

7.5

*IBM WebSphere MQ* 용 모니터링 및 성능

**IBM**

#### 참고

이 정보와 이 정보가 지원하는 제품을 사용하기 전에, [265 페이지의 『주의사항』](#)에 있는 정보를 확인하십시오.

This edition applies to version 7 release 5 of IBM® WebSphere® MQ and to all subsequent releases and modifications until otherwise indicated in new editions.

IBM은 귀하가 IBM으로 보낸 정보를 귀하의 권리를 침해하지 않는 범위 내에서 적절하다고 생각하는 방식으로 사용하거나 배포할 수 있습니다.

© Copyright International Business Machines Corporation 2007년, 2024.

---

# 목차

<b>모니터링 및 성능.....</b>	<b>5</b>
이벤트 모니터링.....	5
도구 이벤트.....	5
성능 이벤트.....	17
구성 이벤트.....	33
명령 이벤트.....	37
로거 이벤트.....	38
도구 이벤트를 모니터링하는 샘플 프로그램.....	44
메시지 모니터링.....	50
활동 및 조작.....	51
메시지 라우트 기술.....	52
활동 기록.....	54
라우트 추적 메시징.....	58
IBM WebSphere MQ 표시 라우트 애플리케이션.....	70
활동 보고서 참조.....	86
라우트 추적 메시지 참조.....	110
라우트 추적 응답 메시지 참조.....	119
회계 및 통계 메시지.....	122
회계 메시지.....	122
통계 메시지.....	125
회계 및 통계 정보 표시.....	129
회계 및 통계 메시지 참조.....	135
애플리케이션 활동 추적.....	180
애플리케이션 활동 추적 정보 수집.....	180
amqsact 샘플 프로그램.....	188
애플리케이션 활동 추적 메시지 참조.....	189
실시간 모니터링.....	252
실시간 모니터링을 제어하는 속성.....	253
큐와 채널 모니터링 데이터 표시.....	254
큐 모니터링.....	255
채널 모니터링.....	258
Windows 성능 모니터.....	264
<b>주의사항.....</b>	<b>265</b>
프로그래밍 인터페이스 정보.....	266
상표.....	266



## 모니터링 및 성능

큐 관리자 네트워크가 실행 중인 방법에 대한 통계 및 기타 특정 정보를 얻기 위해 IBM WebSphere MQ 에서 여러 모니터링 기술을 사용할 수 있습니다. 이 절의 모니터링 정보 및 지침을 사용하면 큐 관리자 네트워크의 성능을 향상시키는 데 도움이 될 수 있습니다.

큐 관리자 네트워크의 크기 및 복잡도에 따라 큐 관리자 네트워크의 모니터링으로부터 정보의 범위를 확보할 수 있습니다. 다음 목록에서는 큐 관리자 네트워크를 모니터링하는 이유에 대한 예를 제공합니다.

- 큐 관리자 네트워크에서 문제점을 감지합니다.
- 큐 관리자 네트워크에서 문제점의 원인을 판별하는 데 도움을 줍니다.
- 큐 관리자 네트워크의 효율성을 향상시킵니다.
- 큐 관리자 네트워크의 실행에 익숙해집니다.
- 큐 관리자 네트워크가 올바르게 실행 중인지 확인합니다.
- 특정 이벤트가 발생하면 메시지를 생성합니다.
- 메시지 활동을 기록합니다.
- 마지막으로 알려진 메시지 위치를 판별합니다.
- 실시간으로 큐 관리자 네트워크의 다양한 통계를 검사합니다.
- 감사 추적을 생성합니다.
- 애플리케이션 자원 사용법을 설명합니다.
- 용량 계획.

### 관련 태스크

#### 구성

[WebSphere MQ 관리](#)

## 이벤트 모니터링

이벤트 모니터링은 큐 관리자 네트워크에서 도구 이벤트 발생을 감지하는 프로세스입니다. 도구 이벤트는 큐 관리자나 채널 인스턴스에서 감지되는 이벤트의 논리적 조합입니다. 이러한 이벤트 때문에 큐 관리자나 채널 인스턴스가 이벤트 메시지라는 특정 메시지를 이벤트 큐에 넣게 됩니다.

IBM WebSphere MQ 도구 이벤트에서는 큐 관리자에서 오류, 경고 및 기타 중요한 발생에 대한 정보를 제공합니다. 이 이벤트를 사용하여 큐 관리자 네트워크에서 큐 관리자의 조작을 모니터링하여 다음 목표를 이룹니다.

- 큐 관리자 네트워크에서 문제점을 감지합니다.
- 큐 관리자 네트워크에서 문제점의 원인을 판별하는 데 도움을 줍니다.
- 감사 추적을 생성합니다.
- 큐 관리자 상태 변경에 반응합니다.

### 관련 참조

#### [이벤트 메시지 참조](#)

#### [7 페이지의 『이벤트 유형』](#)

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

#### [이벤트 메시지 형식](#)

## 도구 이벤트

도구 이벤트는 큐 관리자나 채널 인스턴스가 이벤트 큐에서 이벤트 메시지라는 특수 메시지를 감지하여 넣는 조건의 논리 결합입니다.

IBM WebSphere MQ 도구 이벤트에서는 큐 관리자에서 오류, 경고 및 기타 중요한 발생에 대한 정보를 제공합니다. 이러한 이벤트를 사용하여 큐 관리자의 조작을 모니터링할 수 있습니다 (다른 방법 (예: z/OS®의 경우 Tivoli® NetView®)).

6 페이지의 그림 1에서는 도구 이벤트의 개념을 설명합니다.

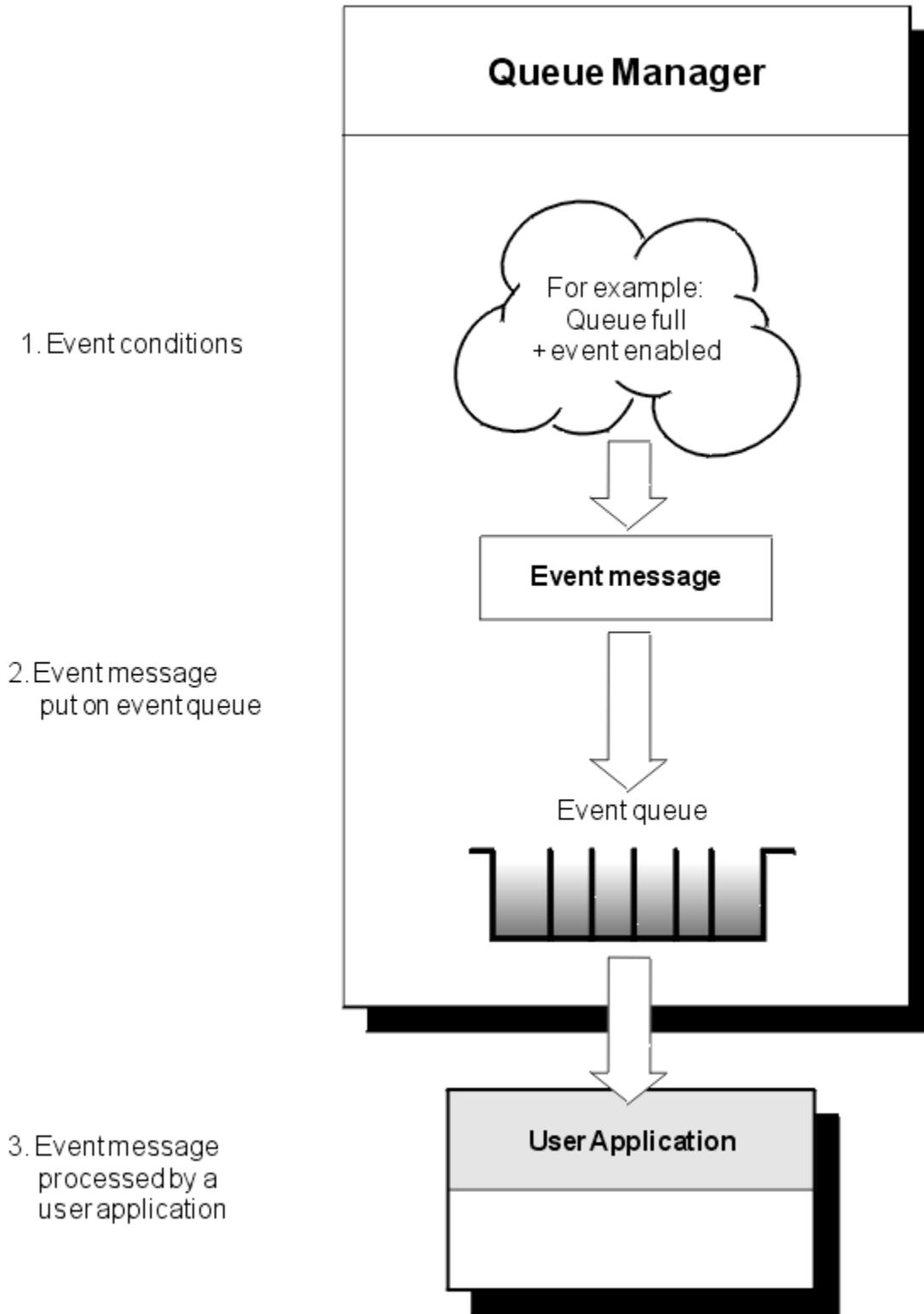


그림 1. 도구 이벤트 이해

## 이벤트 모니터링 애플리케이션

이벤트를 사용하여 큐 관리자를 모니터링하는 애플리케이션은 다음 요소를 포함해야 합니다.

1. 네트워크의 큐 관리자 간에 채널을 설정하십시오.
2. 필요한 데이터 변환을 구현하십시오. 데이터 변환의 일반 규칙이 적용됩니다. 예를 들어, z/OS 큐 관리자에서 UNIX 시스템 큐 관리자에서 이벤트를 모니터링하는 경우, EBCDIC을 ASCII로 변환해야 합니다.

## 이벤트 큐를 통해 이벤트 알림

적절한 이벤트 큐가 정의된 경우, 이벤트가 발생하면 큐 관리자는 이벤트 큐에 이벤트 메시지를 넣습니다(put). 이벤트 메시지는 다음 단계를 수행하는 적당한 MQI 애플리케이션 프로그램을 작성하여 검색할 수 있는 이벤트에 관한 정보를 포함합니다.

- 큐에서 메시지를 가져오십시오.
- 이벤트 데이터를 추출하도록 메시지를 처리하십시오.

관련 정보는 이벤트 메시지의 형식을 설명합니다.

## 이벤트를 발생시키는 조건

다음 목록은 도구 이벤트를 발생시킬 수 있는 조건 예제를 제공합니다.

- 큐에 있는 메시지 수에 대한 임계값 한계에 도달했습니다.
- 채널 인스턴스가 시작되거나 중지됩니다.
- 큐 관리자가 활성이 되거나 중지하도록 요청됩니다.
- An application tries to open a queue specifying a user ID that is not authorized on IBM WebSphere MQ for IBM i, Windows, UNIX and Linux® systems.
- 오브젝트가 작성, 삭제, 변경 또는 새로 고쳐집니다.
- MQSC 또는 PCF 명령이 실행됩니다.
- 큐 관리자는 새 로그 범위를 작성하기 시작합니다.
- 이벤트 조건이 충족되면 데드-레터 큐에 메시지를 넣습니다.

## 관련 개념

### 17 페이지의 『성능 이벤트』

성능 이벤트는 지정된 큐를 사용하는 애플리케이션의 성능에 영향을 줄 수 있는 조건과 관련됩니다. 성능 이벤트의 범위는 큐입니다. 한 큐에 대한 **MQPUT** 호출과 **MQGET** 호출은 다른 큐의 성능 이벤트 생성에 영향을 주지 않습니다.

### 44 페이지의 『도구 이벤트를 모니터링하는 샘플 프로그램』

도구 이벤트를 모니터링하기 위해 샘플 C 프로그램을 보려면 이 페이지를 사용하십시오.

## 이벤트 유형

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

IBM WebSphere MQ 인스트루먼테이션 이벤트의 유형은 다음과 같습니다.

- 큐 관리자 이벤트
- 채널 및 브릿지 이벤트
- 성능 이벤트
- 구성 이벤트
- 명령 이벤트
- 로거 이벤트
- 로컬 이벤트

각 큐 관리자에 대해 이벤트의 각 범주는 그 자신의 이벤트 큐를 갖습니다. 해당 범주의 모든 이벤트는 이벤트 메시지를 동일한 큐에 넣습니다(put).

**이벤트 큐:**

- SYSTEM.ADMIN.QMGR.EVENT
- SYSTEM.ADMIN.CHANNEL.EVENT
- SYSTEM.ADMIN.PERFM.EVENT
- SYSTEM.ADMIN.CONFIG.EVENT
- SYSTEM.ADMIN.COMMAND.EVENT
- SYSTEM.ADMIN.LOGGER.EVENT
- SYSTEM.ADMIN.PUBSUB.EVENT

**포함하는 메시지 출처:**

- 큐 관리자 이벤트
- 채널 이벤트
- 성능 이벤트
- 구성 이벤트
- 명령 이벤트
- 로거 이벤트

이벤트가 발행/구독에 관련되게 합니다. 멀티캐스트에서만 사용됩니다. 자세한 정보는 [멀티캐스트 애플리케이션 모니터링](#)을 참조하십시오.

인스트루멘테이션 이벤트를 사용자의 시스템 관리 애플리케이션에 통합하여 여러 큐 관리자, 여러 다른 노드 및 여러 IBM WebSphere MQ 애플리케이션에서 활동을 모니터링할 수 있습니다. 특히, 8 페이지의 그림 2에 표시된 것처럼 단일 노드 (IBM WebSphere MQ 이벤트를 지원하는 노드) 에서 시스템의 모든 노드를 모니터링할 수 있습니다.

도구 이벤트는 사용자 작성 보고 메커니즘을 통해 이벤트를 운영자에게 제시할 수 있는 관리 애플리케이션으로 보고될 수 있습니다.

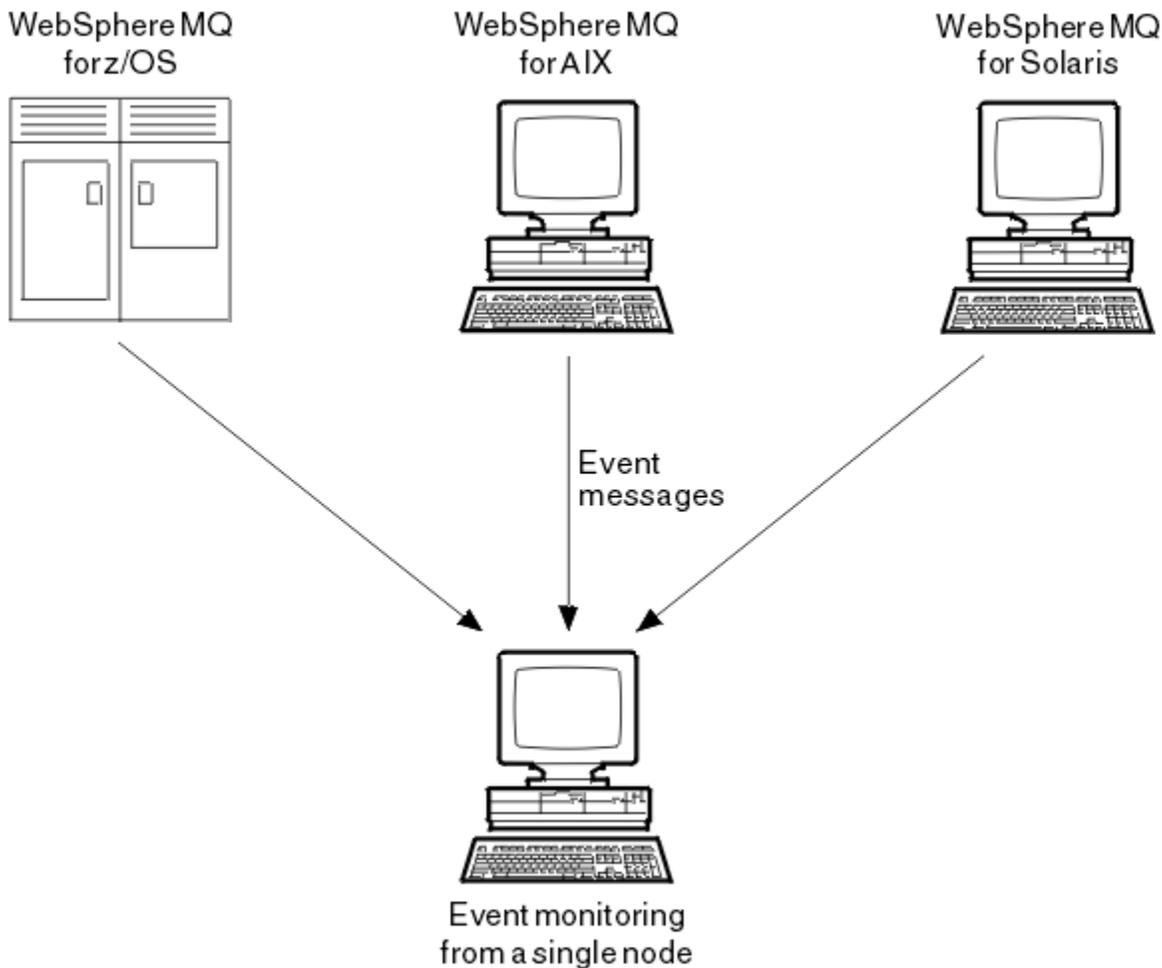


그림 2. 단일 노드에서 상이한 플랫폼 사이의 큐 관리자 모니터링

인스트루먼트이션 이벤트는 또한 다른 관리 네트워크 (예: z/OS의 Tivoli NetView) 에 대해 에이전트로 작동하는 애플리케이션이 보고서를 모니터하고 적절한 경보를 작성할 수 있도록 합니다.

## 큐 관리자 이벤트

큐 관리자 이벤트는 큐 관리자 내 자원 사용과 관련이 있습니다. 예를 들어, 애플리케이션이 존재하지 않는 큐에 메시지를 넣으려고 시도하는 경우에 큐 관리자 이벤트가 생성됩니다.

다음은 큐 관리자 이벤트가 발생할 수 있는 조건입니다.

- 애플리케이션이 실패하는 MQI 호출을 발행합니다. 호출의 이유 코드가 이벤트 메시지의 이유 코드와 동일합니다.

큐 관리자의 내부 조작 중에, 예를 들어 보고 메시지를 생성할 때 비슷한 조건이 발생할 수 있습니다. 이벤트 메시지의 이유 코드는 어떤 애플리케이션과도 연관되지 않은 경우에도 MQI 이유 코드와 일치할 수 있습니다. 이벤트 메시지 이유 코드가 MQI 이유 코드처럼 보이기 때문에 이벤트가 반드시 애플리케이션의 실패한 MQI 호출에 의해 유발되었다고 가정하지 마십시오.

- 명령이 큐 관리자에 발행되고 이 명령 처리가 이벤트를 유발합니다. 예를 들면, 다음과 같습니다.
  - 큐 관리자가 중지 또는 시작됩니다.
  - 명령에 대한 연관된 사용자 ID가 권한 부여되지 않는 명령이 발행됩니다.

WebSphere MQ 는 큐 관리자 이벤트에 대한 메시지를 SYSTEM.ADMIN.QMGR.EVENT 큐에 넣고 다음 큐 관리자 이벤트 유형을 지원합니다.

### 권한(Windows 및 UNIX 시스템만 해당)

권한 이벤트가 필요한 권한을 갖지 않는 큐를 열려고 시도하는 애플리케이션 또는 필수 권한을 갖지 않는 사용자 ID로부터 발행되는 명령 같은 권한 부여를 보고합니다. 권한 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [권한 부여되지 않음\(유형 1\)](#)
- [권한 부여되지 않음\(유형 2\)](#)
- [권한 부여되지 않음\(유형 3\)](#)
- [권한 부여되지 않음\(유형 4\)](#)
- [권한 부여되지 않음\(유형 5\)](#)
- [권한 부여되지 않음\(유형 6\)](#)

모든 권한 이벤트는 Windows 및 UNIX 시스템에서만 유효합니다.

### 금지

금지 이벤트는 큐가 Put 또는 Get에 대해 금지되는 큐 또는 토픽이 발행에 대해 금지되는 토픽에 대해 MQPUT 또는 MQGET 조작이 시도되었음을 표시합니다. 금지 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [Get 금지](#)
- [Put 금지](#)

### 로컬

로컬 이벤트는 애플리케이션(또는 큐 관리자)이 로컬 큐나 다른 로컬 오브젝트에 액세스할 수 없었음을 표시합니다. 예를 들어, 애플리케이션이 정의되지 않은 오브젝트에 액세스하려고 시도할 수 있습니다. 로컬 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [알리어스 기본 큐 유형 오류](#)
- [알 수 없는 알리어스 기본 큐](#)
- [알 수 없는 오브젝트 이름](#)

### 원격

리모트 이벤트는 애플리케이션이나 큐 관리자가 다른 큐 관리자의 리모트 큐에 액세스할 수 없음을 표시합니다. 예를 들어, 사용할 전송 큐가 올바르게 정의되지 않았을 수 있습니다. 원격 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [기본 전송 큐 유형 오류](#)

- [기본 전송 큐 사용 오류](#)
- [큐 유형 오류](#)
- [리모트 큐 이름 오류](#)
- [전송 큐 유형 오류](#)
- [전송 큐 사용 오류](#)
- [알 수 없는 기본 전송 큐](#)
- [알 수 없는 리모트 큐 관리자](#)
- [알 수 없는 전송 큐](#)

### 시작 및 중지

시작 및 중지 이벤트는 큐 관리자가 시작되었거나 시작 또는 일시정지하도록 요청되었음을 표시합니다.

z/OS는 시작 이벤트만 지원합니다.

중지 이벤트는 SYSTEM.ADMIN.QMGR.EVENT 큐의 기본 메시지 지속성이 지속적으로 정의되지 않는 한 보고되지 않습니다. 시작 및 중지 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [큐 관리자 활성화](#)
- [큐 관리자 비활성](#)

이 목록에 있는 각 이벤트 유형에 대해 이벤트 유형을 사용 가능 또는 사용 불가능하도록 큐 관리자 속성을 설정할 수 있습니다.

### 채널 및 브릿지 이벤트

채널은 이들 이벤트를 해당 조작 동안 감지되는 조건의 결과로 보고합니다. 예를 들어, 채널 인스턴스가 중지될 때입니다.

채널 이벤트는 다음 상황에 생성됩니다.

- 명령이 채널을 시작 또는 중지할 때.
- 채널 인스턴스가 시작 또는 중지할 때.
- 채널이 메시지를 가져올 때 변환 오류 경고를 수신할 때.
- 채널을 자동으로 작성하려고 시도할 때. 시도가 성공하든 아니면 실패하든 이벤트가 생성됩니다.

**참고:** 클라이언트 연결은 채널 시작 또는 채널 중지 이벤트를 유발하지 않습니다.

명령이 채널을 시작하는 데 사용될 때 이벤트가 생성됩니다. 채널 인스턴스가 시작할 때 또 다른 이벤트가 생성됩니다. 그러나 리스너, **runmqchl** 명령 또는 큐 관리자 트리거 메시지에 의한 채널 시작은 이벤트를 생성하지 않습니다. 이 경우 채널 인스턴스가 시작할 때만 이벤트가 생성됩니다.

성공적인 채널 시작 또는 중지 명령은 최소한 두 이벤트를 생성합니다. 이들 이벤트는 채널에 의해 연결되는 두 큐 관리자 모두에 대해(이벤트를 지원하는 경우) 생성됩니다.

채널 이벤트가 이벤트 큐에 놓여지는 경우, 오류 조건이 큐 관리자가 이벤트를 작성하게 합니다.

채널 및 브릿지 이벤트에 대한 이벤트 메시지는 SYSTEM.ADMIN.CHANNEL.EVENT 큐에 놓여집니다.

채널 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [채널이 활성화됨](#)
- [채널 자동 정의 오류](#)
- [채널 자동 정의 정상](#)
- [채널 변환 오류](#)
- [채널이 활성화되지 않음](#)
- [채널이 시작됨](#)
- [채널이 중지됨](#)
- [채널이 사용자에게 의해 중지됨](#)
- [채널이 차단됨](#)

## SSL 이벤트

유일한 Secure Sockets Layer(SSL 또는 TLS) 이벤트는 채널 SSL 오류 이벤트입니다. 이 이벤트는 SSL 또는 TLS 를 사용하는 채널이 SSL 연결을 구축하지 못할 때 보고됩니다.

SSL 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [채널 SSL 오류](#)
- [채널 SSL 경고](#)

## 성능 이벤트

성능 이벤트는 자원이 임계값 조건에 도달했다는 알림입니다. 예를 들어, 큐 용량 한계에 도달했습니다.

성능 이벤트는 지정된 큐를 사용하는 애플리케이션의 성능에 영향을 줄 수 있는 조건과 관련됩니다. 이벤트 큐 자체에 대해서는 생성되지 않습니다.

이벤트 유형은 메시지 데이터의 명령 ID 필드에서 리턴됩니다.

큐 관리자가 큐 관리자 이벤트나 성능 이벤트 메시지를 이벤트 큐에 넣으려고 시도하고 일반적으로 이벤트를 작성하는 오류가 감지되는 경우, 다른 이벤트는 작성되지 않고 조치가 취해지지 않습니다.

작업 단위 내의 MQGET 및 MQPUT 호출은 작업 단위가 커밋 또는 백아웃되는지 여부와 상관없이 성능 이벤트를 생성할 수 있습니다.

성능 이벤트에 대한 이벤트 메시지는 SYSTEM.ADMIN.PERFM.EVENT 큐에 넣어집니다.

다음 두 유형의 성능 이벤트가 있습니다.

### 큐 용량 이벤트

큐 용량 이벤트는 큐의 메시지 수, 즉 큐가 얼마나 가득 차거나 비었는지와 관련됩니다. 이들 이벤트는 공유 큐에 대해 지원됩니다. 큐 용량 이벤트 메시지는 다음 이벤트 데이터를 포함할 수 있습니다.

- [큐 용량 높음](#)
- [큐 용량 낮음](#)
- [큐 가득 참](#)

### 큐 서비스 간격 이벤트

큐 서비스 간격 이벤트는 사용자가 지정한 시간 간격 내에서 메시지가 처리되는지 여부와 관련됩니다. 이들 이벤트는 공유 큐에 대해 지원되지 않습니다.

## 구성 이벤트

구성 이벤트는 구성 이벤트가 명시적으로 요청될 때 또는 오브젝트가 작성, 수정 또는 삭제될 때 자동으로 생성됩니다.

구성 이벤트 메시지는 오브젝트의 속성에 관한 정보를 포함합니다. 예를 들어, 이름 목록 오브젝트가 작성되고 이름 목록 오브젝트의 속성에 관한 정보를 포함하는 경우 구성 이벤트 메시지가 생성됩니다.

