `queryAllCars` 함수를 호출하는 조회 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

```
{ "function": "queryAllCars",
    "channel":"mychannel",
    "chaincodeName": "fabcar",
    "args":[]
}
```

응답 예:

```
{
   "statusCode": 200,
   "statusType": "SUCCESS",
   "message": "OK",
   "data": [
  {"Record":{"owner":"Tomoko","colour":"blue","model":"Prius","make":"Toyota"},"Key":"CAR0"},
  {"Record":{"owner":"Brad","colour":"red","model":"Mustang","make":"Ford"},"Key":"CAR1"},
  {"Record":{"owner":"Jin
Soo","colour":"green","model":"Tucson","make":"Hyundai"},"Key":"CAR2"},
  {"Record":
{"owner":"Max","colour":"yellow","model":"Passat","make":"Volkswagen"},"Key":"CAR3"},
  {"Record":{"owner":"Adriana","colour":"black","model":"S","make":"Tesla"},"Key":"CAR4"},
  {"Record":
{"owner":"Michel","colour":"purple","model":"205","make":"Peugeot"},"Key":"CAR5"},
  {"Record":{"owner":"Aarav","colour":"white","model":"S22L","make":"Chery"},"Key":"CAR6"},
  {"Record":{"owner":"Pari","colour":"violet","model":"Punto","make":"Fiat"},"Key":"CAR7"},
  {"Record":{"owner":"Valeria","colour":"indigo","model":"Nano","make":"Tata"},"Key":"CAR8"},
  {"Record":
{"owner":"Shotaro","colour":"brown","model":"Barina","make":"Holden"},"Key":"CAR9"}
]}
```

응답 메시지에는 블록체인에 현재 보유된 모든 자동차 레코드가 포함되어 있습니다.

- 블록체인 원장에 새 자동차 항목을 작성하는 fabcar 체인코드 예에서 `createCar` 함수를 작성하는 업데이트 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

```
{ "function":"createCar",
  "channel":"mychannel",
  "chaincodeName":"fabcar",
  "args":["CAR10", "Ford", "Mustang GT", "Blue", "Bob"]
}
```

응답 예:

```
{
   "statusCode": 200,
   "statusType": "SUCCESS",
```

```
    "message": "OK",
    "data": ""
}
```

블록체인에 새 자동차 항목이 추가되었는지 확인하려면 모든 자동차를 리턴하는 초기 메시지를 다시 사용하면 됩니다.

example02 데모가 작동 중인 Kubernetes 클러스터 네트워크 예

- 블록체인 원장 내에 *"a"* 엔티티에 대한 값을 리턴하는 example02 체인코드에 query 함수를 호출하는 조회 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

```
{ "function":"query",
  "channel":"channel1",
  "chaincodeName":"example02",
  "args":["a"]
}
```

응답 예:

```
{
   "statusCode": 200,
   "statusType": "SUCCESS",
   "message": "OK",
   "data": "100"
}
```

- 세 번째 인수에 지정된 값만큼 첫 번째 인수에 지정된 엔티티는 감소하고 두 번째 인수에 지정된 엔티티는 증분하는 example02 체인코드에 invoke 함수를 호출하는 메시지를 사용하려면 다음과 같이 메시지를 형식화하십시오.

```
{ "function":"invoke",
  "channel":"channel1",
  "chaincodeName":"example02",
  "args":["a", "b", "10"]
}
```

값은 다음과 같습니다.

  - 이전: a=100, b=200
  - 이후: a=90, b=210

응답 예:

```
{
   "statusCode": 200,
   "statusType": "SUCCESS",
   "message": "OK",
   "data": ""
}
```

새 값을 확인하려면 **"a"** 및 **"b"**의 값을 찾는 새 메시지 조회 메시지를 제출하십시오.

응답 메시지 형식

응답 메시지의 상관 ID는 인바운드 메시지의 메시지 ID로 설정됩니다. 사용자 특성은 입력에서 출력 메시지로 복사됩니다. 응답의 사용자 ID는 set-identity 컨텍스트를 통해 발신자의 사용자 ID로 설정됩니다.

성공적인 처리 예:

```
{ "data": "500", "message": "OK", "statusCode": 200, "statusType": "SUCCESS" }
```

이 메시지의 응답 데이터는 체인코드 응답(바이트가 UTF-8 문자열로 변환됨)에서 생성됩니다.