구성 이벤트에 대한 이벤트 메시지는 SYSTEM.ADMIN.CONFIG.EVENT 큐에 넣어집니다.

4가지 유형의 구성 이벤트가 있습니다.

### 오브젝트 작성 이벤트

오브젝트 작성 이벤트는 오브젝트가 작성될 때 생성됩니다. 이벤트 메시지에는 [오브젝트 작성](#) 이벤트 데이터가 포함되어 있습니다.

### 오브젝트 변경 이벤트

오브젝트 변경 이벤트는 오브젝트가 변경될 때 생성됩니다. 이벤트 메시지에는 다음과 같은 이벤트 데이터가 포함되어 있습니다. [Change object](#).

### 오브젝트 삭제 이벤트

오브젝트 삭제 이벤트는 오브젝트가 삭제될 때 생성됩니다. 이벤트 메시지에는 다음과 같은 이벤트 데이터가 포함되어 있습니다. [Delete object](#)

### 오브젝트 새로 고치기 이벤트

오브젝트 새로 고치기 이벤트는 명시적인 새로 고치기 요청에 의해 생성됩니다. 이벤트 메시지에는 [오브젝트 새로 고치기](#) 이벤트 데이터가 포함되어 있습니다.

## 명령 이벤트

MQSC 또는 PCF 명령이 성공적으로 실행될 때 명령 이벤트가 보고됩니다.

명령 이벤트 메시지는 명령의 원본, 컨텍스트 및 콘텐츠에 관한 정보를 포함합니다. 예를 들어, 명령 이벤트 메시지는 MQSC 명령 ALTER QLOCAL이 성공적으로 실행하는 경우에 그런 정보와 함께 생성됩니다.

명령 이벤트에 대한 이벤트 메시지는 SYSTEM.ADMIN.COMMAND.EVENT 큐에 넣어집니다.

명령 이벤트에는 다음 이벤트 데이터가 포함됩니다. [Command](#).

## 로거 이벤트

로거 이벤트는 선형 로깅을 사용하는 큐 관리자가 새 로그 익스텐트에 로그 레코드 쓰기를 시작할 때 보고됩니다.

로거 이벤트 메시지는 매체 복원을 위해 또는 큐 관리자를 재시작하기 위해 큐 관리자에서 필요한 로그 익스텐트를 지정하는 정보를 포함합니다.

로거 이벤트에 대한 이벤트 메시지는 SYSTEM.ADMIN.LOGGER.EVENT 큐에 넣습니다.

로거 이벤트 메시지에 [Logger](#) 이벤트 데이터가 포함되어 있습니다.

## 이벤트 메시지 데이터 요약

이벤트 메시지의 각 유형이 포함할 수 있는 이벤트 데이터에 관한 정보를 얻으려면 이 요약을 사용하십시오.

이벤트 유형	다음 토픽 참조
권한 이벤트	<a href="#">권한 부여되지 않음(유형 1)</a>
	<a href="#">권한 부여되지 않음(유형 2)</a>
	<a href="#">권한 부여되지 않음(유형 3)</a>
	<a href="#">권한 부여되지 않음(유형 4)</a>
	<a href="#">권한 부여되지 않음(유형 5)</a>
	<a href="#">권한 부여되지 않음(유형 6)</a>
채널 이벤트	<a href="#">채널이 활성화됨</a>
	<a href="#">채널 자동 정의 오류</a>
	<a href="#">채널 자동 정의 정상</a>
	<a href="#">채널이 차단됨</a>
	<a href="#">채널 변환 오류</a>
	<a href="#">채널이 활성화되지 않음</a>
	<a href="#">채널이 시작됨</a>
	<a href="#">채널이 중지됨</a>
	<a href="#">채널이 사용자에게 의해 중지됨</a>
명령 이벤트	<a href="#">명령</a>
구성 이벤트	<a href="#">오브젝트 작성</a>
	<a href="#">변경 오브젝트</a>
	<a href="#">오브젝트 삭제</a>
	<a href="#">오브젝트 새로 고치기</a>
IMS 브릿지 이벤트	<a href="#">브릿지가 시작됨</a>
	<a href="#">브릿지가 중지됨</a>

이벤트 유형	다음 토픽 참조
금지 이벤트	<a href="#">Get 금지</a>
	<a href="#">Put 금지</a>
로컬 이벤트	<a href="#">알리어스 기본 큐 유형 오류</a>
	<a href="#">알 수 없는 알리어스 기본 큐</a>
	<a href="#">알 수 없는 오브젝트 이름</a>
로거 이벤트	<a href="#">로거</a>
성능 이벤트	<a href="#">큐 용량 높음</a>
	<a href="#">큐 용량 낮음</a>
	<a href="#">큐 가득 참</a>
	<a href="#">큐 서비스 간격 높음</a>
	<a href="#">큐 서비스 간격 정상</a>
원격 이벤트	<a href="#">기본 전송 큐 유형 오류</a>
	<a href="#">기본 전송 큐 사용 오류</a>
	<a href="#">큐 유형 오류</a>
	<a href="#">리모트 큐 이름 오류</a>
	<a href="#">전송 큐 유형 오류</a>
	<a href="#">전송 큐 사용 오류</a>
	<a href="#">알 수 없는 기본 전송 큐</a>
	<a href="#">알 수 없는 리모트 큐 관리자</a>
	<a href="#">알 수 없는 전송 큐</a>
SSL 이벤트	<a href="#">채널 SSL 오류</a>
시작 및 중지 이벤트	<a href="#">큐 관리자 활성화</a>
	<a href="#">큐 관리자 비활성</a>

## 이벤트 제어

이벤트의 유형에 따라서 큐 관리자, 큐 속성 또는 둘 다에 대한 적절한 값을 지정하여 이벤트를 사용 가능 또는 사용 불가능하게 설정할 수 있습니다.

생성되기 원하는 각 도구 이벤트를 사용 가능하게 설정해야 합니다. 예를 들어, 큐 가득 참 이벤트를 유발하는 조건은 다음과 같습니다.

- 큐 가득 참 이벤트가 지정된 큐에 대해 사용 가능하고,
- 애플리케이션이 해당 큐에 메시지를 넣기 위한 MQPUT 요청을 발행하지만, 큐가 가득 찼기 때문에 요청이 실패합니다.

다음 기술 중 하나를 사용하여 이벤트를 사용 가능 또는 사용 불가능하게 설정하십시오.

- IBM WebSphere MQ 스크립트 명령(MQSC)
- 해당되는 IBM WebSphere MQ PCF 명령
- IBM WebSphere MQ Explorer

**참고:** 명령에 의해서만 큐 및 큐 관리자 둘 다에 대한 이벤트와 관련된 속성을 설정할 수 있습니다. MQI 호출 MQSET은 이벤트와 관련된 속성을 지원하지 않습니다.

## 관련 개념

5 페이지의 『도구 이벤트』

도구 이벤트는 큐 관리자나 채널 인스턴스가 이벤트 큐에서 이벤트 메시지라는 특수 메시지를 감지하여 넣는 조건의 논리 결합입니다.

## 관련 태스크

관리 태스크 자동화

PCF(Programmable Command Format) 사용

## 관련 참조

7 페이지의 『이벤트 유형』

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

MQSC 명령

## 큐 관리자 이벤트 제어

큐 관리자 속성을 사용하여 큐 관리자 이벤트를 제어합니다. 큐 관리자 이벤트를 사용 가능하게 설정하려면 적절한 큐 관리자 속성을 ENABLED로 설정하십시오. 큐 관리자 이벤트를 사용 불가능하게 설정하려면 적절한 큐 관리자 속성을 DISABLED로 설정하십시오.

큐 관리자 이벤트를 사용 가능 또는 사용 불가능하게 설정하려면 MQSC 명령 ALTER QMGR을 사용하고 적절한 큐 관리자 속성을 지정하십시오. 14 페이지의 표 1 큐 관리자 이벤트를 사용 가능하게 하는 방법을 요약합니다. 큐 관리자 이벤트를 사용 불가능하게 설정하려면 적절한 매개변수를 DISABLED로 설정하십시오.

표 1. MQSC 명령을 사용하여 큐 관리자 이벤트 사용 가능	
이벤트	ALTER QMGR 매개변수
권한 금지 로컬 원격 시작 및 중지	AUTHOREV (ENABLED) INHIBTEV (ENABLED) LOCALEV (ENABLED) REMOTEEV (ENABLED) STRSTPEV (ENABLED)

## 채널 및 브릿지 이벤트 제어

큐 관리자 속성을 사용하여 채널 이벤트를 제어합니다. 채널 이벤트를 사용 가능하게 설정하려면 적절한 큐 관리자 속성을 ENABLED로 설정하십시오. 채널 이벤트를 사용 불가능하게 설정하려면 적절한 큐 관리자 속성을 DISABLED로 설정하십시오.

채널 이벤트를 사용 가능 또는 사용 불가능하게 설정하려면 MQSC 명령 ALTER QMGR을 사용하고 적절한 큐 관리자 속성을 지정하십시오. 14 페이지의 표 2 채널 및 브릿지 이벤트를 사용하는 방법을 요약합니다. 큐 관리자 이벤트를 사용 불가능하게 설정하려면 적절한 매개변수를 DISABLED로 설정하십시오.

표 2. MQSC 명령을 사용한 채널 및 브릿지 이벤트 사용 가능 설정	
이벤트	ALTER QMGR 매개변수
채널 채널 오류에만 관련됨 IMS BRIDGE SSL 채널 자동 정의	CHLEV (ENABLED) CHLEV (EXCEPTION) BRIDGEEV (ENABLED) SSLEV (ENABLED) CHADEV(ENABLED)

CHLEV가 예외로 설정될 때, 다음 리턴 코드 및 대응하는 이유 규정자가 생성됩니다.

- MQRC\_CHANNEL\_ACTIVATED
- MQRC\_CHANNEL\_CONV\_ERROR
- MQRC\_CHANNEL\_NOT\_ACTIVATED
- MQRC\_CHANNEL\_STOPPED

- 다음 ReasonQualifiers를 가짐:
  - MQRQ\_CHANNEL\_STOPPED\_ERROR
  - MQRQ\_CHANNEL\_STOPPED\_RETRY
  - MQRQ\_CHANNEL\_STOPPED\_DISABLED
- MQRQ\_CHANNEL\_STOPPED\_BY\_USER
- MQRQ\_CHANNEL\_BLOCKED
  - 다음 ReasonQualifiers를 가짐:
    - MQRQ\_CHANNEL\_BLOCKED\_NOACCESS
    - MQRQ\_CHANNEL\_BLOCKED\_USERID
    - MQRQ\_CHANNEL\_BLOCKED\_ADDRESS

## 성능 이벤트 제어

PERFMEV 큐 관리자 속성을 사용하여 성능 이벤트를 제어합니다. 성능 이벤트를 사용 가능하게 설정하려면 PERFMEV를 ENABLED로 설정하십시오. 성능 이벤트를 사용 불가능하게 설정하려면 PERFMEV 큐 관리자 속성을 DISABLED로 설정하십시오.

PERFMEV 큐 관리자 속성을 ENABLED로 설정하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR PERFMEV (ENABLED)
```

특정 성능 이벤트를 사용 가능으로 설정하려면 적절한 큐 속성을 설정하십시오. 또한, 이벤트를 유발하는 조건을 지정하십시오.

## 큐 용량 이벤트

기본적으로, 모든 큐 용량 이벤트는 사용 불가능합니다. 큐 용량 이벤트 중 하나에 대해 큐를 구성하려면,

1. 큐 관리자에서 성능 이벤트를 사용 가능하게 설정하십시오.
2. 필수 큐에서 이벤트를 사용 가능하게 설정하십시오.
3. 필요한 경우 최대 큐 용량의 백분율로 표현되는 한계를 적절한 수준으로 설정하십시오.

## 큐 서비스 간격 이벤트

큐 서비스 간격 이벤트에 대해 큐를 구성하려면 다음을 수행해야 합니다.

1. 큐 관리자에서 성능 이벤트를 사용 가능하게 설정하십시오.
2. 필요한 대로 큐에서 큐 서비스 간격 높음 또는 정상 이벤트에 대한 제어 속성을 설정하십시오.
3. 큐에 대한 QSVCINT 속성을 적절한 시간으로 설정하여 서비스 간격 시간을 지정하십시오.

**참고:** 사용 가능할 때 큐 서비스 간격 이벤트가 큐에 대한 MQI 호출이 발행될 때까지 반드시 기다리지 않고 적절한 시간에 생성될 수 있습니다. 그러나, MQI 호출이 메시지를 넣거나 제거하기 위해 큐에서 사용되는 경우 모든 적용 가능한 성능 이벤트가 해당 시간에 생성됩니다. 경과 시간이 서비스 간격 시간과 같아질 때는 이벤트가 생성되지 않습니다.

## 구성, 명령 및 로거 이벤트 제어

큐 관리자 속성 CONFIGEV, CMDEV 및 LOGGEREV를 사용하여 구성, 명령 및 로거 이벤트를 제어합니다. 이들 이벤트를 사용 가능하게 설정하려면 적절한 큐 관리자 속성을 ENABLED로 설정하십시오. 이들 이벤트를 사용 불가능하게 설정하려면 적절한 큐 관리자 속성을 DISABLED로 설정하십시오.

## 구성 이벤트

구성 이벤트를 사용 가능하게 설정하려면 CONFIGEV를 ENABLED로 설정하십시오. 구성 이벤트를 사용 불가능하게 설정하려면 CONFIGEV를 DISABLED로 설정하십시오. 예를 들어, 다음 MQSC 명령을 사용하여 구성 이벤트를 사용 가능으로 설정할 수 있습니다.

```
ALTER QMGR CONFIGEV (ENABLED)
```

## 명령 이벤트

명령 이벤트를 사용 가능하게 설정하려면 CMDEV를 ENABLED로 설정하십시오. DISPLAY MQSC 명령 및 조회 PCF 명령을 제외한 명령에 대한 명령 이벤트를 사용 가능하게 설정하려면 CMDEV를 NODISPLAY로 설정하십시오. 명령 이벤트를 사용 불가능하게 설정하려면 CMDEV를 DISABLED로 설정하십시오. 예를 들어, 다음 MQSC 명령을 사용하여 명령 이벤트를 사용 가능으로 설정할 수 있습니다.

```
ALTER QMGR CMDEV (ENABLED)
```

## 로거 이벤트

로거 이벤트를 사용 가능하게 설정하려면 LOGGEREV를 ENABLED로 설정하십시오. 로거 이벤트를 사용 불가능하게 설정하려면 LOGGEREV를 DISABLED로 설정하십시오. 예를 들어, 다음 MQSC 명령을 사용하여 로거 이벤트를 사용 가능으로 설정할 수 있습니다.

```
ALTER QMGR LOGGEREV(ENABLED)
```

## 이벤트 큐

이벤트가 발생할 때 큐 관리자는 이벤트 메시지를 정의된 이벤트 큐에 넣습니다. 이벤트 메시지가 이벤트에 관한 정보를 포함합니다.

이벤트 큐를 로컬 큐, 알리어스 큐 또는 리모트 큐의 로컬 정의로 정의할 수 있습니다. 모든 이벤트 큐를 한 큐 관리자의 동일한 리모트 큐의 로컬 정의로 정의하는 경우, 모니터링 활동을 중앙 집중할 수 있습니다.

이벤트 메시지가 전송 큐에 필요한 메시지 형식과 호환되지 않는 형식을 갖기 때문에 이벤트 큐를 전송 큐로 정의해서는 안 됩니다.

공유 이벤트 큐는 QSGDISP(SHARED) 값을 갖고 정의되는 로컬 큐입니다.

## 이벤트 큐가 사용 불가능할 때

이벤트 큐가 사용 불가능할 때 이벤트가 발생하는 경우 이벤트 메시지가 유실됩니다. 예를 들어, 이벤트의 범주에 대한 이벤트 큐를 정의하지 않은 경우 해당 범주에 대한 모든 이벤트 메시지가 유실됩니다. 예를 들어 이벤트 메시지가 데드 레터(미전달 메시지) 큐에서 저장되지 않습니다.

그러나, 이벤트 큐를 리모트 큐로 정의할 수 있습니다. 그런 다음, 해석된 큐에 메시지를 넣는 원격 시스템에 문제가 있는 경우 이벤트 메시지는 원격 시스템의 데드-레터 큐에 도달합니다.

이벤트 큐는 다음을 포함한 많은 여러 가지 이유 때문에 사용 불가능할 수 있습니다.

- 큐가 정의되지 않았습니다.
- 큐가 삭제되었습니다.
- 큐가 가득 찼습니다.
- 큐가 put 금지되었습니다.

이벤트 큐의 부재가 이벤트가 발생하는 것을 막지 않습니다. 예를 들어, 성능 이벤트 후에 큐 관리자는 큐 속성을 변경하고 큐 통계를 재설정합니다. 이 변경은 이벤트 메시지가 성능 이벤트 큐에 놓여지는지 여부와 상관없이 발생합니다. 구성 및 명령 이벤트의 경우에도 동일하게 적용됩니다.

## 트리거된 이벤트 큐 사용

이벤트가 생성될 때 이벤트 큐에 놓여지는 이벤트 메시지가 사용자가 작성한 모니터링 애플리케이션을 시작하도록 트리거를 갖는 이벤트 큐를 설정할 수 있습니다. 이 애플리케이션은 이벤트 메시지를 처리하고 적절한 조치를 취할 수 있습니다. 예를 들어, 특정 이벤트는 운영자에게 알려져야 할 수 있으며, 다른 이벤트는 일부 관리 태스크를 자동으로 수행하는 애플리케이션을 시작할 수 있습니다.

이벤트 큐는 그들과 연관된 트리거 조치를 가질 수 있으며 트리거 메시지를 작성할 수 있습니다. 그러나, 이들 트리거 메시지가 다시 정상적으로 이벤트를 생성하는 조건을 유발하는 경우 이벤트가 생성되지 않습니다. 이 인스턴스에서 이벤트를 생성하지 않는 것은 루핑이 발생하지 않도록 보장합니다.

## 관련 개념

### [13 페이지의 『이벤트 제어』](#)

이벤트의 유형에 따라서 큐 관리자, 큐 속성 또는 둘 다에 대한 적절한 값을 지정하여 이벤트를 사용 가능 또는 사용 불가능하게 설정할 수 있습니다.

### [17 페이지의 『이벤트 메시지의 형식』](#)

이벤트 메시지는 이벤트 및 그의 원인에 관한 정보를 포함하고 있습니다. 기타 WebSphere MQ 메시지와 같이 이벤트 메시지도 메시지 디스크립터 및 메시지 데이터라는 두 부분을 가집니다.

### [트리거 이벤트의 조건](#)

## 관련 참조

[QSGDisp \(MQLONG\)](#)

## 이벤트 메시지의 형식

이벤트 메시지는 이벤트 및 그의 원인에 관한 정보를 포함하고 있습니다. 기타 WebSphere MQ 메시지와 같이 이벤트 메시지도 메시지 디스크립터 및 메시지 데이터라는 두 부분을 가집니다.

- 메시지 디스크립터는 MQMD 구조를 기반으로 합니다.
- 메시지 데이터는 이벤트 헤더와 이벤트 데이터로 구성됩니다. 이벤트 헤더에는 이벤트 유형을 식별하는 이유 코드가 들어 있습니다. 이벤트 메시지 넣기 및 모든 후속 조치는 해당 이벤트를 유발한 MQI 호출에 의해 리턴되는 이유 코드에 영향을 주지 않습니다. 이벤트 데이터는 이벤트에 관한 추가 정보를 제공합니다.

일반적으로 시스템 관리 애플리케이션이 실행하는 엔터프라이즈의 요구사항에 맞게 조정된 해당 애플리케이션으로 이벤트 메시지를 처리합니다.

큐 공유 그룹의 큐 관리자가 이벤트 메시지 생성을 위한 조건을 감지할 때, 여러 큐 관리자가 공유 큐에 대한 이벤트 메시지를 생성할 수 있으므로 여러 개의 이벤트 메시지가 생성됩니다. 시스템이 여러 큐 관리자의 다중 이벤트 메시지를 상관시킬 수 있도록 하기 위해, 이들 이벤트 메시지는 메시지 디스크립터(MQMD)에서 설정된 고유한 상관 ID(*CorrelId*)를 갖고 있습니다.

## 관련 참조

### [88 페이지의 『활동 보고서 MQMD\(메시지 디스크립터\)』](#)

이 페이지를 사용하여 활동 보고서의 MQMD 구조에 따라 포함된 값을 봅니다.

### [92 페이지의 『활동 보고서 MQEPH\(임베드된 PCF 헤더\)』](#)

이 페이지를 사용하여 활동 보고서의 MQEPH 구조에 따라 포함된 값을 봅니다.

### [93 페이지의 『활동 보고서 MQCFH\(PCF 헤더\)』](#)

이 페이지를 사용하여 활동 보고서의 MQCFH 구조에 따라 포함된 PCF 값을 봅니다.

### [이벤트 메시지 참조](#)

### [이벤트 메시지 형식](#)

### [이벤트 메시지 MQMD\(메시지 디스크립터\)](#)

### [이벤트 메시지 MQCFH\(PCF 헤더\)](#)

### [이벤트 메시지 설명](#)

## 성능 이벤트

성능 이벤트는 지정된 큐를 사용하는 애플리케이션의 성능에 영향을 줄 수 있는 조건과 관련됩니다. 성능 이벤트의 범위는 큐입니다. 한 큐에 대한 **MQPUT** 호출과 **MQGET** 호출은 다른 큐의 성능 이벤트 생성에 영향을 주지 않습니다.

성능 이벤트 메시지는 반드시 큐에 대한 MQI 호출이 발행될 때까지 기다릴 필요없이 임의의 적절한 시간에 생성될 수 있습니다. 그러나 메시지를 넣거나 제거하기 위해 큐에서 MQI 호출을 사용하는 경우 해당 시간에 적절한 성능 이벤트가 생성됩니다.

생성되는 모든 성능 이벤트 메시지는 SYSTEM.ADMIN.PERFM.EVENT 큐에 위치합니다.

이벤트 데이터는 이벤트의 원인을 식별하는 이유 코드, 성능 이벤트 통계 세트 및 기타 데이터를 포함합니다. 성능 이벤트 메시지에서 리턴될 수 있는 이벤트 데이터의 유형이 다음 목록에서 설명됩니다.

- [큐 용량 높음](#)

- [큐 용량 낮음](#)
- [큐 가득 참](#)
- [큐 서비스 간격 높음](#)
- [큐 서비스 간격 정상](#)

성능 이벤트의 사용을 설명하는 예제에서는 적절한 IBM WebSphere MQ 명령 (MQSC) 을 사용하여 큐 속성을 설정한다고 가정합니다. 에서 큐 관리자에 대한 조작 및 제어 패널을 사용하여 큐 속성을 설정할 수도 있습니다.

#### 관련 참조

[7 페이지의 『이벤트 유형』](#)

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

## 성능 이벤트 통계

이벤트 메시지의 성능 이벤트 데이터에는 이벤트에 관한 통계가 들어 있습니다. 지정된 큐의 작동을 분석하려면 이 통계를 사용하십시오.

이벤트 메시지의 이벤트 데이터에는 시스템 관리 프로그램에 대한 이벤트의 정보가 들어 있습니다. 모든 성능 이벤트의 경우 이벤트 데이터는 큐 관리자의 이름과 이벤트와 연관된 큐를 포함합니다. 이벤트 데이터는 또한 이벤트와 관련된 통계를 포함합니다. [18 페이지의 표 3](#)에서는 큐의 작동을 분석하기 위해 사용할 수 있는 이벤트 통계를 요약합니다. 모든 통계는 통계가 마지막으로 재설정된 이후에 발행한 이벤트를 참조합니다.

표 3. 성능 이벤트 통계	
매개변수	설명
TimeSinceReset	통계가 마지막으로 재설정된 이후 경과 시간.
HighQDepth	통계가 마지막으로 재설정된 이후 큐에 있는 메시지의 최대 수.
MsgEnqCount	통계가 마지막으로 재설정된 이후, 큐에 들어간 메시지 수(큐에 대한 MQPUT 호출의 수).
MsgDeqCount	통계가 마지막으로 재설정된 이후, 큐에서 제거된 메시지 수(큐에 대한 MQGET 호출의 수).

성능 이벤트 통계는 다음 변경 중 하나가 발생할 때 재설정됩니다.

- 성능 이벤트가 발생합니다(통계가 모든 활성 큐 관리자에서 재설정됩니다).
- 큐 관리자가 중지하고 재시작합니다.
- PCF 명령인 Reset Queue Statistics가 애플리케이션 프로그램에서 실행됩니다.

#### 관련 개념

[17 페이지의 『성능 이벤트』](#)

성능 이벤트는 지정된 큐를 사용하는 애플리케이션의 성능에 영향을 줄 수 있는 조건과 관련됩니다. 성능 이벤트의 범위는 큐입니다. 한 큐에 대한 MQPUT 호출과 MQGET 호출은 다른 큐의 성능 이벤트 생성에 영향을 주지 않습니다.

[20 페이지의 『서비스 타이머』](#)

큐 서비스 간격 이벤트는 서비스 타이머라고 부르는 내부 타이머를 사용하는데, 이것은 큐 관리자에 의해 제어됩니다. 서비스 타이머는 큐 서비스 간격 이벤트가 사용 가능한 경우에만 사용됩니다.

[20 페이지의 『큐 서비스 간격 이벤트에 대한 규칙』](#)

공식 규칙이 서비스 타이머가 설정되고 큐 서비스 간격 이벤트가 생성되는 시기를 제어합니다.

#### 관련 태스크

[21 페이지의 『큐 서비스 간격 이벤트 사용 가능』](#)

큐 서비스 간격 이벤트에 대해 큐를 구성하려면 적절한 큐 관리자 및 큐 속성을 설정합니다.

#### 관련 참조

[큐 용량 상한](#)

## 큐 서비스 간격 이벤트

큐 서비스 간격 이벤트는 서비스 간격이라고 부르는 사용자 정의 시간 간격 안에서 큐에 대해 조작이 수행되었는지 여부를 표시합니다. 설치에 따라서, 큐 서비스 간격 이벤트를 사용하여 메시지가 충분히 빨리 큐에서 제거되는지 여부를 모니터링할 수 있습니다.

큐 서비스 간격 이벤트는 공유 큐에서 지원되지 않습니다.

다음 유형의 큐 서비스 간격 이벤트가 발생할 수 있는데, 여기서 *Get* 조작이라는 용어는 **MQGET** 호출 또는 큐에서 메시지를 제거하는 활동(예: **CLEAR QLOCAL** 명령 사용)을 의미합니다.

### 큐 서비스 간격 확인

다음 조작 중 하나 후에

- MQPUT 호출
- 비어 있지 않은 큐를 떠나는 Get 조작

서비스 간격으로 알려진 사용자 정의 기간 안에 Get 조작이 수행되었음을 표시합니다.

Get 조작만이 큐 서비스 간격 정상 이벤트 메시지를 유발할 수 있습니다. 큐 서비스 간격 정상 이벤트를 때로는 정상 이벤트라고 설명합니다.

### 큐 서비스 간격 높음

다음 조작 중 하나 후에

- MQPUT 호출
- 비어 있지 않은 큐를 떠나는 Get 조작

사용자가 정의한 서비스 간격 내에 Get 조작이 수행되지 **않았습니다**.

Get 조작이나 MQPUT 호출 중 하나가 큐 서비스 간격 높음 이벤트 메시지를 유발할 수 있습니다. 큐 서비스 간격 높음 이벤트는 때로는 높음 이벤트로 설명됩니다.

큐 서비스 간격 정상 및 큐 서비스 간격 높음 이벤트를 둘 다 사용으로 설정하려면 `QServiceIntervalEvent` 제어 속성을 `High`로 설정하십시오. 큐 서비스 간격 정상 이벤트는 큐 서비스 간격 높음 이벤트가 생성될 때 자동으로 사용 가능하게 됩니다. 큐 서비스 간격 정상 이벤트를 독립적으로 사용 가능하게 설정할 필요가 없습니다.

정상 및 높음 이벤트는 상호 배타적이므로, 하나가 사용 가능하면 다른 하나는 사용 불가능합니다. 그러나 두 이벤트가 모두 동시에 사용 불가능할 수 있습니다.

19 페이지의 그림 3에서는 큐 용량 대 시간의 그래프를 보여줍니다. 시간 P1에서 애플리케이션이 큐에 메시지를 넣기 위해 MQPUT을 실행합니다. 시간 G1에 또 다른 애플리케이션이 큐에서 메시지를 제거하기 위해 MQGET을 실행합니다.

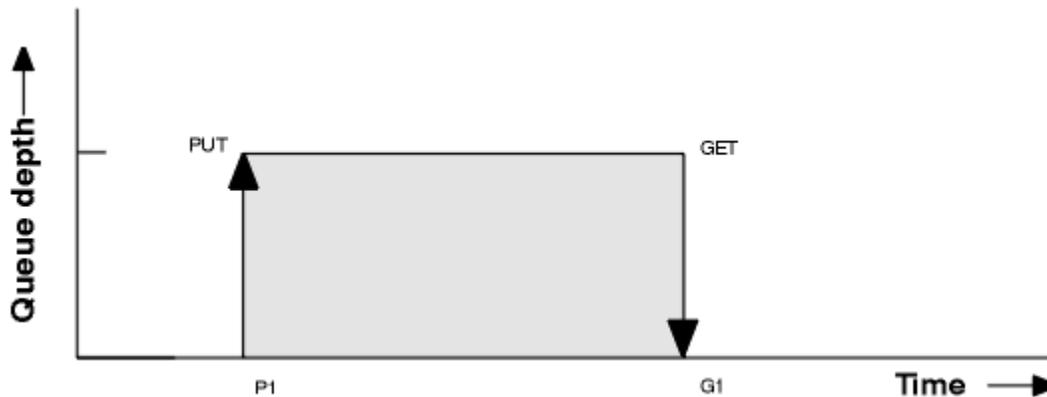


그림 3. 큐 서비스 간격 이벤트 이해

큐 서비스 간격 이벤트의 가능한 결과는 다음과 같습니다.

- PUT과 GET 사이의 경과 시간이 서비스 간격보다 작거나 같습니다.

- 큐 서비스 간격 이벤트가 사용 가능한 경우 시간 G1에 큐 서비스 간격 정상 이벤트가 생성됩니다.
- PUT과 GET 사이의 경과 시간이 서비스 간격보다 큼니다.
  - 큐 서비스 간격 이벤트가 사용 가능한 경우 시간 G1에 큐 서비스 간격 높음 이벤트가 생성됩니다.

서비스 타이머 시작 및 이벤트 생성에 대한 알고리즘은 20 페이지의 『큐 서비스 간격 이벤트에 대한 규칙』에서 설명됩니다.

## 관련 참조

[큐 서비스 간격 확인](#)

[큐 서비스 간격 높음](#)

[QServiceIntervalEvent\(MQLONG\)](#)

[ServiceIntervalEvent 특성](#)

## 서비스 타이머

큐 서비스 간격 이벤트는 서비스 타이머라고 부르는 내부 타이머를 사용하는데, 이것은 큐 관리자에 의해 제어됩니다. 서비스 타이머는 큐 서비스 간격 이벤트가 사용 가능한 경우에만 사용됩니다.

### 서비스 타이머는 정확하게 무엇을 측정합니까?

서비스 타이머는 비어있는 큐에 대한 MQPUT 호출이나 Get 조작과 다음 put 또는 get 사이의 경과 시간을 측정합니다(이들 두 조작 사이에 큐 용량이 0이 아니라고 가정함).

### 언제 서비스 타이머가 활성화입니까?

서비스 타이머는 큐가 메시지를 갖고 있고(용량이 0이 아님) 큐 서비스 간격 이벤트가 사용 가능한 경우 항상 활성화(실행 중)입니다. 큐가 비게 되는 경우(큐 용량이 0), 타이머는 OFF 상태에 들어가고 다음 Put에서 재시작됩니다.

### 언제 서비스 타이머가 재설정됩니다?

서비스 타이머는 항상 Get 조작 후에 재설정됩니다. 또한 빈 큐에 대한 MQPUT 호출에 의해 재설정됩니다. 그러나 큐 서비스 간격 이벤트에서 반드시 재설정되지는 않습니다.

### 서비스 타이머는 어떻게 사용됩니까?

Get 조작이나 MQPUT 호출 뒤에, 큐 관리자가 서비스 타이머에 의해 측정된 경과 시간을 사용자 정의 서비스 간격에 대해 비교합니다. 이 비교의 결과는 다음과 같습니다.

- Get 조작이 있고 경과 시간이 서비스 간격보다 작거나 같고 이 이벤트가 사용 가능한 경우 정상 이벤트가 생성됩니다.
- 경과 시간이 서비스 간격보다 크고 이 이벤트가 사용 가능한 경우 높음 이벤트가 생성됩니다.

### 애플리케이션이 서비스 타이머를 읽을 수 있습니까?

아니오. 서비스 타이머는 애플리케이션에는 사용 불가능한 내부 타이머입니다.

### TimeSinceReset 매개변수의 개념

TimeSinceReset 매개변수는 이벤트 데이터에서 이벤트 통계의 일부로 리턴됩니다. 이벤트 통계가 재설정되지 않은 경우 연속적인 큐 서비스 간격 이벤트 사이의 시간을 지정합니다.

## 큐 서비스 간격 이벤트에 대한 규칙

공식 규칙이 서비스 타이머가 설정되고 큐 서비스 간격 이벤트가 생성되는 시기를 제어합니다.

## 서비스 타이머에 대한 규칙

서비스 타이머는 다음과 같이 0으로 재설정되고 재시작됩니다.

- 비어 있는 큐에 대한 MQPUT 호출 후.
- MQGET 호출 후에 큐가 비어있지 않은 경우 MQGET 호출 후.

타이머 재설정은 이벤트가 생성되었는지 여부에 의존하지 않습니다.

큐 관리자 시동 시에 큐 용량이 0보다 큰 경우 서비스 타이머가 시동 시간으로 설정됩니다.

Get 조작 후에 큐가 비어 있는 경우, 타이머는 OFF 상태에 들어갑니다.

## 큐 서비스 간격 높음 이벤트

큐 서비스 간격 이벤트가 사용 가능하게 설정(HIGH로 설정)되어야 합니다.

큐 서비스 간격 높음 이벤트는 큐 서비스 간격 정상 이벤트가 생성될 때 자동으로 사용 가능하게 됩니다.

서비스 시간이 서비스 간격보다 더 큰 경우, 다음 MQPUT 또는 Get 조작 시 또는 그 이전에 이벤트가 생성됩니다.

## 큐 서비스 간격 정상 이벤트

큐 서비스 간격 정상 이벤트는 큐 서비스 간격 높음 이벤트가 생성될 때 자동으로 사용 가능하게 됩니다.

서비스 시간(경과 시간)이 서비스 간격보다 작거나 같은 경우, 다음 Get 조작 시 또는 그 이전에 이벤트가 생성됩니다.

### 관련 태스크

21 페이지의 『큐 서비스 간격 이벤트 사용 가능』

큐 서비스 간격 이벤트에 대해 큐를 구성하려면 적절한 큐 관리자 및 큐 속성을 설정합니다.

## 큐 서비스 간격 이벤트 사용 가능

큐 서비스 간격 이벤트에 대해 큐를 구성하려면 적절한 큐 관리자 및 큐 속성을 설정합니다.

## 이 태스크 정보

높음 및 정상 이벤트는 상호 배타적입니다. 즉, 하나가 사용 가능하면 다른 하나는 자동으로 사용 불가능합니다.

- 높음 이벤트가 큐에서 생성될 때, 큐 관리자가 자동으로 높음 이벤트를 사용 불가능하게 하고 해당 큐에 대한 정상 이벤트를 사용 가능하게 합니다.
- 정상 이벤트가 큐에서 생성될 때, 큐 관리자가 자동으로 정상 이벤트를 사용 불가능하게 하고 해당 큐에 대한 높음 이벤트를 사용 가능하게 합니다.

큐 서비스 간격 이벤트	큐 속성
큐 서비스 간격 높음 큐 서비스 간격 확인 큐 서비스 간격 이벤트 없음	QSVCI EV (HIGH) QSVCI EV (OK) QSVCI EV (NONE)
서비스 간격	QSVCI NT ( <i>tt</i> ) 여기서 <i>tt</i> 는 서비스입니다. 밀리초 단위의 서비스 간격 시간입니다.

큐 서비스 간격 이벤트를 사용 가능하게 설정하려면 다음 단계를 수행하십시오.

## 프로시저

1. 큐 관리자 속성 PERFM EV를 ENABLED로 설정하십시오.  
성능 이벤트가 큐 관리자에서 사용 가능합니다.
2. 큐의 큐 서비스 간격 높음 또는 정상 이벤트에 대해 제어 속성 QSVCI EV를 필요한 대로 설정하십시오.
3. 적절한 서비스 간격 시간을 지정하기 위해 큐에 대한 QSVCI NT 속성을 설정하십시오.

## 예

10초(10,000밀리초)의 서비스 간격 시간을 갖고 큐 서비스 간격 높음 이벤트를 사용 가능하게 설정하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR PERFMEV(ENABLED)
```

```
ALTER QLOCAL('MYQUEUE') QSVCIINT(10000) QSVCIIEV(HIGH)
```

## 큐 서비스 간격 이벤트 예

큐 서비스 간격 이벤트로부터 얻을 수 있는 정보를 이해하려면 다음 예제를 사용하십시오.

세 예제는 큐 서비스 간격 이벤트의 사용에 대해 보다 복잡한 단계별 설명을 제공합니다.

예제와 함께 제공되는 그림은 동일한 구조를 가집니다.

- 그림 1은 개별 MQGET 호출 및 MQPUT 호출을 표시하는, 큐 용량 대 시간의 그래프입니다.
- 주석 섹션은 시간 제한조건의 비교를 표시합니다. 사용자가 고려해야 하는 세 개의 기간이 있습니다.
  - 사용자 정의 서비스 간격.
  - 서비스 타이머가 측정한 시간.
  - 이벤트 통계가 마지막으로 재설정된 이후의 시간(이벤트 데이터의 TimeSinceReset).
- 이벤트 통계 요약 섹션은 임의의 순간에 사용 가능한 이벤트 및 생성되는 이벤트를 표시합니다.

예는 큐 서비스 간격 이벤트의 다음 측면을 설명합니다.

- 큐 용량이 시간에 따라 변하는 정도.
- 서비스 타이머가 측정하는 경과 시간이 서비스 간격과 비교하는 정도.
- 사용 가능한 이벤트.
- 생성되는 이벤트.

**알아두기:** 예 1은 메시지가 간헐적이고 다음 메시지가 도달하기 전에 각 메시지가 큐에서 제거되는 단순한 사례를 보여줍니다. 이벤트 데이터로부터 큐의 최대 메시지 수가 1이었음을 알고 있습니다. 그러므로, 각 메시지가 큐에 있었던 시간을 파악할 수 있습니다.

그러나 큐에 둘 이상의 메시지가 있고 MQGET 호출과 MQPUT 호출의 순서가 예측 불가능한 일반적인 경우, 큐 서비스 간격 이벤트를 사용하여 개별 메시지가 큐에 남아 있는 기간을 계산할 수 없습니다. 이벤트 데이터에서 리턴되는 TimeSinceReset 매개변수는 큐에 메시지가 없는 시간의 비율을 포함할 수 있습니다. 그러므로 이들 통계로부터 파생시키는 모든 결과는 암시적으로 이들 시간을 포함하도록 평균화됩니다.

### 관련 개념

#### [19 페이지의 『큐 서비스 간격 이벤트』](#)

큐 서비스 간격 이벤트는 서비스 간격이라고 부르는 사용자 정의 시간 간격 안에서 큐에 대해 조작이 수행되었는지 여부를 표시합니다. 설치에 따라서, 큐 서비스 간격 이벤트를 사용하여 메시지가 충분히 빨리 큐에서 제거되는지 여부를 모니터링할 수 있습니다.

#### [20 페이지의 『서비스 타이머』](#)

큐 서비스 간격 이벤트는 서비스 타이머라고 부르는 내부 타이머를 사용하는데, 이것은 큐 관리자에 의해 제어됩니다. 서비스 타이머는 큐 서비스 간격 이벤트가 사용 가능한 경우에만 사용됩니다.

### 큐 서비스 간격 이벤트: 예 1

MQGET 호출과 MQPUT 호출의 기본 순서로, 큐 용량은 항상 1 또는 0입니다.

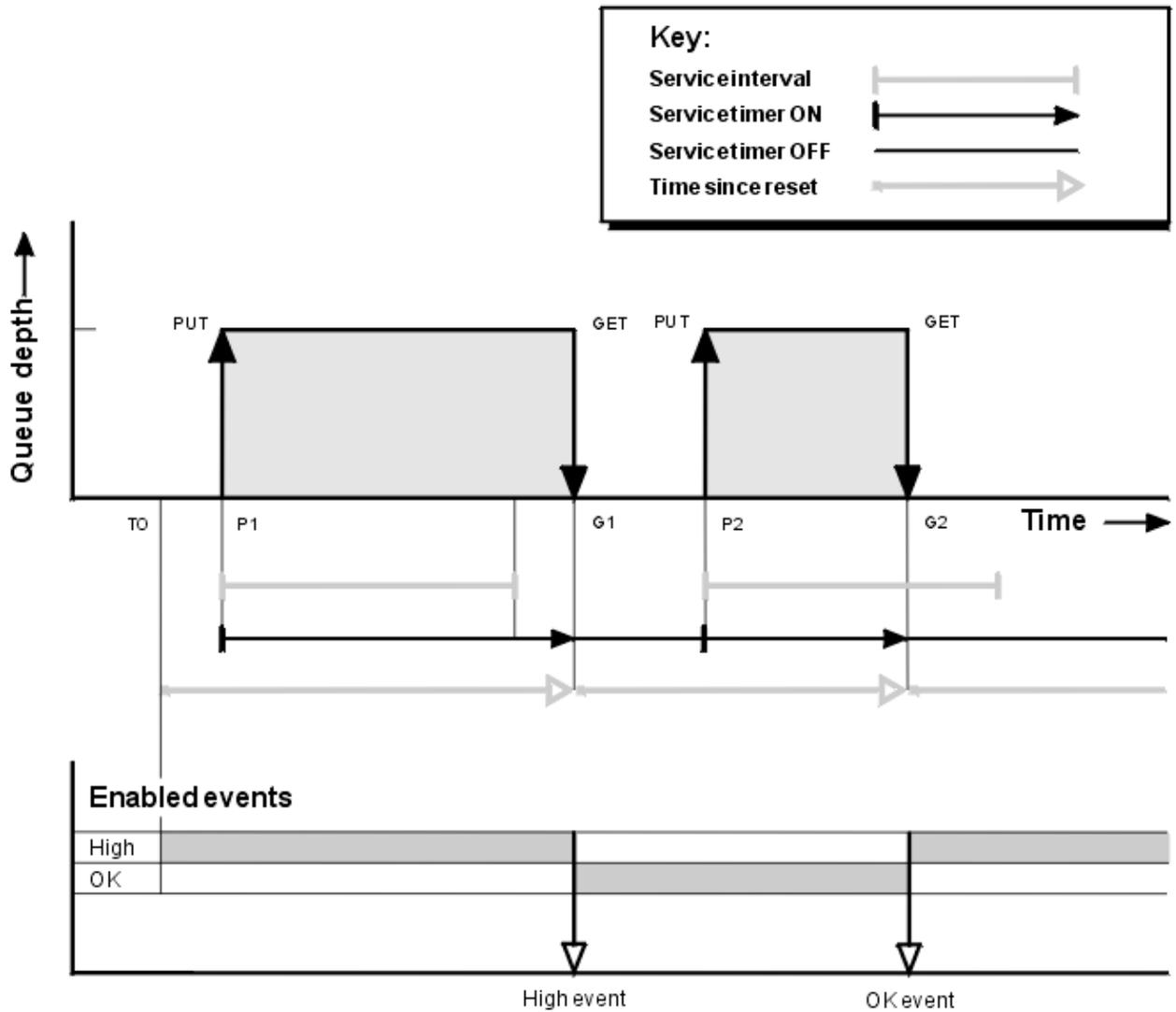


그림 4. 큐 서비스 간격 이벤트 - 예 1

### 주석

- P1에서 애플리케이션이 빈 큐에 메시지를 넣습니다. 이것은 서비스 타이머를 시작합니다.  
T0은 큐 관리자 시동 시간일 수 있음을 주의하십시오.
- G1에서 다른 애플리케이션이 큐에서 메시지를 가져옵니다. P1과 G1 사이의 경과 시간이 서비스 간격보다 크기 때문에 큐 서비스 간격 높음 이벤트가 G1에서의 MQGET 호출에서 생성됩니다. 높음 이벤트가 생성될 때 큐 관리자는 다음과 같도록 이벤트 제어 속성을 재설정합니다.
  - 정상 이벤트가 자동으로 사용 가능하게 됩니다.
  - 높음 이벤트는 사용 불가능합니다.
 큐가 이제 비어 있기 때문에 서비스 타이머는 OFF 상태로 전환됩니다.
- P2에서 두 번째 메시지가 큐에 들어갑니다. 이것은 서비스 타이머를 재시작합니다.
- G2에 메시지가 큐에서 제거됩니다. 그러나 P2와 G2 사이의 경과 시간이 서비스 간격보다 작기 때문에 큐 서비스 간격 정상 이벤트가 G2에서 MQGET 호출에서 생성됩니다. 정상 이벤트가 생성될 때 큐 관리자는 다음과 같도록 제어 속성을 재설정합니다.
  - 높음 이벤트가 자동으로 사용 가능하게 됩니다.
  - 정상 이벤트는 사용 불가능합니다.

큐가 비어 있기 때문에 서비스 타이머는 다시 OFF 상태로 전환됩니다.

## 이벤트 통계 요약

24 페이지의 표 5에서는 이 예에 대한 이벤트 통계를 요약합니다.

표 5. 예 1에 대한 이벤트 통계 요약		
	이벤트 1	이벤트 2
이벤트 시간	T(G1)	T(G2)
이벤트 유형	높음	확인
TimeSinceReset	T(G1) - T(0)	T(G2) - T(G1)
HighQDepth	1	1
MsgEnqCount	1	1
MsgDeqCount	1	1

23 페이지의 그림 4의 중간 부분은 해당 큐에 대한 서비스 간격과 비교되는 서비스 타이머가 측정한 경과 시간을 보여줍니다. 큐 서비스 간격 이벤트가 발행할지 여부를 확인하려면 서비스 타이머를 나타내는 수평선(화살표를 갖는)의 길이를 서비스 간격을 나타내는 선의 길이와 비교하십시오. 서비스 타이머 선이 더 길고 큐 서비스 간격 높음 이벤트가 사용 가능한 경우, 큐 서비스 간격 높음 이벤트가 다음 GET에서 발생합니다. 타이머 선이 더 짧고 큐 서비스 간격 정상 이벤트가 사용 가능한 경우, 큐 서비스 간격 정상 이벤트가 다음 GET에서 발생합니다.

### 큐 서비스 간격 이벤트: 예 2

MQPUT 호출과 MQGET 호출의 순서로, 큐 용량은 항상 1 또는 0이 아닙니다.

이 예는 또한 예를 들어 시간 P2에서 이벤트가 생성되지 않고 재설정되는 타이머의 인스턴스를 보여줍니다.

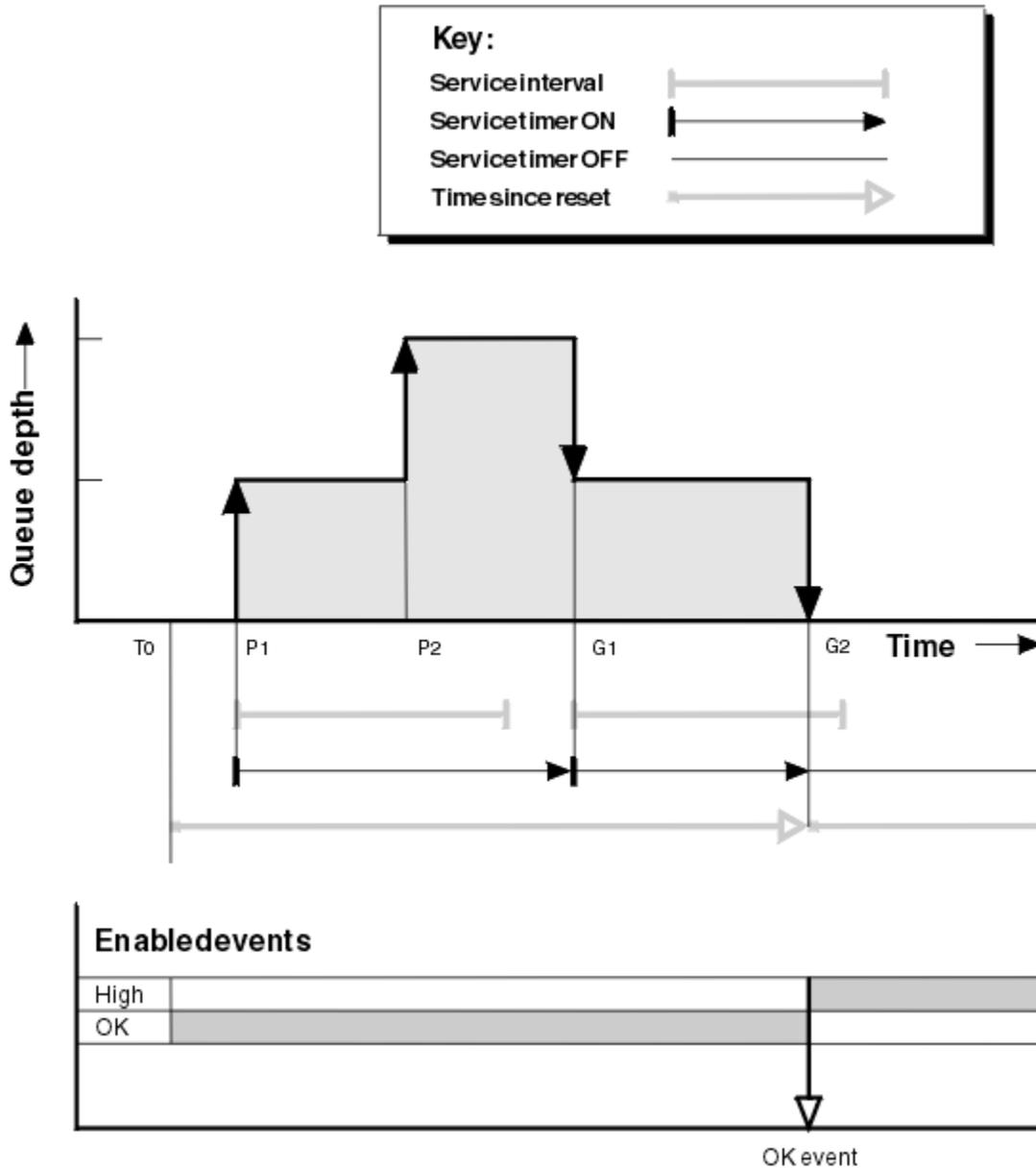


그림 5. 큐 서비스 간격 이벤트 - 예 2

### 주석

이 예에서 정상 이벤트가 초기에 사용 가능하고 큐 통계가 시간 T0에 재설정되었습니다.

1. P1에 첫 번째 PUT이 서비스 타이머를 시작합니다.
2. P2에서, PUT은 정상 이벤트를 유발할 수 없기 때문에 두 번째 PUT이 이벤트를 생성하지 않습니다.
3. G1에서 서비스 간격이 이제 초과되었으므로 정상 이벤트는 생성되지 않습니다. 그러나 MQGET 호출이 서비스 타이머가 재설정되게 만듭니다.
4. G2에서 서비스 간격 내에 두 번째 GET이 발생하고 이번에는 정상 이벤트가 생성됩니다. 큐 관리자는 다음과 같도록 이벤트 제어 속성을 재설정합니다.
  - a. 높음 이벤트가 자동으로 사용 가능하게 됩니다.
  - b. 정상 이벤트는 사용 불가능합니다.

큐가 이제 비어 있기 때문에 서비스 타이머는 OFF 상태로 전환됩니다.

## 이벤트 통계 요약

26 페이지의 표 6에서는 이 예에 대한 이벤트 통계를 요약합니다.

표 6. 예 2에 대한 이벤트 통계 요약	
	<b>이벤트 2</b>
이벤트 시간	T(G2)
이벤트 유형	확인
TimeSinceReset	T(G2) - T(0)
HighQDepth	2
MsgEnqCount	2
MsgDeqCount	2

### 큐 서비스 간격 이벤트: 예 3

이전 예보다 더 산발적인 MQGET 호출과 MQPUT 호출의 순서입니다.