모든 오류 응답은 브릿지, 블록체인에 대한 호출 또는 체인코드 호출을 통해 생성되었는지 여부에 관계없이 동일한 필드를 갖습니다. 예를 들면, 다음과 같습니다.

- 잘못된 채널 이름

```
{
   "message": "Bad newest block expected status 200 got 404, Chain myUnknownChannel",
   "statusCode": 404,
   "statusType": "FAILURE"
}
```

- 잘못된 JSON 입력 메시지

```
{
   "message": "Error: Cannot parse message contents.",
   "statusCode": 2110,
   "statusType": "FAILURE"
}
```

- 잘못된 체인코드 매개변수

```
{
   "message": "Sending proposal to fabric-peer-1a failed because of gRPC
failure=Status{code=UNKNOWN, description={\"Error\":\"Nil amount for c\"}, cause=null}",
   "statusCode": 500,
   "statusType": "FAILURE"
}
```

애플리케이션은 **statusType** 문자열을 검토하거나 데이터 필드가 있는지 검토하여 요청이 성공 또는 실패했는지 여부를 알릴 수 있습니다. 입력 메시지를 처리할 때 오류가 있어 브릿지에서 블록체인으로 송신하지 못한 경우, 브릿지에서 리턴되는 값은 MQRC 값(보통 **MQRC_FORMAT_ERROR**)입니다.

z/OS MQ Adv. VUE V 9.0.3 *IBM MQ Bridge to blockchain* 클라이언트 샘플 실행

IBM MQ Bridge to blockchain에서 제공하는 JMS 클라이언트 샘플을 사용하여 블록체인 브릿지가 검사하는 메시지를 입력 큐에 넣고 수신되는 응답을 확인할 수 있습니다.

시작하기 전에

IBM MQ Bridge to blockchain이 실행 중이고 IBM MQ Advanced 큐 관리자 및 블록체인 네트워크에 연결되었으며 입력 메시지를 처리할 준비가 되었습니다.

이 태스크 정보

IBM MQ Bridge to blockchain의 samp 디렉토리에서 JMS 샘플 애플리케이션을 찾으십시오.

프로시저

1. 클라이언트 샘플 Java 소스 파일을 편집하십시오.

   샘플 지시사항에 따라 사용 중인 IBM MQ 환경 및 블록체인 네트워크와 일치하도록 샘플을 구성하십시오. 샘플의 다음 코드는 브릿지로 보낼 JSON 요청 메시지를 정의합니다.

```
        // Create the JSON request message.
        // Modify "query", "exampleBlockchainChannelName", and "exampleChaincodeName" to
        // match your deployed blockchain chaincode.
        // The "operation" field is optional, but recommended. It should be set to QUERY
        // or UPDATE to match what the chaincode is going to do.

       JSONObject inputMsg = new JSONObject();
        inputMsg.put("operation", "QUERY");

       inputMsg.put("function", "query");
        inputMsg.put("channel", "exampleBlockchainChannelName");
        inputMsg.put("chaincodeName","exampleChaincodeName");


       // Create the JSON arguments for the request message.
        // Modify "a" to match your deployed blockchain chaincode
        // requirements, and add further arguments as necessary

       JSONArray myArgs = new JSONArray();
```

```
        myArgs.add("a");
        inputMsg.put("args", myArgs);


        TextMessage message = session.createTextMessage(inputMsg.serialize());
        message.setJMSReplyTo(replyToQueue);
```

2. 샘플을 컴파일하십시오.

   브릿지 디렉토리에 제공된 IBM MQ 클라이언트 클래스 및 `JSON4j.jar` 파일을 지정하십시오.

```
javac -cp $MQ_JAVA_INSTALL_PATH/lib/*:../prereqs/JSON4J.jar SimpleBCBClient.java
```