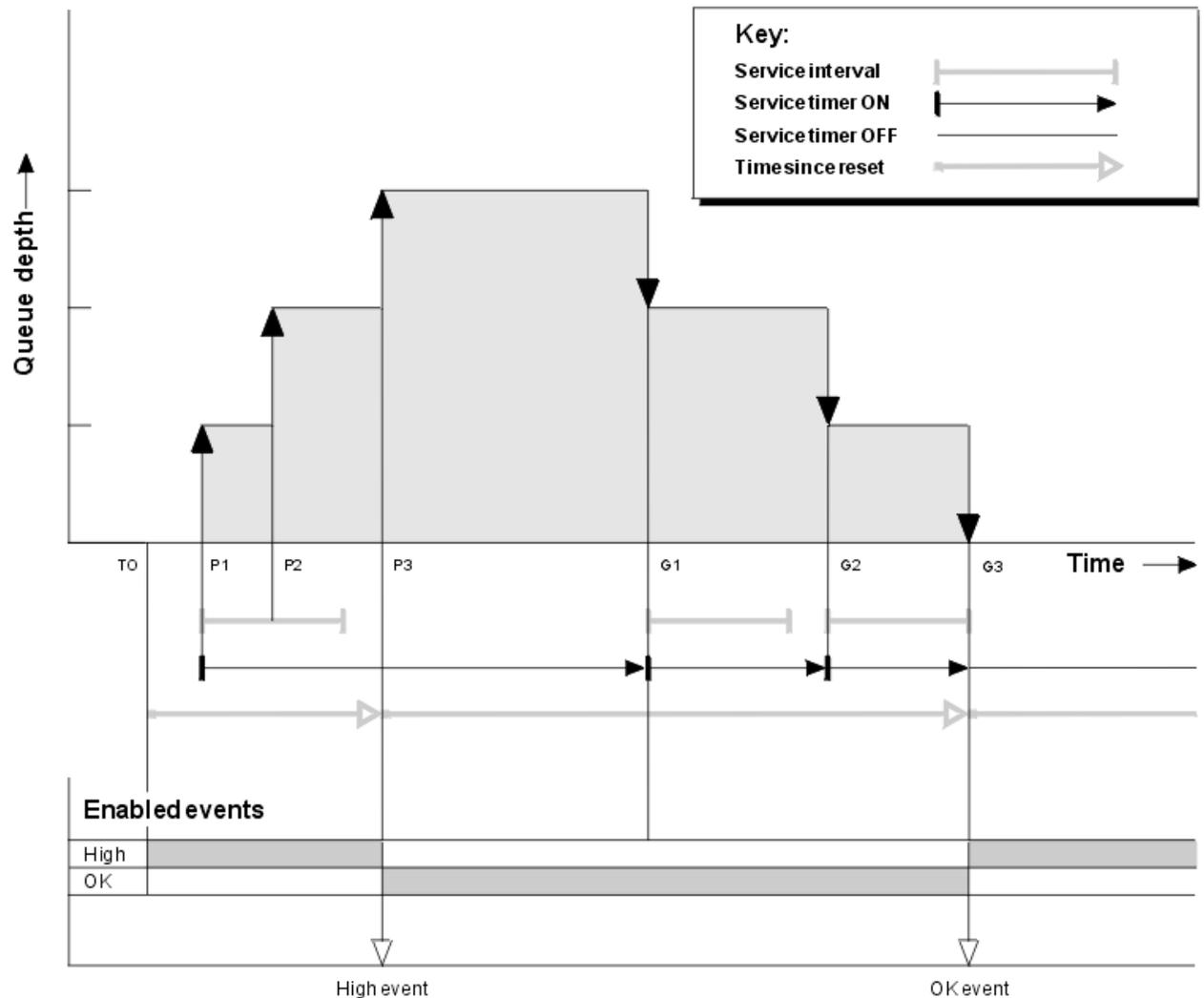


그림 6. 큐 서비스 간격 이벤트 - 예 3

## 주석

1. 시간 T(0)에서, 큐 통계가 재설정되고 큐 서비스 간격 높음 이벤트가 사용 가능합니다.
2. P1에 첫 번째 PUT이 서비스 타이머를 시작합니다.
3. P2에, 두 번째 PUT이 큐 용량을 2로 증가시킵니다. 서비스 간격 시간이 초과되지 않았기 때문에 높음 이벤트가 생성되지 않습니다.
4. P3에서 세 번째 PUT이 높음 이벤트가 생성되게 합니다. (타이머가 서비스 간격을 초과했습니다.) 큐 용량이 PUT 전에 0이 아니었기 때문에 타이머는 재설정되지 않습니다. 그러나 정상 이벤트가 사용 가능합니다.
5. G1에서 서비스 간격이 초과되었고 정상 이벤트가 사용 가능하기 때문에 MQGET 호출이 이벤트를 생성하지 않습니다. 그러나 MQGET 호출은 서비스 타이머를 재설정합니다.
6. G2에서 서비스 간격이 초과되었고 정상 이벤트가 사용 가능하기 때문에 MQGET 호출이 이벤트를 생성하지 않습니다. 다시, MQGET 호출이 서비스 타이머를 재설정합니다.
7. G3에서 세 번째 GET이 큐를 비우고 서비스 타이머는 서비스 간격과 같습니다. 그러므로 정상 이벤트가 생성됩니다. 서비스 타이머가 재설정되고 높음 이벤트가 사용 가능합니다. MQGET 호출이 큐를 비우며, 이것은 타이머를 OFF 상태로 만듭니다.

## 이벤트 통계 요약

27 페이지의 표 7에서는 이 예에 대한 이벤트 통계를 요약합니다.

표 7. 예 3에 대한 이벤트 통계 요약		
	이벤트 1	이벤트 2
이벤트 시간	T(P3)	T(G3)
이벤트 유형	높음	확인
TimeSinceReset	T(P3) - T(0)	T(G3) - T(P3)
HighQDepth	3	3
MsgEnqCount	3	0
MsgDeqCount	0	3

## 큐 용량 이벤트

큐 용량 이벤트를 큐 용량, 즉 큐에 있는 메시지 수와 관련됩니다.

WebSphere MQ 애플리케이션에서는 큐가 가득 차면 안됩니다. 가득 차는 경우 애플리케이션은 지정한 큐에 더 이상 메시지를 넣을 수 없습니다. 이것이 발생하는 경우 메시지가 유실되지 않더라도 가득 찬 큐는 상당한 불편을 유발할 수 있습니다. 메시지를 처리하는 애플리케이션이 덜어낼 수 있는 것보다 더 빨리 큐에 메시지가 넣어지는 경우 큐의 메시지 수가 쌓일 수 있습니다.

이 문제점의 솔루션은 특정 상황에 의존하지만 다음을 포함할 수 있습니다.

- 일부 메시지를 다른 큐로 전환.
- 큐에서 더 많은 메시지를 덜어내기 위한 새 애플리케이션 시작.
- 본질적이 아닌 메시지 트래픽 중지.
- 임시 최대를 극복하기 위해 큐 용량 증가.

문제점이 진행 중일 수 있다는 사전 경고가 예방 조치를 취하기가 더 쉽게 만듭니다. 이 목적으로 WebSphere MQ는 다음과 같은 큐 용량 이벤트를 제공합니다.

### 큐 용량 높음 이벤트

큐 용량이 큐 용량 상한이라고 부르는 사전 정의된 임계값으로 증가했음을 표시합니다.

### 큐 용량 낮음 이벤트

큐 용량이 큐 용량 하한이라고 부르는 사전 정의된 임계값으로 감소했음을 표시합니다.

## 큐 가득 참 이벤트

큐가 최대 용량에 도달했음, 즉 큐가 가득 찼음을 표시합니다.

큐 가득 참 이벤트는 애플리케이션이 최대 용량에 도달한 큐에 메시지를 넣으려고 시도할 때 생성됩니다. 큐 용량 높음 이벤트가 큐가 채워져 가고 있다는 사전 경고를 제공합니다. 이것은 이 이벤트를 수신했을 때 시스템 관리자가 어떤 예방 조치를 취해야 함을 의미합니다. 예방 조치가 성공하고 큐 용량이 안전한 수준으로 떨어지는 경우 큐 관리자가 큐 용량 낮음 이벤트를 생성하도록 큐 관리자를 구성할 수 있습니다.

첫 번째 큐 용량 이벤트 예는 큐가 가득 차게 되기를 막는 추정 조치의 효과를 보여줍니다.

## 관련 개념

29 페이지의 『[큐 용량 이벤트 예](#)』

큐 용량 이벤트로부터 얻을 수 있는 정보를 이해하려면 이들 예를 사용하십시오.

## 관련 참조

[큐 가득 참](#)

[큐 용량 상한](#)

[큐 용량 하한](#)

## 큐 용량 이벤트 사용 가능

임의의 큐 용량 이벤트에 대해 큐를 구성하려면 적절한 큐 관리자 및 큐 속성을 설정합니다.

## 이 태스크 정보

기본적으로, 모든 큐 용량 이벤트는 사용 불가능합니다. 사용 가능할 때 큐 용량 이벤트가 다음과 같이 생성됩니다.

- 큐 용량 높음 이벤트는 메시지가 큐에 넣어져서 큐 용량이 큐 용량 상한에 의해 판별되는 값보다 크거나 같게 될 때 생성됩니다.
  - 큐 용량 높음 이벤트는 동일한 큐에서의 큐 용량 낮음 이벤트에 의해 자동으로 사용 가능하게 됩니다.
  - 큐 용량 높음 이벤트는 동일한 큐에서 큐 용량 낮음 및 큐 가득 참 이벤트를 둘 다 자동으로 사용 가능하게 합니다.
- 큐 용량 낮음 이벤트는 메시지가 Get 조작에 의해 큐에서 제거되어 큐 용량이 큐 용량 하한에 의해 판별되는 값보다 작거나 같게 될 때 생성됩니다.
  - 큐 용량 낮음 이벤트는 동일한 큐에서의 큐 용량 높음 이벤트 또는 큐 가득 참 이벤트에 의해 자동으로 사용 가능하게 됩니다.
  - 큐 용량 낮음 이벤트는 동일한 큐에서 큐 용량 높음 및 큐 가득 참 이벤트를 둘 다 자동으로 사용 가능하게 합니다.
- 큐 가득 참 이벤트는 큐가 가득 찼기 때문에 애플리케이션이 큐에 메시지를 넣을 수 없을 때 생성됩니다.
  - 큐 가득 참 이벤트는 동일한 큐에서의 큐 용량 높음 또는 큐 용량 낮음 이벤트에 의해 자동으로 사용 가능하게 됩니다.
  - 큐 가득 참 이벤트는 동일한 큐에서 큐 용량 낮음 이벤트를 자동으로 사용 가능하게 합니다.

큐 용량 이벤트에 대해 큐를 구성하려면 다음 단계를 수행하십시오.

## 프로시저

1. 큐 관리자 속성 PERFMV를 사용하여 큐 관리자에서 성능 이벤트를 사용 가능하게 하십시오.
2. 필수 큐에서 이벤트를 사용 가능하게 설정하려면 다음 속성 중 하나를 설정하십시오.
  - *QDepthHighEvent*(MQSC의 QDPHIEV)
  - *QDepthLowEvent*(MQSC의 QDPLOEV)
  - *QDepthMaxEvent*(MQSC의 QDPMAXEV)
3. 옵션: 한계를 설정하려면 최대 큐 용량의 백분율로서 다음 속성을 지정하십시오.
  - *QDepthHighLimit*(MQSC의 QDEPTHHI)

- *QDepthLowLimit*(MQSC의 QDEPTHLO)

**제한사항:** QDEPTHHI는 QDEPTHLO보다 작지 않아야 합니다.

QDEPTHHI가 QDEPTHLO와 동일한 경우 큐 용량이 어느 방향으로든 값을 통과할 때마다 이벤트 메시지가 생성됩니다. 상위 임계값은 큐 용량이 해당 값 미만일 때 사용 가능하고 하위 임계값은 용량이 해당 값 이상일 때 사용 가능하기 때문입니다.

## 결과

### 참고:

만기 메시지가 Get 조작에 의해 큐에서 제거되어 큐 용량이 큐 용량 하한에 의해 판별되는 값보다 작거나 같게 될 때는 큐 용량 낮음 이벤트가 생성되지 않습니다.

IBM WebSphere MQ는 성공적인 Get 조작 중에만 낮음 이벤트 메시지를 생성합니다. 그러므로, 만기된 메시지가 큐에서 제거될 때 큐 용량 낮음 메시지가 생성되지 않습니다.

추가로, 큐에서 이런 만기 메시지의 제거 후에 큐 용량 높음 이벤트 및 큐 용량 낮음 이벤트가 재설정되지 않습니다.

### 예

80%에서 설정된 한계를 갖고 MYQUEUE 큐에서 큐 용량 높음 이벤트를 사용 가능하게 설정하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR PERFMEV(ENABLED)
ALTER QLOCAL('MYQUEUE') QDEPTHHI(80) QDPHIEV(ENABLED)
```

20%에서 설정된 한계를 갖고 MYQUEUE 큐에서 큐 용량 낮음 이벤트를 사용 가능하게 설정하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR PERFMEV(ENABLED)
ALTER QLOCAL('MYQUEUE') QDEPTHLO(20) QDPLOEV(ENABLED)
```

MYQUEUE 큐에서 큐 가득 참 이벤트를 사용 가능하게 설정하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR PERFMEV(ENABLED)
ALTER QLOCAL('MYQUEUE') QDPMAEV(ENABLED)
```

## 큐 용량 이벤트 예

큐 용량 이벤트로부터 얻을 수 있는 정보를 이해하려면 이들 예를 사용하십시오.

첫 번째 예는 큐 용량 이벤트의 기본 설명을 제공합니다. 두 번째 예는 더 방대하지만 원리는 첫 번째 예와 동일합니다. 두 예는 모두 다음과 같이 동일한 큐 정의를 사용합니다.

MYQUEUE1 큐는 1000 메시지의 최대 용량을 갖습니다. 큐 용량 상한은 80%이고 큐 용량 하한은 20%입니다. 초기에는 큐 용량 높음 이벤트가 사용 가능한 반면, 다른 큐 용량 이벤트는 사용 불가능합니다.

이 큐를 구성하기 위한 WebSphere MQ 명령(MQSC)은 다음과 같습니다.

```
ALTER QMGR PERFMEV(ENABLED)

DEFINE QLOCAL('MYQUEUE1') MAXDEPTH(1000) QDPMAEV(DISABLED) QDEPTHHI(80)
QDPHIEV(ENABLED) QDEPTHLO(20) QDPLOEV(DISABLED)
```

### 관련 개념

[27 페이지의 『큐 용량 이벤트』](#)

큐 용량 이벤트를 큐 용량, 즉 큐에 있는 메시지 수와 관련됩니다.

## 관련 태스크

28 페이지의 『큐 용량 이벤트 사용 가능』

임의의 큐 용량 이벤트에 대해 큐를 구성하려면 적절한 큐 관리자 및 큐 속성을 설정합니다.

## 관련 참조

MQSC 명령

## 큐 용량 이벤트: 예 1

큐 용량 이벤트의 기본 순서입니다.

30 페이지의 그림 7에서는 시간에 따른 큐 용량의 변화를 보여줍니다.

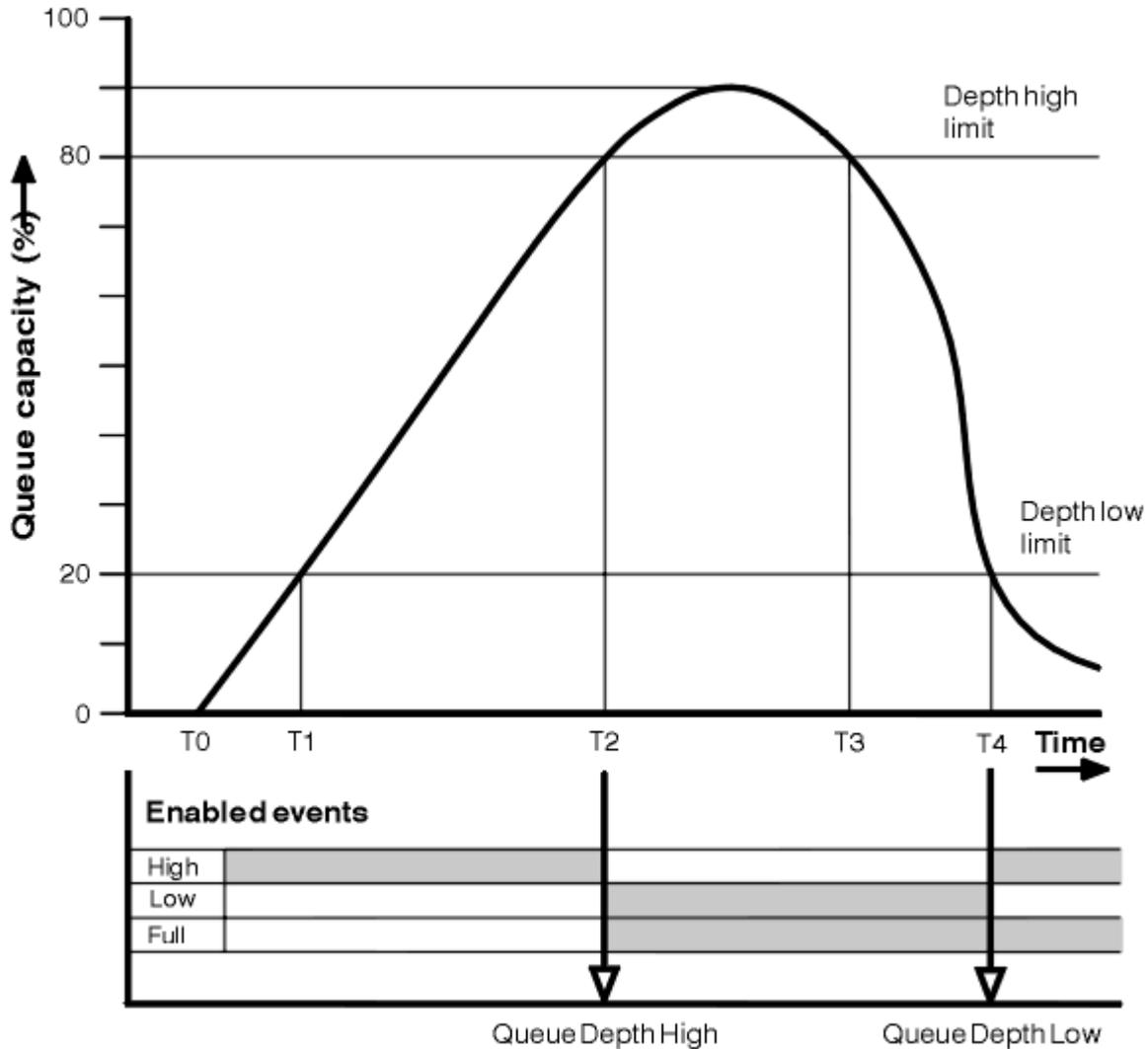


그림 7. 큐 용량 이벤트 (1)

## 주석

1. T(1)에서, 큐 용량이 증가하고(MQGET 호출보다 더 많은 MQPUT 호출) 큐 용량 하한을 지납니다. 이 때는 이벤트가 생성되지 않습니다.
2. 큐 용량이 T(2)까지 계속 증가하고, 이 때 용량 상한(80%)에 도달하고 큐 용량 높음 이벤트가 생성됩니다. 이것은 큐 가득 참 및 큐 용량 낮음 이벤트를 둘 다 사용 가능하게 합니다.
3. 이벤트에 의해 착수되는 (추정된) 예방 조치가 큐가 가득 차지 못하게 막습니다. 시간 T(3)까지, 큐 용량 상한에 다시 도달되었고, 이 시간부터 위입니다. 이 때는 이벤트가 생성되지 않습니다.
4. 용량 하한(20%)에 도달하고 큐 용량 낮음 이벤트가 생성되는 T(4)까지 큐 용량이 계속 감소합니다.

이것은 큐 가득 참 및 큐 용량 높음 이벤트를 둘 다 사용 가능하게 합니다.

## 이벤트 통계 요약

31 페이지의 표 8에서는 큐 이벤트 통계를 요약하고 31 페이지의 표 9에서는 사용 가능한 이벤트를 요약합니다.

표 8. 큐 용량 이벤트(예 1)에 대한 이벤트 통계 요약		
	이벤트 2	이벤트 4
이벤트 시간	T(2)	T(4)
이벤트 유형	큐 용량 상한	큐 용량 하한
TimeSinceReset	T(2) - T(0)	T(4) - T(2)
HighQDepth(재설정 이후 최대 큐 용량)	800	900
MsgEnqCount	1157	1220
MsgDeqCount	357	1820

표 9. 사용 가능한 이벤트를 표시하는 요약			
시간 기간	큐 용량 높음 이벤트	큐 용량 낮음 이벤트	큐 가득 참 이벤트
T(1) 이전	ENABLED	-	-
T(1) - T(2)	ENABLED	-	-
T(2) - T(3)	-	ENABLED	ENABLED
T(3) - T(4)	-	ENABLED	ENABLED
T(4) 이후	ENABLED	-	ENABLED

## 큐 용량 이벤트: 예 2

큐 용량 이벤트의 더 방대한 순서입니다.

32 페이지의 그림 8에서는 시간에 따른 큐 용량의 변화를 보여줍니다.

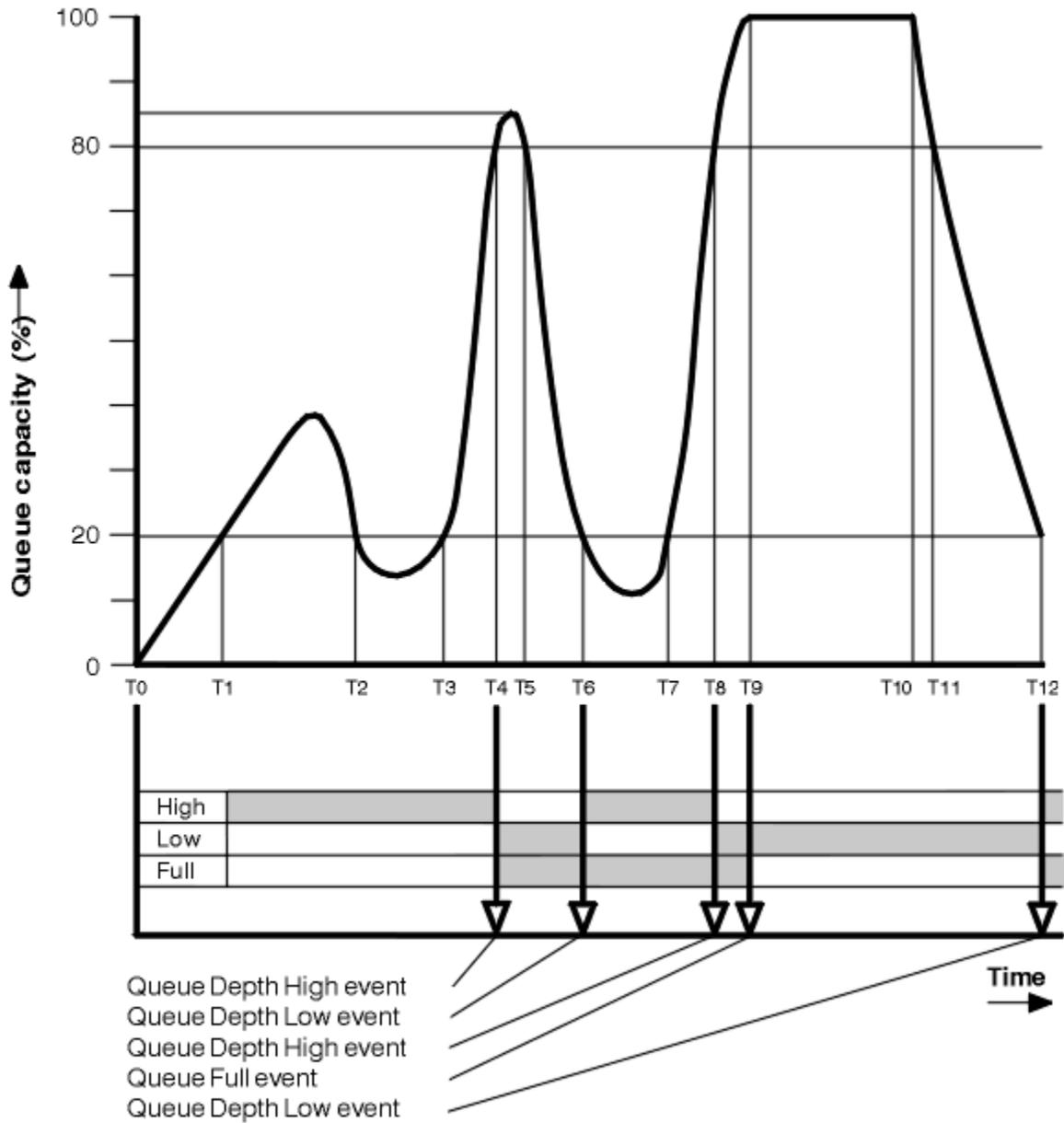


그림 8. 큐 용량 이벤트 (2)

**주석**

- 다음 시간에는 큐 용량 낮음 이벤트가 생성되지 않습니다.
  - T(1) (큐 용량이 증가하고, 사용 가능하지 않음)
  - T(2) (사용 가능하지 않음)
  - T(3) (큐 용량이 증가하고, 사용 가능하지 않음)
- T(4)에서 큐 용량 높음 이벤트가 발생합니다. 이것은 큐 가득 참 및 큐 용량 낮음 이벤트를 둘 다 사용 가능하게 합니다.
- T(9)에서 큐가 가득 찼기 때문에 큐에 넣을 수 없는 첫 번째 메시지 후에 큐 가득 참 이벤트가 발생합니다.
- T(12)에서 큐 용량 낮음 이벤트가 발생합니다.

## 이벤트 통계 요약

33 페이지의 표 10에서는 큐 이벤트 통계를 요약하고 33 페이지의 표 11에서는 이 예에 대해 다른 시간에 사용 가능한 이벤트를 요약합니다.

	이벤트 4	이벤트 6	이벤트 8	이벤트 9	이벤트 12
이벤트 시간	T(4)	T(6)	T(8)	T(9)	T(12)
이벤트 유형	큐 용량 상한	큐 용량 하한	큐 용량 상한	큐 가득 참	큐 용량 하한
TimeSinceReset	T(4) - T(0)	T(6) - T(4)	T(8) - T(6)	T(9) - T(8)	T(12) - T(9)
HighQDepth	800	855	800	1000	1000
MsgEnqCount	1645	311	1377	324	221
MsgDeqCount	845	911	777	124	1021

시간 기간	큐 용량 높음 이벤트	큐 용량 낮음 이벤트	큐 가득 참 이벤트
T(0) - T(4)	ENABLED	-	-
T(4) - T(6)	-	ENABLED	ENABLED
T(6) - T(8)	ENABLED	-	ENABLED
T(8) - T(9)	-	ENABLED	ENABLED
T(9) - T(12)	-	ENABLED	-
T(12) 이후	ENABLED	-	ENABLED

**참고:** 이벤트를 동기점을 벗어납니다. 그러므로 빈 큐를 가진 후 채워서 이벤트를 유발한 후, 동기점 관리자의 제어 하에 모든 메시지를 롤백할 수 있습니다. 그러나, 이벤트 사용 가능은 자동으로 설정되었으므로, 다음에 큐가 채워질 때 이벤트가 생성되지 않습니다.

## 구성 이벤트

구성 이벤트는 오브젝트가 작성, 변경 또는 삭제될 때 생성되며 명시적 요청에 의해서도 생성될 수 있는 알림입니다.

구성 이벤트는 사용자에게 오브젝트 속성에 대한 변경을 알려줍니다. 4가지 유형의 구성 이벤트가 있습니다.

- 오브젝트 작성 이벤트
- 오브젝트 변경 이벤트
- 오브젝트 삭제 이벤트
- 오브젝트 새로 고치기 이벤트

이벤트 데이터는 다음 정보를 포함합니다.

### 원본 정보

변경이 수행된 큐 관리자, 변경을 수행한 사용자의 ID, 변경을 수행한 방식(예를 들어, 콘솔 명령으로) 등을 포함합니다.

### 컨텍스트 정보

명령 메시지의 메시지 데이터의 컨텍스트 정보의 복제본입니다.

컨텍스트 정보는 명령이 SYSTEM.COMMAND.INPUT 큐에 메시지로써 입력되었을 때만 이벤트 데이터에 포함됩니다.

## 오브젝트 ID

오브젝트의 이름, 유형 및 속성 지정으로 구성됩니다.

## 오브젝트 속성

오브젝트에 있는 모든 속성의 값으로 구성됩니다.

오브젝트 변경 이벤트의 경우 두 메시지가 생성되는데, 하나는 변경 전 정보를 갖고 다른 하나는 그 후의 정보를 갖습니다.

생성되는 모든 구성 이벤트 메시지는 SYSTEM.ADMIN.CONFIG.EVENT 큐에 위치합니다.

## 관련 개념

### [11 페이지의 『구성 이벤트』](#)

구성 이벤트는 구성 이벤트가 명시적으로 요청될 때 또는 오브젝트가 작성, 수정 또는 삭제될 때 자동으로 생성됩니다.

## 관련 참조

[오브젝트 작성](#)

[오브젝트 변경](#)

[오브젝트 삭제](#)

[오브젝트 새로 고치기](#)

### [7 페이지의 『이벤트 유형』](#)

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

## 구성 이벤트 생성

구성 이벤트가 생성되게 하는 명령을 보고 구성 이벤트가 생성되지 않는 상황을 이해하려면 이 페이지를 사용하십시오.

구성 이벤트 메시지는 CONFIGEV 큐 관리자 속성이 ENABLED일 때 구성 이벤트 큐에 넣어집니다.

- 다음 명령 중 하나 또는 동등한 PCF가 실행됩니다.
  - DELETE AUTHINFO
  - DELETE CFSTRUCT
  - DELETE CHANNEL
  - DELETE NAMELIST
  - DELETE PROCESS
  - DELETE QMODEL/QALIAS/QREMOTE
  - DELETE STGCLASS
  - DELETE TOPIC
  - REFRESH QMGR
- 다음 명령 중 하나 또는 동등한 PCF가 오브젝트에 대한 변경이 없는 경우에도 실행됩니다.
  - DEFINE/ALTER AUTHINFO
  - DEFINE/ALTER CFSTRUCT
  - DEFINE/ALTER CHANNEL
  - DEFINE/ALTER NAMELIST
  - DEFINE/ALTER PROCESS
  - DEFINE/ALTER QMODEL/QALIAS/QREMOTE
  - DEFINE/ALTER STGCLASS
  - DEFINE/ALTER TOPIC
  - DEFINE MAXSMSGS
  - SET CHLAUTH
  - CONFIGEV 속성이 DISABLED이고 ENABLED로 변경되지 않은 경우 ALTER QMGR

- 다음 명령 중 하나 또는 동등한 PCF가 큐에 대한 변경이 없는 경우에도 임시 동적이 아닌 로컬 큐에 대해 실행됩니다.
  - DELETE QLOCAL
  - DEFINE/ALTER QLOCAL
- 오브젝트에 대한 변경이 없는 경우에도 임시 동적 큐의 경우와는 다른 MQSET 호출이 실행됩니다.

### 구성 이벤트가 생성되지 않을 때

구성 이벤트 메시지는 다음 상황에서 생성되지 않습니다.

- 명령 또는 MQSET 호출이 실패할 때
- 큐 관리자가 이벤트 큐에 구성 이벤트를 넣으려고 시도하는 오류가 발생할 때. 그 경우 명령 또는 MQSET 호출이 완료하지만 이벤트 메시지는 생성되지 않습니다.
- 임시 동적 큐의 경우
- 내부 변경이 TRIGGER 큐 속성에 작성될 때
- REFRESH QMGR 명령에 의해 경우를 제외하고, SYSTEM.ADMIN.CONFIG.EVENT 구성 이벤트 큐의 경우
- 클러스터링 변경을 유발하는 REFRESH/RESET CLUSTER 및 RESUME/SUSPEND QMGR 명령의 경우
- 큐 관리자를 작성 또는 삭제할 때

### 관련 개념

[프로그래밍 가능 명령 형식 소개](#)

33 페이지의 『구성 이벤트』

구성 이벤트는 오브젝트가 작성, 변경 또는 삭제될 때 생성되며 명시적 요청에 의해서도 생성될 수 있는 알림입니다.

### 관련 참조

[MQSC 명령](#)

[MQSET - 오브젝트 속성 설정](#)

## 구성 이벤트 사용법

시스템에 관한 정보를 얻기 위해 구성 이벤트를 사용할 수 있는 방법을 보고 CMDSCOPE 같이 구성 이벤트의 사용에 영향을 줄 수 있는 요인을 이해하려면 이 페이지를 사용하십시오.

다음 목적을 위해 구성 이벤트를 사용할 수 있습니다.

1. 보고서가 생성될 수 있으며 시스템의 구조에 관한 정보가 생성될 수 있는 중앙 구성 저장소를 생성하고 유지 보수하기 위해.
2. 감사 추적을 생성하기 위해. 예를 들어, 오브젝트가 예기치 않게 변경되는 경우 변경을 작성한 사용자와 작성된 시간에 관한 정보가 저장될 수 있습니다.

이것은 특히 명령 이벤트도 사용 가능할 때 유용할 수 있습니다. MQSC 또는 PCF 명령이 구성 이벤트 및 명령 이벤트가 생성되도록 하는 경우, 이벤트 메시지가 둘 다 메시지 디스크립터에서 동일한 상관 ID를 공유합니다.

MQSET 호출 또는 다음 명령 중 하나의 경우,

- DEFINE 오브젝트
- ALTER 오브젝트
- DELETE 오브젝트

큐 관리자 속성 CONFIGEV가 사용 가능하지만 구성 이벤트 메시지가 구성 이벤트 큐에 넣어질 수 없는 경우, 예를 들어 이벤트 큐가 정의되지 않은 경우 명령이나 MQSET 호출이 상관없이 실행됩니다.

## CMDSCOPE의 효과

CMDSCOPE가 사용되는 명령의 경우 구성 이벤트 메시지 또는 메시지들이 명령이 입력되는 곳이 아니라 명령이 실행되는 큐 관리자(들)에서 생성됩니다. 그러나 CMDSCOPE를 사용하는 명령이 소스 큐 관리자에 의해 생성된 명령인 경우에도 이벤트 데이터의 모든 원본 및 컨텍스트 정보가 입력된 원래 명령과 관련됩니다.

큐 공유 그룹이 현재 버전에 있지 않은 경우 큐 관리자를 포함하는 경우, 이전 버전에 있는 것이 아니라 현재 버전에 있는 큐 관리자에서 CMDSCOPE를 통해 실행되는 모든 명령에 대해 이벤트가 생성됩니다. 이것은 명령이 입력되는 큐 관리자가 이전 버전에 있는 경우에도 그런 경우에 컨텍스트 정보가 이벤트 데이터에 포함됨에도 불구하고 발생합니다.

### 관련 개념

[프로그래밍 가능 명령 형식 소개](#)

[33 페이지의 『구성 이벤트』](#)

구성 이벤트는 오브젝트가 작성, 변경 또는 삭제될 때 생성되며 명시적 요청에 의해서도 생성될 수 있는 알림입니다.

### 관련 참조

[MQSET - 오브젝트 속성 설정](#)

## 오브젝트 새로 고치기 구성 이벤트

오브젝트 새로 고치기 구성 이벤트는 명시적으로 요청될 때만 발생하기 때문에 다른 구성 이벤트와는 다릅니다.

작성, 변경 및 삭제 이벤트는 MQSET 호출이나 오브젝트를 변경하기 위한 명령에 의해 생성되지만 오브젝트 새로 고치기 이벤트는 MQSC 명령 REFRESH QMGR 또는 그의 동등한 PCF에 의해 명시적으로 요청될 때만 발생합니다.

REFRESH QMGR 명령은 구성 이벤트를 생성하는 다른 모든 명령과 다릅니다. 다른 모든 명령은 특정 오브젝트에 적용되며 해당 오브젝트에 대한 단일 구성 이벤트를 생성합니다. REFRESH QMGR 명령은 큐 관리자에 의해 저장되는 모든 오브젝트 정의를 잠재적으로 나타내는 많은 구성 이벤트 메시지를 생성할 수 있습니다. 선택되는 각 오브젝트에 대해 하나의 이벤트 메시지가 생성됩니다.

REFRESH QMGR 명령은 세 선택 기준의 조합을 사용하여 포함된 오브젝트의 수를 필터링합니다.

- 오브젝트 이름
- 오브젝트 유형
- 새로 고치기 간격

REFRESH QMGR 명령에 어떤 선택 기준도 지정하지 않는 경우, 각 선택 기준에 대해 기본값이 사용되며 큐 관리자에 의해 저장되는 모든 오브젝트 정의에 대해 구성 새로 고치기 이벤트 메시지가 생성됩니다. 이것은 허용할 수 없는 처리 시간 및 이벤트 메시지 생성을 유발할 수 있습니다. 몇 가지 선택 기준 지정을 고려하십시오.

새로 고치기 이벤트를 생성하는 REFRESH QMGR 명령은 다음 상황에서 사용될 수 있습니다.

- 오브젝트가 최근에 조작되었는지 여부와 상관없이 시스템에 있는 오브젝트의 일부 또는 전부에 관한 구성 데이터를 원하는 경우, 예를 들어 구성 이벤트가 처음 사용 가능할 때.  
각각이 서로 다른 오브젝트의 선택을 갖지만 모두가 포함되는 여러 명령의 사용을 고려하십시오.
- SYSTEM.ADMIN.CONFIG.EVENT 큐에 오류가 있었던 경우. 이 상황에서는 작성, 변경 또는 삭제 이벤트에 대해 구성 이벤트 메시지가 생성되지 않습니다. 큐의 오류가 정정되었을 때, 큐 관리자 새로 고치기 명령이 이벤트 메시지의 생성을 요청하는 데 사용될 수 있는데 큐에 오류가 있는 동안 유실되었습니다. 이 상황에서는 새로 고치기 간격을 큐가 사용 불가능한 시간으로 설정할 것을 고려하십시오.

### 관련 개념

[33 페이지의 『구성 이벤트』](#)

구성 이벤트는 오브젝트가 작성, 변경 또는 삭제될 때 생성되며 명시적 요청에 의해서도 생성될 수 있는 알림입니다.

### 관련 참조

[REFRESH QMGR](#)

[큐 관리자 새로 고치기](#)

## 명령 이벤트

명령 이벤트는 MQSC 또는 PCF 명령이 성공적으로 실행했다는 알림입니다.

이벤트 데이터는 다음 정보를 포함합니다.

### 원본 정보

명령이 실행된 큐 관리자, 명령을 실행한 사용자의 ID 및 명령이 실행된 방법(예를 들어, 콘솔 명령에 의해)으로 구성됩니다.

### 컨텍스트 정보

명령 메시지의 메시지 데이터의 컨텍스트 정보의 복제본입니다. 명령이 메시지를 사용하여 입력되지 않은 경우 컨텍스트 정보는 생략됩니다.

컨텍스트 정보는 명령이 SYSTEM.COMMAND.INPUT 큐에 메시지로 입력되었을 때만 이벤트 데이터에 포함됩니다.

### 명령 정보

실행된 명령의 유형입니다.

### 명령 데이터

- PCF 명령의 경우 명령 데이터의 복제본
- MQSC 명령의 경우 명령 텍스트

명령 데이터 형식이 원래 명령의 형식과 반드시 일치하지는 않습니다. 예를 들어, 분산 플랫폼에서 원래 요청이 MQSC 명령인 경우에도 명령 데이터 형식은 항상 PCF 형식입니다.

생성되는 모든 명령 이벤트 메시지는 명령 이벤트 큐 SYSTEM.ADMIN.COMMAND.EVENT에 위치합니다.

### 관련 참조

#### 명령

[7 페이지의 『이벤트 유형』](#)

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

## 명령 이벤트 생성

명령 이벤트가 생성되게 하는 상황을 보고 명령 이벤트가 생성되지 않는 상황을 이해하려면 이 페이지를 사용하십시오.

### 명령 이벤트가 생성되지 않을 때

명령 이벤트 메시지는 다음 상황에서 생성됩니다.

- CMDEV 큐 관리자 속성이 ENABLED로 지정되고 MQSC 또는 PCF 명령이 성공적으로 실행할 때.
- CMDEV 큐 관리자 속성이 NODISPLAY로 지정되고 DISPLAY 명령(MQSC) 및 Inquire 명령(PCF)을 제외한 모든 명령이 성공적으로 실행할 때.
- MQSC 명령 ALTER QMGR 또는 PCF 명령인 Change Queue Manager를 실행하고 CMDEV 큐 관리자 속성이 다음 조건 중 하나를 만족할 때:
  - 변경 후 CMDEV가 DISABLED로 지정됨
  - 변경 전 CMDEV가 DISABLED로 지정되지 않았음

명령이 명령 이벤트 큐 SYSTEM.ADMIN.COMMAND.EVENT에 대해 실행하는 경우, 큐가 아직 존재하고 put 금지되지 않은 경우 명령 이벤트가 생성됩니다.

### 명령 이벤트가 생성되지 않을 때

명령 이벤트 메시지는 다음 상황에서 생성되지 않습니다.

- 명령이 실패할 때
- 큐 관리자가 이벤트 큐에 명령 이벤트를 넣으려는 오류를 만날 때. 이 경우 상관없이 명령이 실행하지만 이벤트 메시지가 생성되지 않습니다.

- MQSC 명령 REFRESH QMGR TYPE (EARLY)의 경우
- MQSC 명령 START QMGR MQSC의 경우
- 매개변수 LOG가 지정되는 경우 MQSC 명령 SUSPEND QMGR의 경우
- 매개변수 LOG가 지정되는 경우 MQSC 명령 RESUME QMGR의 경우

#### 관련 개념

37 페이지의 『명령 이벤트』

명령 이벤트는 MQSC 또는 PCF 명령이 성공적으로 실행했다는 알림입니다.

#### 관련 참조

[REFRESH QMGR](#)

[SUSPEND QMGR](#)

[RESUME QMGR](#)

[SUSPEND QMGR, RESUME QMGR 및 클러스터](#)

## 명령 이벤트 사용법

명령 이벤트를 사용하여 실행한 명령의 감사 추적을 생성할 수 있는 방법을 보려면 이 페이지를 사용하십시오.

예를 들어, 오브젝트가 예기치 않게 변경되는 경우 변경을 작성한 사용자와 작성된 시간에 관한 정보가 저장될 수 있습니다. 이것은 특히 구성 이벤트도 사용 가능할 때 유용할 수 있습니다. MQSC 또는 PCF 명령이 명령 이벤트 및 구성 이벤트가 생성되도록 하는 경우, 이벤트 메시지가 둘 다 메시지 디스크립터에서 동일한 상관 ID를 공유합니다.

명령 이벤트 메시지가 생성되지만 명령 이벤트 큐에 넣을 수 없는 경우, 예를 들어 명령 이벤트 큐가 정의되지 않은 경우 명령 이벤트가 생성된 명령이 상관없이 아직 실행합니다.

## CMDSCOPE의 효과

CMDSCOPE가 사용되는 명령의 경우 명령 이벤트 메시지 또는 메시지들이 명령이 입력되는 곳이 아니라 명령이 실행하는 큐 관리자(들)에서 생성됩니다. 그러나 CMDSCOPE를 사용하는 명령이 소스 큐 관리자에 의해 생성된 명령인 경우에도 이벤트 데이터의 모든 원본 및 컨텍스트 정보가 입력된 원래 명령과 관련됩니다.

#### 관련 개념

37 페이지의 『명령 이벤트』

명령 이벤트는 MQSC 또는 PCF 명령이 성공적으로 실행했다는 알림입니다.

37 페이지의 『명령 이벤트 생성』

명령 이벤트가 생성되게 하는 상황을 보고 명령 이벤트가 생성되지 않는 상황을 이해하려면 이 페이지를 사용하십시오.

#### 관련 참조

[MQSC 명령](#)

[그룹 내의 PCF 명령 및 응답](#)

## 로거 이벤트

로거 이벤트는 큐 관리자가 새 로그 익스텐트에 작성을 시작했음을 알립니다.

이벤트 데이터는 다음 정보를 포함합니다.

- 현재 로그 익스텐트의 이름.
- 재시작 복구를 위해 필요한 가장 빠른 로그 익스텐트의 이름.
- 매체 복원을 위해 필요한 가장 이른 로그 범위의 이름.
- 로그 익스텐트가 위치하는 디렉토리.

생성되는 모든 로거 이벤트 메시지는 로거 이벤트 큐 SYSTEM.ADMIN.LOGGER.EVENT에 위치합니다.

#### 관련 참조

[로거](#)

## [7 페이지의 『이벤트 유형』](#)

큐 관리자 또는 채널 인스턴스가 보고할 수 있는 도구 이벤트의 유형을 보려면 이 페이지를 사용하십시오.

## 로거 이벤트 생성

로거 이벤트가 생성되게 하는 상황을 보고 로거 이벤트가 생성되지 않는 상황을 이해하려면 이 페이지를 사용하십시오.

다음 상황에서 로거 이벤트 메시지가 생성됩니다.

- LOGGEREV 큐 관리자 속성이 ENABLED로 지정되고 큐 관리자가 새 로그 익스텐트 또는 IBM i에서는 저널 수신자에 쓰기 시작할 때.
- LOGGEREV 큐 관리자 속성이 ENABLED로 지정되고 큐 관리자가 시작할 때.
- LOGGEREV 큐 관리자 속성이 DISABLED에서 ENABLED로 변경될 때.

**팁:** RESET QMGR MQSC 명령을 사용하여 큐 관리자가 새 로그 익스텐트에 쓰기 시작하도록 요청할 수 있습니다.

## 로거 이벤트가 생성되지 않을 때

로거 이벤트 메시지는 다음 상황에서 생성되지 않습니다.

- 큐 관리자가 순환 로깅을 사용하도록 구성될 때.  
이 경우 LOGGEREV 큐 관리자 속성이 DISABLED로 설정되며 변경할 수 없습니다.
- 큐 관리자가 이벤트 큐에 로거 이벤트를 넣으려는 오류를 만날 때. 이 경우 이벤트를 유발한 조치가 완료되지만 이벤트 메시지가 생성되지 않습니다.

### 관련 개념

#### [38 페이지의 『로거 이벤트』](#)

로거 이벤트는 큐 관리자가 새 로그 익스텐트에 작성을 시작했음을 알립니다.

### 관련 참조

[LoggerEvent\(MQLONG\)](#)

[RESET QMGR](#)

## 로거 이벤트 사용법

로거 이벤트를 사용하여 큐 관리자 재시작 또는 매체 복원에 더 이상 필요하지 않은 로그 익스텐트를 판별할 수 있는 방법을 보려면 이 페이지를 사용하십시오.

활성 로그 디렉토리로부터 제거하기 전에 재해 복구를 위해 테이프 같은 매체로 여분의 로그 익스텐트를 아카이브할 수 있습니다. 여분의 로그 익스텐트의 정기적인 제거는 디스크 공간 사용량을 최소로 유지합니다.

예를 들어 이벤트 큐가 정의되지 않았기 때문에 LOGGEREV 큐 관리자 속성이 사용 가능하지만 로거 이벤트 메시지를 로거 이벤트 큐에 넣을 수 없는 경우, 해당 이벤트를 유발한 조치는 상관없이 계속됩니다.

### 관련 개념

#### [38 페이지의 『로거 이벤트』](#)

로거 이벤트는 큐 관리자가 새 로그 익스텐트에 작성을 시작했음을 알립니다.

### 관련 참조

[LoggerEvent\(MQLONG\)](#)

#### [39 페이지의 『로거 이벤트 생성』](#)

로거 이벤트가 생성되게 하는 상황을 보고 로거 이벤트가 생성되지 않는 상황을 이해하려면 이 페이지를 사용하십시오.

## 로거 이벤트 큐를 모니터링하는 샘플 프로그램

새 이벤트 메시지에 대한 로거 이벤트 큐를 모니터링하고 해당 메시지를 읽고 메시지의 콘텐츠를 stdout에 넣는 샘플 C 프로그램을 보려면 이 페이지를 사용하십시오.

```

/*****/
/*
/* Program name: AMQSLOG0.C
/*
/* Description: Sample C program to monitor the logger event queue and output
/* a message to stdout when a logger event occurs
/* <N_OCO_COPYRIGHT>
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 2005, 2024. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/* <NOC_COPYRIGHT>
/*****/
/*
/* Function: AMQSLOG is a sample program which monitors the logger event
/* queue for new event messages, reads those messages, and puts the contents
/* of the message to stdout.
/*
/*****/
/*
/* AMQSLOG has 1 parameter - the queue manager name (optional, if not
/* specified then the default queue manager is implied)
/*
/*****/

/*****/
/* Includes
/*****/
#include <stdlib.h>
#include <stdio.h>
#include <string.h>

#include <cmqc.h> /* MQI constants*/
#include <cmqcfc.h> /* PCF constants*/

/*****/
/* Constants
/*****/

#define MAX_MESSAGE_LENGTH 8000

typedef struct _ParmTableEntry
{
    MQLONG ConstVal;
    PMQCHAR Desc;
} ParmTableEntry;

ParmTableEntry ParmTable[] =
{
    0, "",
    MQCA_Q_MGR_NAME, "Queue Manager Name",
    MQCMD_LOGGER_EVENT, "Logger Event Command",
    MQRC_LOGGER_STATUS, "Logger Status",
    MQCACF_CURRENT_LOG_EXTENT_NAME, "Current Log Extent",
    MQCACF_RESTART_LOG_EXTENT_NAME, "Restart Log Extent",
    MQCACF_MEDIA_LOG_EXTENT_NAME, "Media Log Extent",
    MQCACF_LOG_PATH, "Log Path"};

/*****/
/* Function prototypes
/*****/

static void ProcessPCF(MQHCONN hConn,
                      MQHOBJ hEventQueue,
                      PMQCHAR pBuffer);

static PMQCHAR ParmToString(MQLONG Parameter);

/*****/
/* Function: main
/*****/
int main(int argc, char * argv[])
{
    MQLONG CompCode;
    MQLONG Reason;
    MQHCONN hConn = MQHC_UNUSABLE_HCONN;
    MQOD ObjDesc = { MQOD_DEFAULT };

```

```

MQCHAR    QMName[MQ_Q_MGR_NAME_LENGTH+1] = "";
MQCHAR    LogEvQ[MQ_Q_NAME_LENGTH] = "SYSTEM.ADMIN.LOGGER.EVENT";
MQHOBJS   hEventQueue;
PMQCHAR    pBuffer = NULL;

printf("\n/*****\n");
printf("/* Sample Logger Event Monitor start */\n");
printf("/*****\n");

/*****/
/* Parse any command line options */
/*****/

if (argc > 1)
    stncpy(QMName, argv[1], (size_t)MQ_Q_MGR_NAME_LENGTH);

pBuffer = (char *)malloc(MAX_MESSAGE_LENGTH);
if (!pBuffer)
{
    printf("Can't allocate %d bytes\n",MAX_MESSAGE_LENGTH);
    goto MOD_EXIT;
}

/*****/
/* Connect to the specified (or default) queue manager */
/*****/

MQCONN(QMName,
        &hConn,
        &CompCode,
        &Reason);

if (Reason != MQCC_OK)
{
    printf("Error in call to MQCONN, Reason %d, CompCode %d\n", Reason,
        CompCode);
    goto MOD_EXIT;
}

/* Open the logger event queue for input */

stncpy(ObjDesc.ObjectQMgrName,QMName, MQ_Q_MGR_NAME_LENGTH);
stncpy(ObjDesc.ObjectName, LogEvQ, MQ_Q_NAME_LENGTH);

MQOPEN( hConn,
        &ObjDesc,
        MQOO_INPUT_EXCLUSIVE,
        &hEventQueue,
        &CompCode,
        &Reason);

if (Reason)
{
    printf("MQOPEN failed for queue manager %.48s Queue %.48s Reason: %d\n",
        ObjDesc.ObjectQMgrName,
        ObjDesc.ObjectName,
        Reason);

    goto MOD_EXIT;
}
else
{
    ProcessPCF(hConn, hEventQueue, pBuffer);
}

MOD_EXIT:

if (pBuffer != NULL) {
    free(pBuffer);
}

/*****/
/* Disconnect */
/*****/
if (hConn != MQHC_UNUSABLE_HCONN) {
    MQDISC(&hConn, &CompCode, &Reason);
}

return 0;
}

/*****/
/* Function: ProcessPCF */
/*****/

```

```

/*                                                                    */
/* Input Parameters:  Handle to queue manager connection                */
/*                                                                    */
/*                   Handle to the opened logger event queue object     */
/*                                                                    */
/*                   Pointer to a memory buffer to store the incoming PCF msg*/
/*                                                                    */
/* Output Parameters: None                                             */
/*                                                                    */
/* Logic: Wait for messages to appear on the logger event queue and display */
/* their contents.                                                    */
/*                                                                    */
/*                                                                    */
/*****

static void ProcessPCF(MQHCONN    hConn,
                     MQHOBJ     hEventQueue,
                     PMQCHAR    pBuffer)
{
    MQCFH    * pCfh;
    MQCFST   * pCfst;
    MQGMO    Gmo      = { MQGMO_DEFAULT };
    MQMD     Mqmd     = { MQMD_DEFAULT };
    PMQCHAR  pPCFCmd;
    MQLONG   Reason   = 0;
    MQLONG   CompCode;
    MQLONG   MsgLen;
    PMQCHAR  Parm = NULL;

    Gmo.Options    |= MQGMO_WAIT;
    Gmo.Options    |= MQGMO_CONVERT;
    Gmo.WaitInterval = MQWI_UNLIMITED;
    /*****
    /* Process response Queue
    /*****
    while (Reason == MQCC_OK)
    {
        memcpy(&Mqmd.MsgId;    , MQMI_NONE, sizeof(Mqmd.MsgId));
        memset(&Mqmd.CorrelId, 0, sizeof(Mqmd.CorrelId));

        MQGET( hConn,
              hEventQueue,
              &Mqmd,
              &Gmo,
              MAX_MESSAGE_LENGTH,
              pBuffer,
              &MsgLen,
              &CompCode,
              &Reason);
        if (Reason != MQCC_OK)
        {
            switch(Reason)
            {
                case MQRC_NO_MSG_AVAILABLE:
                    printf("Timed out");
                    break;

                default:
                    printf("MQGET failed RC(%d)\n", Reason);
                    break;
            }
            goto MOD_EXIT;
        }

        /*****
        /* Only expect PCF event messages on this queue
        /*****
        if (memcmp(Mqmd.Format, MQFMT_EVENT, sizeof(Mqmd.Format)))
        {
            printf("Unexpected message format '%8s' received\n",Mqmd.Format);
            continue;
        }

        /*****
        /* Build the output by parsing the received PCF message, first the */
        /* header, then each of the parameters
        /*****

        pCfh = (MQCFH *)pBuffer;

        if (pCfh -> Reason)
        {
            printf("-----\n");
            printf("Event Message Received\n");

```

```

Parm = ParmToString(pCfh->Command);
if (Parm != NULL) {
    printf("Command  :%s \n",Parm);
}
else
{
    printf("Command  :%d \n",pCfh->Command);
}

printf("CompCode :%d\n"      ,pCfh->CompCode);

Parm = ParmToString(pCfh->Reason);
if (Parm != NULL) {
    printf("Reason   :%s \n",Parm);
}
else
{
    printf("Reason   :%d \n",pCfh->Reason);
}
}

pPCFCmd = (char *) (pCfh+1);
printf("-----\n");
while(pCfh -> ParameterCount--)
{
    pCfst = (MQCFST *) pPCFCmd;
    switch(pCfst -> Type)
    {
        case MQCFT_STRING:
            Parm = ParmToString(pCfst -> Parameter);
            if (Parm != NULL) {
                printf("%-32s",Parm);
            }
            else
            {
                printf("%-32d",pCfst -> Parameter);
            }

            fwrite( pCfst -> String, pCfst -> StringLength, 1, stdout);
            pPCFCmd += pCfst -> Struclength;
            break;
        default:
            printf("Unrecognised datatype %d returned\n",pCfst->Type);
            goto MOD_EXIT;
    }
    putchar('\n');
}
printf("-----\n");
}
MOD_EXIT:
return;
}

/*****
*/
/* Function: ParmToString
*/
/*****
*/
/* Input Parameters: Parameter for which to get string description
*/
/*
*/
/* Output Parameters: None
*/
/*
*/
/* Logic: Takes a parameter as input and returns a pointer to a string
*/
/* description for that parameter, or NULL if the parameter does not
*/
/* have an associated string description
*/
/*****

static PMQCHAR ParmToString(MQLONG Parameter){
    long i;
    for (i=0 ; i< sizeof(ParmTable)/sizeof(ParmTableEntry); i++)
    {
        if (ParmTable[i].ConstVal == Parameter ParmTable[i].Desc)
            return ParmTable[i].Desc;
    }
    return NULL;
}

```

## 샘플 출력

이 애플리케이션은 다음 양식의 출력을 생성합니다.