3. 컴파일된 클래스를 실행하십시오.

```
java -cp $MQ_JAVA_INSTALL_PATH/lib/*:../prereqs/JSON4J.jar:. SimpleBCBClient
```

```

Starting Simple MQ Blockchain Bridge Client
Created the message. Starting the connection
Sent message:

 JMSMessage class: jms_text
  JMSType:          null
  JMSDeliveryMode:  2
  JMSDeliveryDelay: 0
  JMSDeliveryTime:  1508427559117
  JMSExpiration:    0
  JMSPriority:      4
  JMSMessageID:     ID:414d5120424342514d2020202020202009063e859ea36aa24
  JMSTimestamp:     1508427559117
  JMSCorrelationID: null
  JMSDestination:   queue:///APPL1.BLOCKCHAIN.INPUT.QUEUE
  JMSReplyTo:       queue:///APPL1.BLOCKCHAIN.REPLY.QUEUE
  JMSRedelivered:   false
    JMSXAppID: java
   JMSXDeliveryCount: 0
    JMSXUserID: USER1
   JMS_IBM_PutApplType: 6
    JMS_IBM_PutDate: 20171019
    JMS_IBM_PutTime: 15391912
{"args":
["a"],"function":"query","channel":"exampleBlockchainChannelName","operation":"QUERY","chainc
odeName":"exampleChaincodeName"}
```

응답 메시지:

```

  JMSMessage class: jms_text
  JMSType:          null
  JMSDeliveryMode:  1
  JMSDeliveryDelay: 0
  JMSDeliveryTime:  0
  JMSExpiration:    0
  JMSPriority:      4
  JMSMessageID:     ID:c3e2d840e2e2f0f84040404040404040d2afa27229838af2
  JMSTimestamp:     1497439784000
  JMSCorrelationID: ID:414d5120424342514d2020202020202009063e859ea36aa24  *(JMSMessageID of
the input message)
  JMSDestination:   null
  JMSReplyTo:       null
  JMSRedelivered:   false
    JMSXAppID: java
    JMSXDeliveryCount: 1
    JMSXUserID: USER1
    JMS_IBM_Character_Set: UTF-8
    JMS_IBM_Encoding: 273
    JMS_IBM_Format: MQSTR
    JMS_IBM_MsgType: 8
    JMS_IBM_PutApplType: 2
    JMS_IBM_PutDate: 20171019
    JMS_IBM_PutTime: 15392014
{
   "data": "20",
   "message": "OK",
   "statusCode": 200,
   "statusType": "SUCCESS"
```

```
}
Response text:
{
   "data": "20",
   "message": "OK",
   "statusCode": 200,
   "statusType": "SUCCESS"
}
SUCCESS
```

클라이언트가 응답을 기다리는 동안 제한시간 초과 오류를 수신할 경우 브릿지가 실행 중인지 확인하십시오.

주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다.

IBM은 다른 국가에서 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 한국 IBM 담당자에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 해당 IBM 제품, 프로그램 또는 서비스만을 사용할 수 있다는 것을 의미하지는 않습니다. IBM의 지적 재산권을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램 또는 서비스를 대신 사용할 수도 있습니다. 그러나 비IBM 제품, 프로그램 또는 서비스의 운영에 대한 평가 및 검증은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 특허 출원 중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 라이센스까지 부여하는 것은 아닙니다. 라이센스에 대한 의문사항은 다음으로 문의하십시오.

150-945
서울특별시 영등포구
국제금융로 10, 3IFC
한국 아이.비.엠 주식회사
U.S.A.

2바이트(DBCS) 정보에 관한 라이센스 문의는 한국 IBM에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

지적 재산권 라이센스 부여
2-31 Roppongi 3-chome, Minato-Ku
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

다음 단락은 현지법과 상충하는 영국이나 기타 국가에서는 적용되지 않습니다. IBM은 타인의 권리 비침해, 상품성 및 특정 목적에의 적합성에 대한 묵시적 보증을 포함하여(단, 이에 한하지 않음) 명시적 또는 묵시적인 일체의 보증 없이 이 책을 "현상태대로" 제공합니다. 일부 국가에서는 특정 거래에서 명시적 또는 묵시적 보증의 면책사항을 허용하지 않으므로, 이 사항이 적용되지 않을 수도 있습니다.

이 정보에는 기술적으로 부정확한 내용이나 인쇄상의 오류가 있을 수 있습니다. 이 정보는 주기적으로 변경되며, 변경된 사항은 최신판에 통합됩니다. IBM은 이 책에서 설명한 제품 및/또는 프로그램을 사전 통지 없이 언제든지 개선 및/또는 변경할 수 있습니다.

이 정보에서 언급되는 비IBM의 웹 사이트는 단지 편의상 제공된 것으로, 어떤 방식으로든 이들 웹 사이트를 옹호하고자 하는 것은 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로 해당 웹 사이트 사용으로 인한 위험은 사용자 본인이 감수해야 합니다.