```

/*****
/* Sample Logger Event Monitor start */
/*****
-----
Event Message Received
Command :Logger Event Command
CompCode :0
Reason :Logger Status
-----
Queue Manager Name          CSIM

Current Log Extent          AMQA000001
Restart Log Extent         AMQA000001
Media Log Extent           AMQA000001
Log Path                    QMCSIM
-----

```

### 관련 개념

#### 39 페이지의 『로거 이벤트 사용법』

로거 이벤트를 사용하여 큐 관리자 재시작 또는 매체 복원에 더 이상 필요하지 않은 로그 익스텐트를 판별할 수 있는 방법을 보려면 이 페이지를 사용하십시오.

#### 38 페이지의 『명령 이벤트 사용법』

명령 이벤트를 사용하여 실행한 명령의 감사 추적을 생성할 수 있는 방법을 보려면 이 페이지를 사용하십시오.

### 관련 참조

#### 39 페이지의 『로거 이벤트 생성』

로거 이벤트가 생성되게 하는 상황을 보고 로거 이벤트가 생성되지 않는 상황을 이해하려면 이 페이지를 사용하십시오.

## 도구 이벤트를 모니터링하는 샘플 프로그램

도구 이벤트를 모니터링하기 위해 샘플 C 프로그램을 보려면 이 페이지를 사용하십시오.

이 샘플 프로그램은 IBM WebSphere MQ 제품의 일부가 아니므로 실제 실제 항목으로 제공되지 않습니다. 이 예는 지정된 조치의 모든 가능한 결과를 열거하지 않는다는 점에서 불완전합니다. 그러나, 이벤트를 사용하는 사용자 고유의 프로그램, 특히 이벤트 메시지에서 사용되는 PCF 형식을 위한 기초로 이 샘플을 사용할 수 있습니다. 그러나 사용자 자신의 시스템에서 실행하기 전에 이 프로그램을 수정해야 합니다.

```

/*****
/*
/* Program name: EVMON
/*
/* Description: C program that acts as an event monitor
/*
/*
/*
/*****
/*
/* Function:
/*
/*
/* EVMON is a C program that acts as an event monitor - reads an
/* event queue and tells you if anything appears on it
/*
/* Its first parameter is the queue manager name, the second is
/* the event queue name. If these are not supplied it uses the
/* defaults.
/*
/*
/*****
#include <time.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#ifndef min
#define min(a,b) ((a) < (b)) ? (a) : (b)
#endif

/*****

```

```

/* includes for MQI                                                                    */
/*****                                                                    */
#include <cmqc.h>
#include <cmqcfc.h>
void printfmqcfst(MQCFST* pmqcfst);
void printfmqcfin(MQCFIN* pmqcfst);
void printreas(MQLONG reason);

#define PRINTREAS(param) \
    case param: \
        printf("Reason = %s\n", #param); \
        break;

/*****                                                                    */
/* global variable                                                                */
/*****                                                                    */
MQCFH    *evtmsg;          /* evtmsg message buffer */

int main(int argc, char **argv)
{
/*****                                                                    */
/* declare variables                                                            */
/*****                                                                    */
int i;          /* auxiliary counter */
/*****                                                                    */
/* Declare MQI structures needed                                              */
/*****                                                                    */
MQOD    od = {MQOD_DEFAULT};      /* Object Descriptor */
MQMD    md = {MQMD_DEFAULT};      /* Message Descriptor */
MQGMO   gmo = {MQGMO_DEFAULT};    /* get message options */
/*****                                                                    */
/* note, uses defaults where it can                                          */
/*****                                                                    */

MQHCONN Hcon;          /* connection handle */
MQHOBJ  Hobj;          /* object handle */
MQLONG  O_options;    /* MQOPEN options */
MQLONG  C_options;    /* MQCLOSE options */
MQLONG  CompCode;     /* completion code */
MQLONG  OpenCode;     /* MQOPEN completion code */
MQLONG  Reason;       /* reason code */
MQLONG  CReason;      /* reason code for MQCONN */
MQLONG  buflen;       /* buffer length */
MQLONG  evtmsglen;    /* message length received */
MQCHAR  command[1100]; /* call command string ... */
MQCHAR  p1[600];      /* ApplId insert */
MQCHAR  p2[900];      /* evtmsg insert */
MQCHAR  p3[600];      /* Environment insert */
MQLONG  mytype;       /* saved application type */
char    QMName[50];   /* queue manager name */
MQCFST  *paras;       /* the parameters */
int      counter;     /* loop counter */
time_t   ltime;

/*****                                                                    */
/* Connect to queue manager                                                    */
/*****                                                                    */
QMName[0] = 0;          /* default queue manager */
if (argc > 1)
    strcpy(QMName, argv[1]);
MQCONN(QMName,          /* queue manager */
        &Hcon,          /* connection handle */
        &CompCode,     /* completion code */
        &CReason);    /* reason code */

/*****                                                                    */
/* Initialize object descriptor for subject queue                            */
/*****                                                                    */
strcpy(od.ObjectName, "SYSTEM.ADMIN.QMGR.EVENT");
if (argc > 2)
    strcpy(od.ObjectName, argv[2]);

/*****                                                                    */
/* Open the event queue for input; exclusive or shared. Use of */
/* the queue is controlled by the queue definition here */
/*****                                                                    */

O_options = MQOO_INPUT_AS_Q_DEF /* open queue for input */

```

```

+ MQOO_FAIL_IF QUIESCING      /* but not if qmgr stopping */
+ MQOO_BROWSE;
MQOPEN(Hcon,                    /* connection handle */
       &od,                     /* object descriptor for queue*/
       0_options,              /* open options */
       &Hobj,                   /* object handle */
       &CompCode,              /* completion code */
       &Reason);               /* reason code */

/*****
/* Get messages from the message queue */
/*****
while (CompCode != MQCC_FAILED)
{
/*****
/* I don't know how big this message is so just get the
/* descriptor first */
/*****
gmo.Options = MQGMO_WAIT + MQGMO_LOCK
+ MQGMO_BROWSE_FIRST + MQGMO_ACCEPT_TRUNCATED_MSG;

gmo.WaitInterval = MQWI_UNLIMITED; /* no time limit */
buflen = 0; /* amount of message to get */

/*****
/* clear selectors to get messages in sequence */
/*****
memcpy(md.MsgId, MQMI_NONE, sizeof(md.MsgId));
memcpy(md.CorrelId, MQCI_NONE, sizeof(md.CorrelId));

/*****
/* wait for event message */
/*****
printf("...>\n");
MQGET(Hcon,                    /* connection handle */
      Hobj,                    /* object handle */
      &md,                    /* message descriptor */
      &gmo,                   /* get message options */
      buflen,                  /* buffer length */
      evtmsg,                  /* evtmsg message buffer */
      &evtmsglen,             /* message length */
      &CompCode,              /* completion code */
      &Reason);               /* reason code */

/*****
/* report reason, if any */
/*****
if (Reason != MQRC_NONE && Reason != MQRC_TRUNCATED_MSG_ACCEPTED)
{
printf("MQGET ==> %ld\n", Reason);
}
else
{
gmo.Options = MQGMO_NO_WAIT + MQGMO_MSG_UNDER_CURSOR;
buflen = evtmsglen; /* amount of message to get */
evtmsg = malloc(buflen);
if (evtmsg != NULL)
{
/*****
/* clear selectors to get messages in sequence */
/*****
memcpy(md.MsgId, MQMI_NONE, sizeof(md.MsgId));
memcpy(md.CorrelId, MQCI_NONE, sizeof(md.CorrelId));

/*****
/* get the event message */
/*****
printf("...>\n");
MQGET(Hcon,                    /* connection handle */
      Hobj,                    /* object handle */
      &md,                    /* message descriptor */
      &gmo,                   /* get message options */
      buflen,                  /* buffer length */
      evtmsg,                  /* evtmsg message buffer */
      &evtmsglen,             /* message length */
      &CompCode,              /* completion code */
      &Reason);               /* reason code */

/*****
/* report reason, if any */
/*****

```

```

/*****
if (Reason != MQRC_NONE)
{
    printf("MQGET ==> %ld\n", Reason);
}
}
else
{
    CompCode = MQCC_FAILED;
}
}

/*****
/* . . . process each message received */
/*****

if (CompCode != MQCC_FAILED)
{
/*****
/* announce a message */
/*****
printf("\a\a\a\a\a");
time(&ltime);
printf(ctime(&ltime));

if (evtmsglen != buflen)
    printf("DataLength = %ld?\n", evtmsglen);
else
{
/*****
/* right let's look at the data */
/*****
if (evtmsg->Type != MQCFT_EVENT)
{
    printf("Something's wrong this isn't an event message,"
           " its type is %ld\n",evtmsg->Type);
}
else
{
    if (evtmsg->Command == MQCMD_Q_MGR_EVENT)
    {
        printf("Queue Manager event: ");
    }
    else
    {
        if (evtmsg->Command == MQCMD_CHANNEL_EVENT)
        {
            printf("Channel event: ");
        }
        else
        :

        {
            printf("Unknown Event message, %ld.",
                   evtmsg->Command);
        }

        if (evtmsg->CompCode == MQCC_OK)
            printf("CompCode(OK)\n");
        else if (evtmsg->CompCode == MQCC_WARNING)
            printf("CompCode(WARNING)\n");
        else if (evtmsg->CompCode == MQCC_FAILED)
            printf("CompCode(FAILED)\n");
        else
            printf("* CompCode wrong * (%ld)\n",
                   evtmsg->CompCode);

        if (evtmsg->StrucLength != MQCFH_STRUC_LENGTH)
        {
            printf("it's the wrong length, %ld\n",evtmsg->StrucLength);
        }

        if (evtmsg->Version != MQCFH_VERSION_1)
        {
            printf("it's the wrong version, %ld\n",evtmsg->Version);
        }

        if (evtmsg->MsgSeqNumber != 1)
        {
            printf("it's the wrong sequence number, %ld\n",

```

```

        evtmsg->MsgSeqNumber);
    }

    if (evtmsg->Control != MQCFC_LAST)
    {
        printf("it's the wrong control option, %ld\n",
            evtmsg->Control);
    }

    printreas(evtmsg->Reason);
    printf("parameter count is %ld\n", evtmsg->ParameterCount);
    /******
    /* get a pointer to the start of the parameters */
    /******

    paras = (MQCFST *) (evtmsg + 1);
    counter = 1;
    while (counter <= evtmsg->ParameterCount)
    {
        switch (paras->Type)
        {
            case MQCFT_STRING:
                printfmqcfst(paras);
                paras = (MQCFST *) ((char *) paras
                    + paras->StrucLength);
                break;
            case MQCFT_INTEGER:
                printfmqcfint((MQCFIN*) paras);
                paras = (MQCFST *) ((char *) paras
                    + paras->StrucLength);
                break;
            default:
                printf("unknown parameter type, %ld\n",
                    paras->Type);
                counter = evtmsg->ParameterCount;
                break;
        }
        counter++;
    }
} /* end evtmsg action */
free(evtmsg);
evtmsg = NULL;
} /* end process for successful GET */
} /* end message processing loop */

/******
/* close the event queue - if it was opened */
/******
if (OpenCode != MQCC_FAILED)
{
    C_options = 0; /* no close options */
    MQCLOSE(Hcon, /* connection handle */
        &Hobj, /* object handle */
        C_options,
        &CompCode, /* completion code */
        &Reason); /* reason code */
    /******
    /* Disconnect from queue manager (unless previously connected) */
    /******
    if (CReason != MQRC_ALREADY_CONNECTED)
    {
        MQDISC(&Hcon, /* connection handle */
            &CompCode, /* completion code */
            &Reason); /* reason code */
    }

    /******
    /* END OF EVMON */
    /******
}

#define PRINTPARAM(param) \
    case param: \
    { \
        char *p = #param; \
        stncpy(thestring, pmqcfst->String, min(sizeof(thestring), \
            pmqcfst->StringLength)); \
        printf("%s %s\n", p, thestring); \
    }

```

```

    }
    break;

#define PRINTAT(param)
    case param:
        printf("MQIA_APPL_TYPE = %s\n",#param);
        break;

void printfmqcfst(MQCFST* pmqcfst)
{
    char thestring[100];

    switch (pmqcfst->Parameter)
    {
        PRINTPARAM(MQCA_BASE_Q_NAME)
        PRINTPARAM(MQCA_PROCESS_NAME)
        PRINTPARAM(MQCA_Q_MGR_NAME)
        PRINTPARAM(MQCA_Q_NAME)
        PRINTPARAM(MQCA_XMIT_Q_NAME)
        PRINTPARAM(MQCACF_APPL_NAME)
        :
        default:
            printf("Invalid parameter, %ld\n",pmqcfst->Parameter);
            break;
    }
}

void printfmqcfin(MQCFIN* pmqcfst)
{
    switch (pmqcfst->Parameter)
    {
        case MQIA_APPL_TYPE:
            switch (pmqcfst->Value)
            {
                PRINTAT(MQAT_UNKNOWN)
                PRINTAT(MQAT_OS2)
                PRINTAT(MQAT_DOS)
                PRINTAT(MQAT_UNIX)
                PRINTAT(MQAT_QMGR)
                PRINTAT(MQAT_OS400)
                PRINTAT(MQAT_WINDOWS)
                PRINTAT(MQAT_CICS_VSE)
                PRINTAT(MQAT_VMS)
                PRINTAT(MQAT_GUARDIAN)
                PRINTAT(MQAT_VOS)
            }
            break;
        case MQIA_Q_TYPE:
            if (pmqcfst->Value == MQQT_ALIAS)
            {
                printf("MQIA_Q_TYPE is MQQT_ALIAS\n");
            }
            else
            :
            {
                if (pmqcfst->Value == MQQT_REMOTE)
                {
                    printf("MQIA_Q_TYPE is MQQT_REMOTE\n");
                    if (evtmsg->Reason == MQRC_ALIAS_BASE_Q_TYPE_ERROR)
                    {
                        printf("but remote is not valid here\n");
                    }
                }
                else
                {
                    printf("MQIA_Q_TYPE is wrong, %ld\n",pmqcfst->Value);
                }
            }
            break;

            case MQIACF_REASON_QUALIFIER:
                printf("MQIACF_REASON_QUALIFIER %ld\n",pmqcfst->Value);
                break;

            case MQIACF_ERROR_IDENTIFIER:
                printf("MQIACF_ERROR_IDENTIFIER %ld (X'%1X')\n",

```

```

        pmqcfst->Value,pmqcfst->Value);
    break;

    case MQIACF_AUX_ERROR_DATA_INT_1:
        printf("MQIACF_AUX_ERROR_DATA_INT_1 %ld (X'%lX')\n",
            pmqcfst->Value,pmqcfst->Value);
        break;

    case MQIACF_AUX_ERROR_DATA_INT_2:
        printf("MQIACF_AUX_ERROR_DATA_INT_2 %ld (X'%lX')\n",
            pmqcfst->Value,pmqcfst->Value);
        break;
    :
default :
    printf("Invalid parameter, %ld\n",pmqcfst->Parameter);
    break;
}
}

void printreas(MQLONG reason)
{
    switch (reason)
    {
        PRINTREAS(MQRCCF_CFH_TYPE_ERROR)
        PRINTREAS(MQRCCF_CFH_LENGTH_ERROR)
        PRINTREAS(MQRCCF_CFH_VERSION_ERROR)
        PRINTREAS(MQRCCF_CFH_MSG_SEQ_NUMBER_ERR)
        :
        PRINTREAS(MQRC_NO_MSG_LOCKED)
        PRINTREAS(MQRC_CONNECTION_NOT_AUTHORIZED)
        PRINTREAS(MQRC_MSG_TOO_BIG_FOR_CHANNEL)
        PRINTREAS(MQRC_CALL_IN_PROGRESS)
        default:
            printf("It's an unknown reason, %ld\n",
                reason);
            break;
    }
}
}
}

```

## 관련 개념

### 5 페이지의 『도구 이벤트』

도구 이벤트는 큐 관리자나 채널 인스턴스가 이벤트 큐에서 이벤트 메시지라는 특수 메시지를 감지하여 넣는 조건의 논리 결합입니다.

### 5 페이지의 『이벤트 모니터링』

이벤트 모니터링은 큐 관리자 네트워크에서 도구 이벤트 발생을 감지하는 프로세스입니다. 도구 이벤트는 큐 관리자나 채널 인스턴스에서 감지되는 이벤트의 논리적 조합입니다. 이러한 이벤트 때문에 큐 관리자나 채널 인스턴스가 이벤트 메시지라는 특정 메시지를 이벤트 큐에 넣게 됩니다.

## 관련 참조

### C 프로그래밍

#### 39 페이지의 『로거 이벤트 큐를 모니터링하는 샘플 프로그램』

새 이벤트 메시지에 대한 로거 이벤트 큐를 모니터링하고 해당 메시지를 읽고 메시지의 콘텐츠를 stdout에 넣는 샘플 C 프로그램을 보려면 이 페이지를 사용하십시오.

## 메시지 모니터링

메시지 모니터링은 큐 관리자 네트워크를 통해 메시지를 가져오는 라우트를 식별하는 프로세스입니다. 활동 유형 및 메시지를 위해 수행된 활동의 순서를 식별하여, 메시지 라우트를 판별할 수 있습니다.

메시지는 큐 관리자 네트워크를 통해 전달되면 다양한 프로세스가 메시지를 위해 활동을 수행합니다. 다음 기술 중 하나를 사용하여 메시지 라우트를 판별하십시오.

- IBM WebSphere MQ 표시 라우트 애플리케이션(dspmqrte)
- 활동 기록
- 라우트 추적 메시징

이 기술 모두는 큐 관리자 네트워크를 통해 전달될 때 메시지에 수행된 활동에 관한 정보를 포함한 특수 메시지를 생성합니다. 이 특수 메시지에서 리턴된 정보를 사용하여 다음 목표를 이룹니다.

- 메시지 활동을 기록합니다.
- 마지막으로 알려진 메시지 위치를 판별합니다.
- 큐 관리자 네트워크에서 라우팅 문제를 감지합니다.
- 큐 관리자 네트워크에서 라우팅 문제의 원인 판별을 돕습니다.
- 큐 관리자 네트워크가 올바르게 실행 중인지 확인합니다.
- 큐 관리자 네트워크의 실행에 익숙해집니다.
- 발행된 메시지를 추적합니다.

## 관련 개념

[메시지의 유형](#)

## 활동 및 조작

활동은 애플리케이션이 메시지를 위해 수행하는 별개의 조치입니다. 활동은 조작으로 구성되는데, 조작은 애플리케이션이 수행하는 작업의 단일 부분입니다.

다음 조치가 활동의 예입니다.

- 메시지 채널 에이전트(MCA)는 전송 큐에서 채널을 따라 메시지를 송신합니다.
- MCA가 채널에서 메시지를 받아서 대상 큐에 넣습니다.
- 애플리케이션이 큐에서 메시지를 가져와서 응답의 응답 메시지에 놓습니다.
- WebSphere MQ 발행/구독 엔진이 메시지를 처리합니다.

활동은 하나 이상의 조작으로 구성됩니다. 조작은 애플리케이션이 수행하는 작업의 단일 조각입니다. 예를 들어, 전송 큐에서 채널을 따라 메시지를 송신하는 MCA의 활동은 다음 조작으로 구성됩니다.

1. 전송 큐에서 메시지 가져오기(*Get* 조작).
2. 채널을 따라 메시지 송신(송신 조작).

발행/구독 네트워크에서 WebSphere MQ 발행/구독 엔진 처리 메시지의 활동은 다음과 같은 다중 조작으로 구성될 수 있습니다.

1. 메시지를 토픽 문자열에 넣기(*Put* 조작).
2. 메시지의 수령으로 간주되는 각 구독자에 대한 하나 이상의 조작(발행 조작, 제거된 발행 조작 또는 제외된 발행 조작).

## 활동의 정보

메시지가 큐 관리자 네트워크를 통해 라우트되므로 정보를 기록하여 메시지에 수행되는 활동의 순서를 식별할 수 있습니다. 메시지에 대해 수행되는 활동의 순서로부터 큐 관리자 네트워크를 통해 메시지의 라우트를 판별할 수 있으며, 다음 정보를 얻을 수 있습니다.

### 메시지의 마지막으로 알려진 위치

메시지가 그의 의도된 목적지에 도달하지 않는 경우, 전체 또는 부분적 메시지 라우트로부터 메시지의 마지막 알려진 위치를 판별할 수 있습니다.

### 큐 관리자 네트워크의 구성 문제

큐 관리자 네트워크를 통한 메시지의 라우트를 연구할 때, 메시지가 예상한 곳으로 이동하지 않았음을 확인할 수 있습니다. 이것이 발생할 수 있는 많은 이유가 있는데, 예를 들어 채널이 비활성인 경우 메시지는 대체 라우트를 택할 수 있습니다.

발행/구독 애플리케이션의 경우, 구독자에게 발행되는 결과로 큐 관리자 네트워크에서 이동하는 토픽 및 모든 메시지에 발행되는 메시지의 라우트를 판별할 수도 있습니다.

그런 상황에서 시스템 관리자는 큐 관리자 네트워크에 문제가 있는지 여부를 판별할 수 있으며, 문제가 있는 경우 이를 정정할 수 있습니다.

## 메시지 라우트

메시지 라우트를 판별하는 이유에 따라서, 다음의 일반 접근법을 사용할 수 있습니다.

### 라우트 추적 메시지를 위해 기록된 활동 정보 사용

라우트 추적 메시지는 특정 목적을 위한 활동 정보를 기록합니다. 이들을 사용하여 큐 관리자 네트워크의 구성 문제를 판별하거나 메시지의 마지막으로 알려진 위치를 판별할 수 있습니다. 의도된 목적지에 도달하지 않은 메시지의 마지막으로 알려진 위치를 판별하기 위해 라우트 추적 메시지가 생성되는 경우 원래 메시지를 흉내낼 수 있습니다. 이것은 라우트 추적 메시지에 원래 메시지가 취한 라우트를 따르는 가장 큰 기회를 제공합니다.

WebSphere MQ 라우트 표시 애플리케이션은 라우트 추적 메시지를 생성할 수 있습니다.

### 원래 메시지를 위해 기록된 활동 정보 사용

활동 기록을 위해 모든 메시지를 사용 가능하게 하고 그 대신 기록되는 활동 정보를 가질 수 있습니다. 메시지가 의도된 목적지에 도달하지 않는 경우, 기록된 활동 정보를 사용하여 메시지의 마지막으로 알려진 위치를 판별할 수 있습니다. 원래 메시지의 활동 정보를 사용하면 마지막으로 알려진 위치로 이끄는 가장 정확한 가능한 메시지 라우트를 판별할 수 있습니다. 이 접근법을 사용하려면 원래 메시지가 활동 기록에 대해 사용 가능해야 합니다.

**경고:** 활동 기록을 위해 큐 관리자 네트워크의 모든 메시지를 사용 가능하게 하는 것은 피하십시오. 활동 기록을 위해 사용 가능한 메시지는 그 대신 생성되는 많은 활동 보고서를 가질 수 있습니다. 큐 관리자 네트워크의 모든 메시지가 활동 기록에 대해 사용 가능한 경우, 큐 관리자 네트워크 트래픽이 허용할 수 없는 레벨로 증가할 수 있습니다.

### 관련 개념

#### [50 페이지의 『메시지 모니터링』](#)

메시지 모니터링은 큐 관리자 네트워크를 통해 메시지를 가져오는 라우트를 식별하는 프로세스입니다. 활동 유형 및 메시지를 위해 수행된 활동의 순서를 식별하여, 메시지 라우트를 판별할 수 있습니다.

#### [52 페이지의 『메시지 라우트 기술』](#)

활동 기록 및 라우트 추적 메시징은 메시지가 큐 관리자 네트워크를 따라 라우트될 때 메시지에 대한 활동 정보를 기록할 수 있게 하는 기술입니다.

#### [58 페이지의 『라우트 추적 메시징』](#)

라우트 추적 메시징은 라우트 추적 메시지를 사용하여 메시지에 대한 활동 정보를 기록하는 기술입니다. 라우트 추적 메시징은 큐 관리자 네트워크로 라우트 추적 메시지 송신을 포함합니다.

### 관련 태스크

[사용자 소유의 메시지 채널 에이전트 작성](#)

## 메시지 라우트 기술

활동 기록 및 라우트 추적 메시징은 메시지가 큐 관리자 네트워크를 따라 라우트될 때 메시지에 대한 활동 정보를 기록할 수 있게 하는 기술입니다.

### 활동 기록

메시지가 적절한 보고서 옵션이 지정된 경우, 메시지가 큐 관리자 네트워크를 통해 라우트될 때 애플리케이션이 활동 보고서를 생성할 것을 요청합니다. 애플리케이션이 메시지를 위해 활동을 수행할 때, 활동 보고서가 생성되고 적절한 위치로 전달될 수 있습니다. 활동 보고서에는 메시지에 수행된 활동에 관한 정보가 들어 있습니다.

활동 보고서를 사용하여 수집된 활동 정보는 메시지 라우트가 판별될 수 있기 전에 순서대로 정렬되어야 합니다.

### 라우트 추적 메시징

라우트 추적 메시징은 큐 관리자 네트워크로 라우트 추적 송신을 포함하는 기술입니다. 애플리케이션이 라우트 추적 메시지를 위해 활동을 수행할 때, 활동 정보가 라우트 추적 메시지의 메시지 데이터에 누적될 수 있거나 활동 보고서가 생성될 수 있습니다. 활동 정보가 라우트 추적 메시지의 메시지 데이터에 누적되는 경우, 대상 큐에 도달할 때 라우트 추적 메시지의 모든 정보를 포함하는 라우트 추적 응답 메시지가 생성되어 적절한 위치로 전달될 수 있습니다.

라우트 추적 메시지가 그 대신에 수행되는 활동의 순서를 기록하는 데 전용되기 때문에 활동 보고서를 요청하는 정상 메시지와 비교하여 사용 가능한 추가 처리 옵션이 있습니다.

### 활동 기록과 라우트 추적 메시징의 비교

활동 기록과 라우트 추적 메시징은 둘 다 큐 관리자 네트워크를 통해 메시지가 사용한 라우트를 판별하기 위한 활동 정보를 제공할 수 있습니다. 두 메소드는 고유한 장점을 갖고 있습니다.

이점	활동 기록	라우트 추적 메시징
메시지의 마지막 알려진 위치를 판별할 수 있습니까?	예	예
큐 관리자 네트워크의 구성 문제를 판별할 수 있습니까?	예	예
임의의 메시지가 요청할 수 있습니까? (라우트 추적 메시징과의 사용으로 제한되지 않음)	예	아니오
메시지 데이터가 수정되지 않은 채로 있음	예	아니오
메시지가 정상적으로 처리됨	예	아니오
활동 정보가 메시지 데이터에 누적될 수 있음	아니오	예
대상 큐로 선택적인 메시지 전달	아니오	예
메시지가 무한 루프에 갇힌 경우 감지하고 다룰 수 있음	아니오	예
활동 정보를 신뢰할 수 있는 순서로 넣을 수 있음	아니오	예
활동 정보를 표시하기 위해 애플리케이션이 제공됨	아니오	예

### 메시지 라우트 완전성

일부 경우에는 메시지를 위해 수행되는 활동의 전체 순서를 식별하는 것이 불가능하므로, 부분 메시지 라우트만 판별할 수 있습니다. 메시지 라우트의 완전성은 메시지가 라우트되는 큐 관리자 네트워크에 직접적으로 영향을 받습니다. 메시지 라우트의 완전성은 다음과 같이 큐 관리자 네트워크에서 큐 관리자의 레벨에 따라 다릅니다.

#### WebSphere MQ 버전 6.0 및 후속 릴리스에서의 큐 관리자

WebSphere MQ 버전 6.0 또는 후속 릴리스에서 큐 관리자에 연결된 MCA 및 사용자가 작성한 애플리케이션은 메시지를 대신하여 수행되는 활동에 관련된 정보를 기록할 수 있습니다. 활동 정보의 기록은 큐 관리자 속성 ACTIVREC 및 ROUTEREC에 의해 제어됩니다. 큐 관리자 네트워크가 WebSphere MQ 버전 6.0 또는 후속 릴리스의 큐 관리자만으로 구성되는 경우, 완전한 메시지 라우트를 판별할 수 있습니다.

#### 버전 6.0 미만의 WebSphere MQ 큐 관리자

버전 6.0 미만의 WebSphere MQ 큐 관리자에 연결된 애플리케이션은 메시지를 대신하여 수행된 활동을 기록하지 않습니다. 큐 관리자 네트워크에 버전 6.0 미만의 WebSphere MQ 큐 관리자가 포함된 경우, 부분적인 메시지 라우트만 판별될 수 있습니다.

### 활동 정보가 저장되는 방법

WebSphere MQ는 활동 정보를 활동 보고서, 라우트 추적 메시지 또는 라우트 추적 응답 메시지에 저장합니다. 각 경우에 정보는 Activity PCF 그룹이라는 구조에 저장됩니다. 라우트 추적 메시지 또는 라우트 추적 응답 메시지는 메시지에서 수행되는 활동 수에 따라서 많은 Activity PCF 그룹을 포함할 수 있습니다. 활동 보고서는 모든 기록된 활동에 대해 별도의 활동 보고서가 생성되기 때문에 하나의 Activity PCF 그룹을 포함합니다.

라우트 추적 메시징을 사용하면 추가 정보를 기록할 수 있습니다. 이 추가 정보는 TraceRoute PCF 그룹이라는 구조에 저장됩니다. TraceRoute PCF 그룹은 추가 활동 정보를 저장하고 큐 관리자 네트워크를 통해 라우트될 때 라우트 추적 메시지가 처리되는 방법을 판별하는 옵션을 지정하는 데 사용되는 많은 PCF 구조를 포함합니다.

#### 관련 개념

54 페이지의 『활동 기록』

활동 기록은 큐 관리자 네트워크를 통해 메시지가 전달되는 라우트를 판별하는 기술입니다. 메시지가 취하는 라우트를 판별하려면 메시지를 위해 수행되는 활동이 기록됩니다.

#### 58 페이지의 『라우트 추적 메시징』

라우트 추적 메시징은 라우트 추적 메시지를 사용하여 메시지에 대한 활동 정보를 기록하는 기술입니다. 라우트 추적 메시징은 큐 관리자 네트워크로 라우트 추적 메시지 송신을 포함합니다.

#### 관련 참조

#### 63 페이지의 『TraceRoute PCF 그룹』

TraceRoute PCF 제어 그룹의 속성이 라우트 추적 메시지의 작동을 제어합니다. TraceRoute PCF 그룹은 모든 라우트 추적 메시지의 메시지 데이터에 있습니다.

#### 94 페이지의 『활동 보고서 메시지 데이터』

이 페이지를 사용하여 활동 보고서 메시지의 Activity PCF 그룹에 포함된 매개변수를 봅니다. 일부 매개변수는 특정 조작이 수행되었을 때에만 리턴됩니다.

## 활동 기록

활동 기록은 큐 관리자 네트워크를 통해 메시지가 전달되는 라우트를 판별하는 기술입니다. 메시지가 취하는 라우트를 판별하려면 메시지를 위해 수행되는 활동이 기록됩니다.

활동 레코드를 사용할 때, 메시지를 위해 수행된 각 활동을 활동 보고서에 기록할 수 있습니다. 활동 보고서는 보고 메시지의 유형입니다. 각 활동 보고서에는 메시지를 위해 활동을 수행한 애플리케이션에 대한 정보, 활동 발생 시기 및 활동의 일부로 수행된 조작에 관한 정보가 포함됩니다. 일반적으로 활동 보고서는 함께 수집된 응답 대상 큐에 전달됩니다. 메시지와 관련된 활동 보고서를 연구해서 큐 관리자 네트워크를 통해 메시지가 취한 라우트를 판별할 수 있습니다.

## 활동 보고서 사용법

메시지가 큐 관리자 네트워크를 통해 라우트되면 활동 보고서가 생성될 수 있습니다. 다음 방법으로 활동 보고서 정보를 사용할 수 있습니다.

### 마지막으로 알려진 메시지 위치 판별

활동 기록을 위해 사용할 수 있는 메시지가 의도된 목적지에 도달하지 않은 경우, 큐 관리자 네트워크를 통해 라우트되는 메시지에 대해 생성되는 활동 보고서를 연구하여 마지막으로 알려진 메시지 위치를 판별할 수 있습니다.

### 큐 관리자 네트워크의 구성 문제 판별

활동 기록으로 사용할 수 있는 여러 메시지를 큐 관리자 네트워크로 송신할 수 있습니다. 각 메시지와 관련된 활동 보고서를 연구하면 예상된 라우트를 취하지 않았음이 분명해질 수 있습니다. 이런 일이 발생할 수 있는 여러 이유가 있습니다. 예를 들어 강제로 메시지가 다른 라우트를 취하도록 하면서 채널이 중지될 수 있습니다. 이런 상황에서는 시스템 관리자가 큐 관리자 네트워크에 문제가 있는지 여부를 판별하고, 있는 경우 정정할 수 있습니다.

**참고:** WebSphere MQ 표시 라우트 애플리케이션을 사용하여 라우트 추적 메시지와 결합하여 활동 레코드를 사용할 수 있습니다.

## 활동 보고서 형식

활동 보고서는 메시지를 위해 활동을 수행하는 애플리케이션에서 생성된 PCF 메시지입니다. 활동 보고서는 다음과 같이 메시지 디스크립터 및 메시지 데이터를 포함하는 표준 WebSphere MQ 보고서 메시지입니다.

### 메시지 디스크립터

- MQMD 구조

### 메시지 데이터

- 임베드된 PCF 헤더(MQEPH)
- 활동 보고서 메시지 데이터

활동 보고서 메시지 데이터는 활동 PCF 그룹으로 구성되며, 라우트 추적 메시징으로 생성된 경우 TraceRoute PCF 그룹으로 구성됩니다.

## 관련 참조

[MQMD - 메시지 디스크립터](#)

[MQEPH - 임베드된 PCF 헤더](#)

## 활동 기록 제어

큐 관리자 레벨에서 활동 기록을 사용 가능하게 하십시오. 전체 큐 관리자 네트워크를 사용 가능하게 설정하려면 활동 기록을 위해 네트워크의 모든 큐 관리자를 개별적으로 사용 가능하게 설정하십시오. 추가 큐 관리자를 사용 가능하게 하면 추가 활동 보고서가 생성됩니다.

## 이 태스크 정보

큐 관리자를 통해 라우트될 때 메시지에 대한 활동 보고서를 생성하려면 활동 보고서를 요청하는 메시지를 정의하고, 활동 기록을 위해 큐 관리자를 사용 가능하게 하고, 메시지에 대해 활동을 수행하는 애플리케이션이 활동 보고서를 생성할 수 있는지 확인하십시오.

메시지가 큐 관리자를 통해 라우트될 때 메시지에 대한 활동 보고서가 생성되기를 원하지 않는 경우, 활동 기록에 대해 큐 관리자를 사용 불가능하게 하십시오.

## 프로시저

1. 메시지에 대한 활동 보고서를 요청하십시오.

a) 메시지의 메시지 디스크립터에서, *Report* 필드에 `MQRO_ACTIVITY`를 지정하십시오.

b) 메시지의 메시지 디스크립터에서, *ReplyToQ* 필드에 응답 대상 큐의 이름을 지정하십시오.

**경고:** 활동 기록을 위해 큐 관리자 네트워크의 모든 메시지를 사용 가능하게 하는 것은 피하십시오. 활동 기록을 위해 사용 가능한 메시지는 그 대신 생성되는 많은 활동 보고서를 가질 수 있습니다. 큐 관리자 네트워크의 모든 메시지가 활동 기록에 대해 사용 가능한 경우, 큐 관리자 네트워크 트래픽이 허용할 수 없는 레벨로 증가할 수 있습니다.

2. 활동 기록에 대해 큐 관리자를 사용 가능 또는 사용 불가능하게 하십시오.

매개변수 `ACTIVREC`를 지정한 `MQSC` 명령 `ALTER QMGR`을 사용하여 큐 관리자 속성의 값을 변경하십시오. 가능한 값은 다음과 같습니다.

### MSG

큐 관리자가 활동 기록에 대해 사용 가능합니다. 생성되는 모든 활동 보고서가 메시지의 메시지 디스크립터에 지정된 응답 대상 큐로 전달됩니다. 이 값은 기본값입니다.

### 큐

큐 관리자가 활동 기록에 대해 사용 가능합니다. 생성되는 모든 활동 보고서가 로컬 시스템 큐 `SYSTEM.ADMIN.ACTIVITY.QUEUE`로 전달됩니다. 시스템 큐는 또한 활동 보고서를 공용 큐로 전달하는데 사용될 수 있습니다.

### DISABLED

큐 관리자가 활동 기록에 대해 사용 불가능합니다. 이 큐 관리자의 범위에 있는 중에는 활동 보고서가 생성되지 않습니다.

예를 들어, 활동 기록에 대해 큐 관리자를 사용 가능하게 하고 생성되는 모든 활동 보고서가 로컬 시스템 큐 `SYSTEM.ADMIN.ACTIVITY.QUEUE`로 전달되도록 지정하려면 다음 `MQSC` 명령을 사용하십시오.

```
ALTER QMGR ACTIVREC(Queue)
```

**알아두기:** `ACTIVREC` 큐 관리자 속성을 수정할 때 채널이 재시작될 때까지 실행 중인 MCA가 변경을 감지하지 않습니다.

3. 애플리케이션이 MCA가 메시지의 활동 보고서를 생성하는지 여부를 판별하기 위해 사용하는 것과 동일한 알고리즘을 사용하는지 확인하십시오.
  - a) 메시지가 활동 보고서 생성되도록 요청했는지 확인하십시오.
  - b) 메시지가 현재 상주하는 큐 관리자가 활동 기록에 대해 사용 가능한지 확인하십시오.
  - c) `ACTIVREC` 큐 관리자 속성에 의해 판별되는 큐에 활동 보고서를 넣으십시오.

## 활동 보고서를 위한 공용 큐 설정

보고서가 로컬 시스템 큐로 전달될 때 특정 메시지와 관련된 활동 보고서의 위치를 판별하려면 단일 노드에서 공용 큐를 사용하는 것이 더 효율적입니다.

### 시작하기 전에

활동 기록을 위해 큐 관리자를 사용 가능하게 하고 생성되는 모든 활동 보고서가 로컬 시스템 큐 SYSTEM.ADMIN.ACTIVITY.QUEUE로 전달되도록 지정하려면 ACTIVREC 매개변수를 설정하십시오.

### 이 태스크 정보

큐 관리자 네트워크의 많은 큐 관리자가 로컬 시스템 큐로 활동 보고서를 전달하도록 설정되는 경우, 특정 메시지와 관련된 활동 보고서의 위치를 판별하는 것은 시간 소모적일 수 있습니다. 그렇지 않으면 공용 큐를 호스트하는 큐 관리자인 단일 노드를 사용하십시오. 큐 관리자 네트워크의 모든 큐 관리자가 이 공용 큐로 활동 보고서를 전달할 수 있습니다. 공용 큐 사용의 이점은 큐 관리자가 메시지에 지정된 응답 대상 큐로 활동 보고서를 전달할 필요가 없고 메시지와 관련된 활동 보고서의 위치를 판별할 때 하나의 큐만 조회한다는 점입니다.

공용 큐를 설정하려면 다음 단계를 수행하십시오.

### 프로시저

1. 큐 관리자를 단일 노드로 선택 또는 정의하십시오.
2. 단일 노드에서 공용 큐에 사용할 큐를 선택 또는 정의하십시오.
3. 활동 보고서가 공용 큐로 전달될 모든 큐 관리자에서 로컬 시스템 큐 SYSTEM.ADMIN.ACTIVITY.QUEUE를 리모트 큐 정의로 다시 정의하십시오.
  - a) 단일 노드의 이름을 리모트 큐 관리자 이름으로 지정하십시오.
  - b) 공용 큐의 이름을 리모트 큐 이름으로 지정하십시오.

### 메시지 라우트 정보 판별

메시지 라우트를 판별하려면 수집된 활동 보고서로부터 정보를 얻으십시오. 필수 정보를 판별하고 활동 보고서를 순서대로 정렬할 수 있기에 충분한 활동 보고서가 응답 대상 큐에 있는지 판별하십시오.

### 이 태스크 정보

활동 보고서가 응답 대상 큐에 놓여지는 순서가 반드시 활동이 수행된 순서와 상관되지는 않습니다. 라우트 추적 메시지에 대해 활동 보고서가 생성되지 않은 경우 활동 보고서의 순서를 수동으로 지정해야 합니다. 이 경우, WebSphere MQ 라우트 표시 애플리케이션을 사용하여 활동 보고서의 순서를 지정할 수 있습니다.

필요한 정보를 얻기 위해 응답 대상 큐에 충분한 활동 보고서가 있는지 판별하십시오.

### 프로시저

1. 활동 보고서의 ID와 원래 메시지를 비교하여 응답 대상 큐의 모든 관련 활동 보고서를 식별하십시오. 활동 보고서가 원래 메시지와 상관될 수 있도록 원래 메시지의 보고서 옵션을 설정해야 합니다.
2. 응답 대상 큐의 식별된 활동 보고서를 정렬하십시오.  
활동 보고서의 다음 매개변수를 사용할 수 있습니다.

#### **OperationType**

수행되는 조작의 유형으로 현재 활동 보고서 직전 또는 직후에 생성된 활동 보고서를 판별할 수도 있습니다.

예를 들어, 활동 보고서에서 MCA가 전송 큐에서 채널을 따라 메시지를 송신했음을 상세히 설명합니다. The last operation detailed in the activity report has an *OperationType* of send and details that the message was sent using the channel, CH1, to the destination queue manager, QM1. 이것은 메시지에 수행된 다음 활동이 큐 관리자 QM1에서 발행했으며 채널 CH1의 receive 조작으로 시작했음을 의미합니다. 이 정보를 사용하여 다음 활동 보고서가 존재하고 획득되었다고 가정할 때 다음 활동 보고서를 식별할 수 있습니다.

## OperationDate 및 OperationTime

각 활동 보고서에서 조작의 날짜 및 시간으로부터 활동의 일반적인 순서를 판별할 수 있습니다.

**경고:** 큐 관리자 네트워크의 모든 큐 관리자가 시스템 시계를 동기화하지 않은 경우, 날짜 및 시간순으로 정렬하는 것이 활동 보고서가 올바른 순서라고 보증하지 않습니다. 순서를 수동으로 설정해야 합니다.

활동 보고서의 순서는 메시지가 큐 관리자 네트워크를 통해 잡은 라우트 또는 부분 라우트를 나타냅니다.

3. 정렬된 활동 보고서의 활동 정보로부터 필요한 정보를 얻으십시오.

메시지에 관한 정보를 충분하지 않은 경우, 추가 활동 보고서를 획득할 수도 있습니다.

## 추가 활동 보고서 검색

메시지 라우트를 판별하기 위해서는 수집된 활동 보고서로부터 충분한 정보가 사용 가능해야 합니다. 메시지가 지정한 응답 대상 큐로부터 메시지 관련 활동 보고서를 검색하지만 필요한 정보가 없는 경우 추가 활동 보고서를 찾으십시오.

## 이 태스크 정보

추가 활동 보고서의 위치를 판별하려면 다음 단계를 수행하십시오.

## 프로시저

1. 활동 보고서를 공용 큐로 전달하는 큐 관리자 네트워크의 모든 큐 관리자의 경우, 원래 메시지의 *MsgId*와 일치하는 *CorrelId*를 갖는 공용 큐에서 활동 보고서를 검색하십시오.
2. 공용 큐로 활동 보고서를 전달하지 않는 큐 관리자 네트워크의 모든 큐 관리자의 경우, 다음과 같이 활동 보고서를 검색하십시오.
  - a) 기존 활동 보고서를 조사하여 메시지가 라우트된 큐 관리자를 식별하십시오.
  - b) 이들 큐 관리자에 대해 활동 기록에 대해 사용 가능한 큐 관리자를 식별하십시오.
  - c) 이들 큐 관리자에 대해, 지정된 응답 대상 큐로 활동 보고서를 리턴하지 않은 모든 큐 관리자를 식별하십시오.
  - d) 식별한 각 큐 관리자에 대해, 시스템 큐 SYSTEM.ADMIN.ACTIVITY.QUEUE를 검사하고 원래 메시지의 *MsgId*와 일치하는 *CorrelId*를 갖는 모든 활동 보고서를 검색하십시오.
  - e) 시스템 큐에서 활동 보고서를 찾지 못하는 경우, 큐 관리자 데드-레터 큐(있는 경우)를 확인하십시오. 활동 보고서는 보고서 옵션 MQRO\_DEAD\_LETTER\_Q가 설정되는 경우 데드-레터 큐로만 전달될 수 있습니다.
3. 획득한 모든 활동 보고서를 순서대로 배열하십시오.  
그러면 활동 보고서의 순서는 메시지가 잡은 라우트 또는 부분 라우트를 나타냅니다.
4. 정렬된 활동 보고서의 활동 정보로부터 필요한 정보를 얻으십시오.  
일부 상황에서는 기록된 활동 정보가 지정된 응답 대상 큐, 공용 큐 또는 시스템 큐에 접근할 수 없습니다.

## 활동 정보가 획득되지 않는 상황

메시지를 위해 수행되는 활동의 전체 순서를 판별하려면 모든 활동과 관련된 정보가 획득되어야 합니다. 임의의 활동에 관련된 정보가 기록되지 않았거나 획득되지 않은 경우 활동의 부분적인 순서만 판별할 수 있습니다.

다음 상황에서 활동 정보가 기록되지 않습니다.

- 메시지가 버전 6.0 미만의 WebSphere MQ 큐 관리자에 의해 처리됩니다.
- 메시지가 활동 기록에 대해 사용 가능하지 않은 큐 관리자에 의해 처리됩니다.
- 메시지를 처리할 것으로 예상한 애플리케이션이 실행 중이 아닙니다.

다음 상황에서는 기록된 활동 정보가 지정된 응답 대상 큐에 도달할 수 없습니다.

- 응답 대상 큐로 활동 보고서를 라우트하도록 정의된 채널이 없습니다.
- 활동 보고서를 응답 대상 큐로 라우트할 채널이 실행 중이 아닙니다.

- 활동 보고서를 다시 응답 대상 큐가 상주하는 큐 관리자(큐 관리자 알리어스)로 라우트할 리모트 큐 정의가 정의되지 않았습니다.
- 원래 메시지를 생성한 사용자가 큐 관리자 알리어스에 대한 열기 또는 Put 권한이 없습니다.
- 원래 메시지를 생성한 사용자가 응답 대상 큐에 대한 열기 또는 Put 권한이 없습니다.
- 응답 대상 큐가 put 금지되었습니다.

다음 상황에서는 기록된 활동 정보가 시스템 큐 또는 공용 큐에 도달할 수 없습니다.

- 공용 큐가 사용되며 활동 보고서를 공용 큐로 라우트하도록 정의된 채널이 없는 경우.
- 공용 큐가 사용되며 활동 보고서를 공용 큐로 라우트할 채널이 실행 중이 아닌 경우.
- 공용 큐가 사용되며 시스템 큐가 올바르게 않게 정의된 경우.
- 원래 메시지를 생성한 사용자가 시스템 큐에 대한 열기 또는 Put 권한이 없습니다.
- 시스템 큐가 put 금지되었습니다.
- 공용 큐가 사용되며 원래 메시지를 생성한 사용자가 공용 큐에 대한 열기 또는 Put 권한이 없는 경우.
- 공용 큐가 사용되며 공용 큐가 put 금지된 경우.

이런 상황에서는 활동 보고서가 보고서 옵션 MQRO\_DISCARD\_MSG가 지정되지 않은 경우 활동 보고서가 거부되는 큐 관리자에서 데드-레터 큐가 정의된 경우 활동 보고서를 데드-레터 큐에서 검색할 수 있습니다. 활동 보고서는 활동 보고서가 생성된 원래 메시지가 메시지 디스크립터의 보고서 필드에서 MQRO\_PASS\_DISCARD\_AND\_EXPIRY 및 MQRO\_DISCARD\_MSG가 둘 다 지정된 경우에만 이 보고서 옵션이 지정됩니다.

## 라우트 추적 메시지

라우트 추적 메시징은 라우트 추적 메시지를 사용하여 메시지에 대한 활동 정보를 기록하는 기술입니다. 라우트 추적 메시징은 큐 관리자 네트워크로 라우트 추적 메시지 송신을 포함합니다.

라우트 추적 메시징이 큐 관리자 네트워크를 통해 라우트될 때 활동 정보가 기록됩니다. 이 활동 정보에는 활동을 수행한 애플리케이션, 수행된 시기, 활동의 일부로 수행된 조작에 관한 정보가 포함됩니다. 라우트 추적 메시징을 사용하여 기록된 정보를 다음 목적으로 사용할 수 있습니다.

### 메시지의 마지막 알려진 위치 판별

메시지가 의도된 목적지에 도달하지 않는 경우, 라우트 추적 메시지에 대해 기록된 활동 정보를 사용하여 메시지의 마지막으로 알려진 위치를 판별할 수 있습니다. 라우트 추적 메시지는 동일한 라우트를 따르기 위해 원래 메시지와 동일한 대상 목적지를 갖는 큐 관리자 네트워크로 송신됩니다. 활동 정보는 라우트 추적 메시지의 메시지 데이터에 누적되거나 활동 보고서를 사용하여 기록될 수 있습니다. 라우트 추적 메시지가 원래 메시지와 동일한 라우트를 따를 확률을 늘리기 위해 원래 메시지를 흉내내도록 라우트 추적 메시지를 수정할 수 있습니다.

### 큐 관리자 네트워크의 구성 문제 판별

라우트 추적 메시지는 큐 관리자 네트워크로 송신되고 활동 정보가 기록됩니다. 라우트 추적 메시지에 대해 기록되는 활동 정보를 연구하여 라우트 추적 메시지가 예상 라우트를 따르지 않았음이 명백해질 수 있습니다. 이런 일이 발생할 수 있는 여러 이유가 있습니다. 예를 들어 채널이 비활성이어서 메시지가 다른 라우트를 취하도록 강제할 수 있습니다. 이런 상황에서는 시스템 관리자가 큐 관리자 네트워크에 문제가 있는지 여부를 판별하고, 있는 경우 정정할 수 있습니다.

WebSphere MQ 라우트 표시 애플리케이션을 사용하여 라우트 추적 메시지를 큐 관리자 네트워크에 대해 구성 및 생성하거나 Put 조작을 수행할 수 있습니다.

**경고:** 라우트 추적 메시지를 분배 목록에 넣는 경우 결과는 정의되지 않습니다.

### 관련 개념

[110 페이지의 『라우트 추적 메시지 참조』](#)

이 페이지를 사용하여 라우트 추적 메시지 형식을 개괄합니다. 라우트 추적 메시지 데이터는 라우트 추적 메시지가 유발한 활동을 설명하는 매개변수를 포함합니다.

## 활동 정보가 기록되는 방법

라우트 추적 메시징을 사용하면 라우트 추적 메시지의 메시지 데이터에 활동 정보를 기록하거나 활동 보고서를 사용할 수 있습니다. 그렇지 않으면 두 기술을 모두 사용할 수 있습니다.

### 라우트 추적 메시지의 메시지 데이터에 활동 정보 누적

라우트 추적 메시지가 큐 관리자 네트워크를 통해 라우트될 때, 라우트 추적 메시지를 위해 수행되는 활동에 관한 정보가 라우트 추적 메시지의 메시지 데이터에 누적될 수 있습니다. 활동 정보는 *Activity* PCF 그룹에 저장됩니다. 라우트 추적 메시지를 위해 수행되는 모든 활동의 경우 *Activity* PCF 그룹이 라우트 추적 메시지의 메시지 데이터에 있는 PCF 블록의 끝에 기록됩니다.

추가 활동 정보는 *TraceRoute* PCF 그룹이라는 PCF 그룹에서 라우트 추적 메시지에 기록됩니다. 추가 활동 정보는 이 PCF 그룹에 저장되며, 기록된 활동의 순서를 판별하는 데 사용될 수 있습니다. 이 기술은 *TraceRoute* PCF 그룹의 *Accumulate* 매개변수에 의해 제어됩니다.

### 활동 보고서를 사용한 활동 정보 기록

라우트 추적 메시지가 큐 관리자 네트워크를 통해 라우트될 때, 라우트 추적 메시지를 위해 수행되는 모든 활동에 대한 활동 보고서가 생성될 수 있습니다. 활동 정보는 *Activity* PCF 그룹에 저장됩니다. 라우트 추적 메시지를 위해 수행되는 모든 활동에 대해 *Activity* PCF 그룹을 포함하는 활동 보고서가 생성됩니다. 라우트 추적 메시지에 대한 활동 기록은 다른 메시지와 동일한 방식으로 작업합니다.

라우트 추적 메시지에 대해 생성되는 활동 보고서는 다른 메시지에 대해 생성되는 메시지에 비교하여 추가 활동 정보를 포함합니다. 추가 정보는 *TraceRoute* PCF 그룹에서 리턴됩니다. *TraceRoute* PCF 그룹에 들어있는 정보는 활동 보고서가 생성된 시간부터만 정확합니다. 추가 정보를 사용하여 라우트 추적 메시지를 위해 수행되는 활동의 순서를 판별할 수 있습니다.

### 기록된 활동 정보 획득

라우트 추적 메시지가 그의 의도된 목적지에 도달했거나 제거될 때, 활동 정보를 획득하기 위해 사용하는 방법은 해당 정보가 기록된 방법에 따라 다릅니다.

### 시작하기 전에

활동 정보에 익숙하지 않은 경우 59 페이지의 『활동 정보가 기록되는 방법』의 내용을 참조하십시오.

### 이 태스크 정보

라우트 추적 메시지가 그의 의도된 목적지에 도달했거나 제거된 후 활동 정보를 획득하려면 다음 방법을 사용하십시오.

### 프로시저

- 라우트 추적 메시지를 검색하십시오.  
*TraceRoute* PCF 그룹의 *Deliver* 매개변수가 라우트 추적 메시지가 도착 시 대상 큐에 배치되는지 아니면 제거되는지 여부를 제어합니다. 라우트 추적 메시지가 대상 큐로 전달되는 경우 이 큐에서 라우트 추적 메시지를 검색할 수 있습니다. 그런 다음 *WebSphere MQ* 표시 라우트 애플리케이션을 사용하여 활동 정보를 표시할 수 있습니다.  
활동 정보가 라우트 추적 메시지의 메시지 데이터에 누적되도록 요청하려면 *TraceRoute* PCF 그룹의 *Accumulate* 매개변수를 *MQROUTE\_ACCUMULATE\_IN\_MSG*로 설정하십시오.
- 라우트 추적 응답 메시지를 사용하십시오.  
라우트 추적 메시지가 의도된 목적지에 도달하거나 라우트 추적 메시지가 큐 관리자 네트워크에서 더 이상 라우트될 수 없을 때, 라우트 추적 응답 메시지가 생성될 수 있습니다. 라우트 추적 응답 메시지에는 라우트 추적 메시지의 모든 활동 정보의 복제가 들어 있으며, 지정된 응답 대상 큐 또는 시스템 큐 *SYSTEM.ADMIN.TRACE.ROUTE.QUEUE*로 전달됩니다. *WebSphere MQ* 표시 라우트 애플리케이션을 사용하여 활동 정보를 표시할 수 있습니다.  
라우트 추적 응답 메시지를 요청하려면 *TraceRoute* PCF 그룹의 *Accumulate* 매개변수를 *MQROUTE\_ACCUMULATE\_AND\_REPLY*로 설정하십시오.

- 활동 보고서를 사용하십시오.  
라우트 추적 메시지에 대한 활동 보고서가 생성되는 경우, 활동 정보를 획득할 수 있기 전에 활동 보고서를 찾아야 합니다. 그런 다음 활동 순서를 판별하기 위해 활동 보고서를 정렬해야 합니다.

## 라우트 추적 메시징 제어

해당 큐 관리자의 범위에 있는 애플리케이션이 라우트 추적 메시지에 활동 정보를 쓸 수 있도록 큐 관리자 레벨에서 라우트 추적 메시징을 사용 가능하게 하십시오. 전체 큐 관리자 네트워크를 사용 가능하게 설정하려면 라우트 추적 메시징을 위해 네트워크의 모든 큐 관리자를 개별적으로 사용 가능하게 설정하십시오. 추가 큐 관리자를 사용 가능하게 하면 추가 활동 보고서가 생성됩니다.

### 시작하기 전에

활동 보고서를 사용하여 라우트 추적 메시지에 대한 활동 정보를 기록 중인 경우 [55 페이지의 『활동 기록 제어』](#)의 내용을 참조하십시오.

### 이 태스크 정보

큐 관리자를 통해 라우트될 때 라우트 추적 메시지에 대한 활동 정보를 기록하려면 다음 단계를 수행하십시오.

### 프로시저

- 활동 정보가 라우트 추적 메시지에 대해 기록될 방법을 정의하십시오.  
[62 페이지의 『라우트 추적 메시지 생성 및 구성』](#)의 내용을 참조하십시오.
- 라우트 추적 메시지에 활동 정보를 누적하려는 경우 큐 관리자가 라우트 추적 메시지에 대해 사용 가능한지 확인하십시오.
- 라우트 추적 메시지에 활동 정보를 누적하려는 경우, 라우트 추적 메시지에 활동을 수행하는 애플리케이션이 라우트 추적 메시지의 메시지 데이터에 활동 정보를 쓸 수 있는지 확인하십시오.

### 관련 개념

[62 페이지의 『라우트 추적 메시지 생성 및 구성』](#)

라우트 추적 메시지는 특정 메시지 디스크립터와 메시지 데이터 부분으로 구성됩니다. 라우트 추적 메시지를 생성하려면 수동으로 메시지를 생성하거나 WebSphere MQ 라우트 표시 애플리케이션을 사용하십시오.

### 관련 태스크

[55 페이지의 『활동 기록 제어』](#)

큐 관리자 레벨에서 활동 기록을 사용 가능하게 하십시오. 전체 큐 관리자 네트워크를 사용 가능하게 설정하려면 활동 기록을 위해 네트워크의 모든 큐 관리자를 개별적으로 사용 가능하게 설정하십시오. 추가 큐 관리자를 사용 가능하게 하면 추가 활동 보고서가 생성됩니다.

## 라우트 추적 메시지에 대한 큐 관리자 사용 가능

큐 관리자가 라우트 추적 메시지에 대해 사용 가능 또는 사용 불가능한지 여부를 제어하려면 큐 관리자 속성 ROUTEREC를 사용하십시오.

큐 관리자 속성의 값을 변경하려면 ROUTEREC 매개변수를 지정한 MQSC 명령 ALTER QMGR을 사용하십시오. 가능한 값은 다음과 같습니다.

### MSG

큐 관리자가 라우트 추적 메시징에 사용 가능합니다. 큐 관리자의 범위 안에 있는 애플리케이션은 라우트 추적 메시지에 활동 정보를 쓸 수 있습니다.

TraceRoute PCF 그룹의 *Accumulate* 매개변수가 MQROUTE\_ACCUMULATE\_AND\_REPLY로 설정되고 라우트 추적 메시지에 수행될 다음 활동이

- 제거됨
- 로컬 큐(대상 큐 또는 데드-레터 큐)에 넣어짐
- 라우트 추적 메시지에 대해 수행되는 총 활동 수가 TraceRoute PCF 그룹의 *MaxActivities* 매개변수 값을 초과하게 됨

라우트 추적 응답 메시지가 생성되고, 라우트 추적 메시지의 메시지 디스크립터에서 지정되는 응답 대상으로 전달됩니다.

## 큐

큐 관리자가 라우트 추적 메시징에 사용 가능합니다. 큐 관리자의 범위 안에 있는 애플리케이션은 라우트 추적 메시징에 활동 정보를 쓸 수 있습니다.

TraceRoute PCF 그룹의 *Accumulate* 매개변수가 MQROUTE\_ACCUMULATE\_AND\_REPLY로 설정되고 라우트 추적 메시징에 수행될 다음 활동이

- 제거됨
- 로컬 큐(대상 큐 또는 데드-레터 큐)에 넣어짐
- 라우트 추적 메시징에 대해 수행되는 총 활동 수가 TraceRoute PCF 그룹의 *MaxActivities* 매개변수 값을 초과하게 됨

라우트 추적 응답 메시지가 생성되고, 로컬 시스템 큐 SYSTEM.ADMIN.TRACE.ROUTE.QUEUE로 전달됩니다.

## DISABLED

큐 관리자가 라우트 추적 메시징에 사용 불가능합니다. 활동 정보가 라우트 추적 메시징에 누적되지 않지만, TraceRoute PCF 그룹이 이 큐 관리자의 범위에 있는 동안 업데이트될 수 있습니다.

예를 들어, 라우트 추적 메시징에 대해 큐 관리자를 사용 불가능하게 설정하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR ROUTEREC(DISABLED)
```

**알아두기:** ROUTEREC 큐 관리자 속성을 수정할 때, 실행 중인 MCA는 채널이 재시작될 때까지 변경을 감지하지 못합니다.

## 라우트 추적 메시징에 대해 애플리케이션 사용 가능

사용자 애플리케이션에 대해 라우트 추적 메시징을 사용 가능하게 설정하려면 MCA(message channel agent)가 사용하는 알고리즘을 사용자 알고리즘의 기반으로 하십시오.

## 시작하기 전에

라우트 추적 메시징의 형식에 익숙하지 않은 경우 [110 페이지의 『라우트 추적 메시지 참조』](#)의 내용을 참조하십시오.

## 이 태스크 정보

MCA(Message channel agent)는 라우트 추적 메시징에 대해 사용 가능합니다. 사용자 애플리케이션을 라우트 추적 메시징에 대해 사용 가능하게 설정하려면 MCA가 사용하는 알고리즘에서 다음 단계를 사용하십시오.

## 프로시저

1. 처리되는 메시지가 라우트 추적 메시지인지 여부를 판별하십시오.  
메시지가 라우트 추적 메시징의 형식을 준수하지 않는 경우, 메시지는 라우트 추적 메시징으로서 처리되지 않습니다.
2. 활동 정보가 기록되는지 여부를 판별하십시오.  
수행되는 활동의 세부사항 레벨이 *Detail* 매개변수에 의해 지정되는 세부사항 레벨보다 낮지 않은 경우, 활동 정보는 특정 상황에서 기록됩니다. 이 정보는 라우트 추적 메시징이 누적을 요청하고 큐 관리자가 라우트 추적 메시징에 대해 사용 가능한 경우 또는 라우트 추적 메시징이 활동 보고서를 요청하고 큐 관리자가 활동 기록에 대해 사용 가능한 경우에만 기록됩니다.
  - 활동 정보가 기록될 경우 *RecordedActivities* 매개변수를 증분하십시오.
  - 활동 정보가 기록되지 않을 경우 *UnrecordedActivities* 매개변수를 증분하십시오.
3. 라우트 추적 메시징에 수행되는 활동의 총 수가 *MaxActivities* 매개변수의 값을 초과하는지 판별하십시오.  
활동의 총 수는 *RecordedActivities*, *UnrecordedActivities* 및 *DiscontinuityCount*의 합입니다.

활동의 총 수가 *MaxActivities*를 초과하는 경우 메시지를 MQFB\_MAX\_ACTIVITIES 피드백과 함께 거부하십시오.

4. *Accumulate*의 값이 MQROUTE\_ACCUMULATE\_IN\_MSG 또는 MQROUTE\_ACCUMULATE\_AND\_REPLY로 설정되고 큐 관리자가 라우트 추적 메시징에 대해 사용 가능한 경우, 라우트 추적 메시지의 메시지 데이터에 있는 PCF 블록의 끝에 Activity PCF 그룹을 쓰십시오.
5. 라우트 추적 메시지를 로컬 큐로 전달하십시오.
  - *Deliver* 매개변수가 MQROUTE\_DELIVER\_NO로 지정되는 경우 라우트 추적 메시지를 MQFB\_NOT\_DELIVERED 피드백과 함께 거부하십시오.
  - *Deliver* 매개변수가 MQROUTE\_DELIVER\_YES로 지정되는 경우 라우트 추적 메시지를 로컬 큐로 전달하십시오.
6. 다음 조건이 모두 참인 경우 라우트 추적 응답 메시지를 생성하십시오.
  - 라우트 추적 메시지가 로컬 큐로 전달되거나 거부되었습니다.
  - *Accumulate* 매개변수의 값이 MQROUTE\_ACCUMULATE\_AND\_REPLY입니다.
  - 큐 관리자가 라우트 추적 메시징에 대해 사용 가능합니다.라우트 추적 응답 메시지는 ROUTEREC 큐 관리자 속성에 의해 판별되는 큐에 넣어집니다.
7. 라우트 추적 메시지가 활동 보고서를 요청했고 큐 관리자가 활동 기록에 대해 사용 가능한 경우 활동 보고서를 생성하십시오.  
활동 보고서는 ACTIVREC 큐 관리자 속성에 의해 판별되는 큐에 넣어집니다.

## 라우트 추적 메시지 생성 및 구성

라우트 추적 메시지는 특정 메시지 디스크립터와 메시지 데이터 부분으로 구성됩니다. 라우트 추적 메시지를 생성하려면 수동으로 메시지를 생성하거나 WebSphere MQ 라우트 표시 애플리케이션을 사용하십시오.

라우트 추적 메시지는 다음 부분으로 구성됩니다.

### 메시지 디스크립터

MQFMT\_ADMIN 또는 MQFMT\_EMBEDDED\_PCF로 설정된 *Format* 필드를 갖는 MQMD 구조입니다.

### 메시지 데이터

다음 조합 중 하나입니다.

- *Format*이 MQFMT\_ADMIN으로 설정된 경우 PCF 헤더(MQCFH)와 라우트 추적 메시지
- *Format*이 MQFMT\_EMBEDDED\_PCF로 설정된 경우 임베드된 PCF 헤더(MQEPH), 라우트 추적 메시지 데이터 및 추가 사용자 지정 메시지 데이터

라우트 추적 메시지 데이터는 *TraceRoute* PCF 그룹과 하나 이상의 *Activity* PCF 그룹으로 구성됩니다.

## 수동 생성

라우트 추적 메시지를 수동으로 생성할 때 *Activity* PCF 그룹은 필수가 아닙니다. *Activity* PCF 그룹은 MCA 또는 사용자 작성 애플리케이션이 그 대신에 활동을 수행할 때 라우트 추적 메시지의 메시지 데이터에 기록됩니다.

## WebSphere MQ 표시 라우트 애플리케이션

라우트 추적 메시지를 큐 관리자 네트워크에 대해 구성 및 생성하거나 Put 조작을 수행하려면 WebSphere MQ 라우트 표시 애플리케이션인 dspmqrte를 사용하십시오. 메시지 디스크립터의 *Format* 매개변수를 MQFMT\_ADMIN으로 설정하십시오. WebSphere MQ 라우트 표시 애플리케이션에 의해 생성된 라우트 추적 메시지에 사용자 데이터를 추가할 수 없습니다.

**제한사항:** dspmqrte cannot be issued on queue managers before WebSphere MQ Version 6.0 or on WebSphere MQ for z/OS queue managers. 첫 번째 큐 관리자가 라우트 추적 메시지가 이 유형의 큐 관리자로 라우트되도록 하려면 선택적 매개변수 -c를 사용하여 큐 관리자에 WebSphere MQ 버전 6.0 이상의 클라이언트로 연결하십시오.

## 원래 메시지 흉내내기

라우트 추적 메시지를 사용하여 다른 메시지가 큐 관리자 네트워크를 통해 취한 라우트를 판별할 때, 라우트 추적 메시지가 원래 메시지를 더 가깝게 흉내낼수록 라우트 추적 메시지가 원래 메시지와 동일한 라우트를 따를 기회가 더 큽니다.

다음 메시지 특성이 큐 관리자 네트워크 내에서 메시지가 전달되는 장소에 영향을 미칠 수 있습니다.

### Priority

우선순위는 메시지의 메시지 디스크립터에서 지정할 수 있습니다.

### 지속

지속은 메시지의 메시지 디스크립터에서 지정할 수 있습니다.

### 완료

만기는 메시지의 메시지 디스크립터에서 지정할 수 있습니다.

### 보고서 옵션

보고서 옵션은 메시지의 메시지 디스크립터에서 지정할 수 있습니다.

### 메시지 크기

메시지의 크기를 흉내내기 위해 추가 데이터를 메시지의 메시지 데이터에 기록할 수 있습니다. 이 목적의 경우 추가 메시지 데이터는 의미가 없을 수 있습니다.

**팁:** WebSphere MQ 표시 라우트 애플리케이션은 메시지 크기를 지정할 수 없습니다.

### 메시지 데이터

일부 큐 관리자 네트워크는 콘텐츠 기반 라우팅을 사용하여 메시지가 전달되는 장소를 판별합니다. 이 경우 라우트 추적 메시지의 메시지 데이터가 원래 메시지의 메시지 데이터를 흉내내도록 작성되어야 합니다.

**팁:** WebSphere MQ 표시 라우트 애플리케이션은 메시지 데이터를 지정할 수 없습니다.

## TraceRoute PCF 그룹

TraceRoute PCF 제어 그룹의 속성이 라우트 추적 메시지의 작동을 제어합니다. TraceRoute PCF 그룹은 모든 라우트 추적 메시지의 메시지 데이터에 있습니다.

다음 표는 MCA가 인식하는 TraceRoute 그룹의 매개변수를 나열합니다. 67 페이지의 『[추가 활동 정보](#)』에 설명된 대로 사용자 작성 애플리케이션이 인식하도록 작성된 경우 매개변수를 더 추가할 수 있습니다.

매개변수	유형
TraceRoute	MQCFGR
Detail	MQCFIN
RecordedActivities	MQCFIN
UnrecordedActivities	MQCFIN
DiscontinuityCount	MQCFIN
MaxActivities	MQCFIN
누적	MQCFIN
Forward	MQCFIN
전달	MQCFIN

TraceRoute PCF 그룹에 있는 각 매개변수의 설명은 다음과 같습니다.

### Detail

기록되는 활동 정보의 세부사항 레벨을 지정합니다. 가능한 값은 다음과 같습니다.

#### **MQROUTE\_DETAIL\_LOW**

사용자 애플리케이션이 수행한 활동만 기록됩니다.

#### **MQROUTE\_DETAIL\_MEDIUM**

MQROUTE\_DETAIL\_LOW에서 지정된 활동이 기록됩니다. 또한 MCA가 수행한 활동이 기록됩니다.

#### **MQROUTE\_DETAIL\_HIGH**

MQROUTE\_DETAIL\_LOW 및 MQROUTE\_DETAIL\_MEDIUM에서 지정되는 활동이 기록됩니다. MCA는 이 세부사항 레벨에서 어떤 추가 활동 정보도 기록하지 않습니다. 이 옵션은 추가 활동 정보를 기록할 사

용자 애플리케이션에만 사용할 수 있습니다. 예를 들어, 사용자 애플리케이션이 특정 메시지 특성을 고려하여 메시지가 택하는 라우트를 판별하는 경우, 라우팅 논리에 관한 정보가 이 세부사항 레벨에 포함될 수 있습니다.

### **RecordedActivities**

라우트 추적 메시지를 위해 수행되는 기록된 활동의 수를 지정합니다. 활동은 그에 관한 정보가 라우트 추적 메시지에 작성된 경우 또는 활동 보고서가 생성된 경우에 기록된 것으로 간주됩니다. 모든 기록된 활동에 대해 *RecordedActivities*가 하나씩 증분합니다.

### **UnrecordedActivities**

라우트 추적 메시지를 위해 수행되는 기록되지 않은 활동의 수를 지정합니다. 활동은 라우트 추적 메시징에 대해 사용 가능한 애플리케이션이 누적하지 않거나 관련 활동 정보가 활동 보고서에 기록되지 않는 경우 기록되지 않는 것으로 간주됩니다.

라우트 추적 메시지를 위해 수행되는 활동은 다음 상황에서 기록되지 않습니다.

- 수행되는 활동의 세부사항 레벨이 *Detail* 매개변수에 의해 지정되는 세부사항 레벨보다 낮습니다.
- 라우트 추적 메시지가 활동 보고서를 요청하지만 누적은 요청하지 않으며, 큐 관리자가 활동 기록에 대해 사용 가능하지 않습니다.
- 라우트 추적 메시지가 누적을 요청하지만 활동 보고서는 요청하지 않으며, 큐 관리자가 라우트 추적 메시징에 대해 사용 가능하지 않습니다.
- 라우트 추적 메시지가 누적과 활동 보고서를 둘 다 요청하고, 큐 관리자가 활동 기록 및 라우트 추적 메시징에 대해 사용 가능하지 않습니다.
- 라우트 추적 메시지가 누적이나 활동 보고서를 요청하지 않습니다.

모든 기록되지 않는 활동에 대해 *UnrecordedActivities* 매개변수가 하나씩 증분합니다.

### **DiscontinuityCount**

라우트 추적 메시지가 라우트 추적 메시징에 대해 사용 가능하지 않은 애플리케이션으로 큐 관리자를 통해 라우트된 횟수를 지정합니다. 이 값은 큐 관리자에 의해 증분됩니다. 값이 0보다 큰 경우 부분 메시지 라우트만 판별할 수 있습니다.

### **MaxActivities**

라우트 추적 메시지를 위해 수행될 수 있는 활동의 최대 수를 지정합니다.

활동의 총 수는 *RecordedActivities*, *UnrecordedActivities* 및 *DiscontinuityCount*의 합입니다. 활동의 총 수는 *MaxActivities*의 값을 초과해서는 안됩니다.

*MaxActivities*의 값은 다음과 같을 수 있습니다.

#### **양수**

활동의 최대 수입니다.

활동의 최대 수가 초과되는 경우 라우트 추적 메시지는 MQFB\_MAX\_ACTIVITIES 피드백과 함께 거부됩니다. 이것은 라우트 추적 메시지가 무한 루프에 갇힌 경우 무한히 전송되지 않도록 막을 수 있습니다.

### **MQROUTE\_UNLIMITED\_ACTIVITIES**

무제한 숫자의 활동이 라우트 추적 메시지를 위해 수행될 수 있습니다.

### **Accumulate**

활동 정보를 누적하는 데 사용되는 방법을 지정합니다. 가능한 값은 다음과 같습니다.

#### **MQROUTE\_ACCUMULATE\_IN\_MSG**

큐 관리자가 라우트 추적 메시징에 대해 사용 가능한 경우, 활동 정보는 라우트 추적 메시지의 메시지 데이터에 누적됩니다.

이 값이 지정되는 경우 라우트 추적 메시지 데이터는 다음으로 구성됩니다.

- *TraceRoute* PCF 그룹.
- 0개 이상의 *Activity* PCF 그룹.

#### **MQROUTE\_ACCUMULATE\_AND\_REPLY**

큐 관리자가 라우트 추적 메시징에 대해 사용 가능한 경우 활동 정보가 라우트 추적 메시지의 메시지 데이터에 누적되고 다음 중 하나가 발생하는 경우 라우트 추적 응답 메시지가 생성됩니다.

- 라우트 추적 메시지는 WebSphere MQ 버전 6(이상의) 큐 관리자에 의해 제거됩니다.
- 라우트 추적 메시지는 WebSphere MQ 버전 6(이상) 큐 관리자에 의해 로컬 큐(대상 큐 또는 데드-레터 큐)로 들어갑니다.
- 라우트 추적 메시지에 대해 수행되는 활동의 수가 *MaxActivities*의 값을 초과합니다.

이 값이 지정되는 경우 라우트 추적 메시지 데이터는 다음으로 구성됩니다.

- *TraceRoute* PCF 그룹.
- 0개 이상의 *Activity* PCF 그룹.

#### **MQRROUTE\_ACCUMULATE\_NONE**

라우트 추적 메시지의 메시지 데이터에 활동 정보가 누적되지 않습니다.

이 값이 지정되는 경우 라우트 추적 메시지 데이터는 다음으로 구성됩니다.

- *TraceRoute* PCF 그룹.

#### **Forward**

라우트 추적 메시지가 전달될 수 있는 위치를 지정합니다. 가능한 값은 다음과 같습니다.

#### **MQRROUTE\_FORWARD\_IF\_SUPPORTED**

라우트 추적 메시지는 *TraceRoute* 그룹의 *Deliver* 매개변수의 값을 존중하는 큐 관리자로서만 전달됩니다.

#### **MQRROUTE\_FORWARD\_ALL**

라우트 추적 메시지가 *Deliver* 매개변수의 값이 존중되는지 여부와 상관없이 모든 큐 관리자로 전달됩니다.

큐 관리자는 라우트 추적 메시지를 리모트 큐 관리자로 전달할지 여부를 판별할 때 다음 알고리즘을 사용합니다.

1. 리모트 큐 관리자가 라우트 추적 메시지를 지원할 수 있는지 여부를 판별합니다.
  - 리모트 큐 관리자가 라우트 추적 메시지를 지원할 수 있는 경우 알고리즘은 65 페이지의 『4』 단계에서 계속합니다.
  - 리모트 큐 관리자가 라우트 추적 메시지를 지원할 수 없는 경우 알고리즘은 65 페이지의 『2』 단계에서 계속합니다.
2. *TraceRoute* 그룹의 *Deliver* 매개변수가 MQRROUTE\_DELIVER\_REJ\_UNSUP\_MASK 비트 마스트의 인식되지 않는 전달 옵션을 포함하는지 판별합니다.
  - 인식되지 않는 전달 옵션이 발견된 경우 라우트 추적 메시지는 MQFB\_UNSUPPORTED\_DELIVERY 피드백과 함께 거부됩니다.
  - 인식되지 않는 전달 옵션이 발견되지 않는 경우 알고리즘은 65 페이지의 『3』 단계에서 계속됩니다.
3. 라우트 추적 메시지의 *TraceRoute* PCF 그룹에 있는 *Deliver* 매개변수의 값을 판별합니다.
  - *Deliver*가 MQRROUTE\_DELIVER\_YES로 지정되는 경우, 라우트 추적 메시지는 리모트 큐 관리자로 전달됩니다.
  - *