IBM은 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 귀하가 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램과 기타 프로그램(본 프로그램 포함) 간의 정보 교환 및 (ii) 교환된 정보의 상호 이용을 목적으로 본 프로그램에 관한 정보를 얻고자 하는 라이센스 사용자는 다음 주소로 문의하십시오.

서울특별시 영등포구
서울특별시 강남구 도곡동 467-12,
군인공제회관빌딩
한국 아이.비.엠 주식회사
U.S.A.

이러한 정보는 해당 조건(예를 들면, 사용료 지불 등)하에서 사용될 수 있습니다.

이 정보에 기술된 라이센스가 부여된 프로그램 및 프로그램에 대해 사용 가능한 모든 라이센스가 부여된 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이센스 계약(IPLA) 또는 이와 동등한 계약에 따라 제공한 것입니다.

본 문서에 포함된 모든 성능 데이터는 제한된 환경에서 산출된 것입니다. 따라서 다른 운영 환경에서 얻어진 결과는 상당히 다를 수 있습니다. 일부 성능은 개발 단계의 시스템에서 측정되었을 수 있으므로 이러한 측정치가 일반적으로 사용되고 있는 시스템에서도 동일하게 나타날 것이라고는 보증할 수 없습니다. 또한 일부 성능은 추정을

통해 추측되었을 수도 있으므로 실제 결과는 다를 수 있습니다. 이 책의 사용자는 해당 데이터를 본인의 특정 환경에서 검증해야 합니다.

비IBM 제품에 관한 정보는 해당 제품의 공급업체, 공개 자료 또는 기타 범용 소스로부터 얻은 것입니다. IBM에서는 이러한 제품들을 테스트하지 않았으므로, 비IBM 제품과 관련된 성능의 정확성, 호환성 또는 기타 청구에 대해서는 확신할 수 없습니다. 비IBM 제품의 성능에 대한 의문사항은 해당 제품의 공급업체에 문의하십시오.

IBM이 제시하는 방향 또는 의도에 관한 모든 언급은 특별한 통지 없이 변경될 수 있습니다.

이 정보에는 일상의 비즈니스 운영에서 사용되는 자료 및 보고서에 대한 예제가 들어 있습니다. 이들 예제에는 개념을 가능한 완벽하게 설명하기 위하여 개인, 회사, 상표 및 제품의 이름이 사용될 수 있습니다. 이들 이름은 모두 가공의 것이며 실제 기업의 이름 및 주소와 유사하더라도 이는 전적으로 우연입니다.

이 정보를 소프트카피로 확인하는 경우에는 사진과 컬러 삽화가 제대로 나타나지 않을 수도 있습니다.

프로그래밍 인터페이스 정보

프로그래밍 인터페이스 정보는 본 프로그램과 함께 사용하기 위한 응용프로그램 소프트웨어 작성을 돕기 위해 제공됩니다.

이 책에는 고객이 프로그램을 작성하여 WebSphere MQ서비스를 얻을 수 있도록 하는 계획된 프로그래밍 인터페이스에 대한 정보가 포함되어 있습니다.

그러나 본 정보에는 진단, 수정 및 성능 조정 정보도 포함되어 있습니다. 진단, 수정 및 성능 조정 정보는 응용프로그램 소프트웨어의 디버그를 돕기 위해 제공된 것입니다.

중요사항: 이 진단, 수정 및 튜닝 정보는 변경될 수 있으므로 프로그래밍 인터페이스로 사용하지 마십시오.

상표

IBM, IBM 로고, ibm.com®는 전세계 여러 국가에 등록된 IBM Corporation의 상표입니다. 현재 IBM 상표 목록은 웹 "저작권 및 상표 정보"(www.ibm.com/legal/copytrade.shtml) 에 있습니다. 기타 제품 및 서비스 이름은 IBM 또는 타사의 상표입니다.

Microsoft 및 Windows는 미국 또는 기타 국가에서 사용되는 Microsoft Corporation의 상표입니다.

UNIX는 미국 또는 기타 국가에서 사용되는 The Open Group의 등록상표입니다.

Linux는 미국 또는 기타 국가에서 사용되는 Linus Torvalds의 등록상표입니다.

이 제품에는 Eclipse 프로젝트 (http://www.eclipse.org/) 에서 개발한 소프트웨어가 포함되어 있습니다.

Java 및 모든 Java 기반 상표와 로고는 Oracle 및/또는 그 계열사의 상표 또는 등록상표입니다.

부품 번호:

(1P) P/N: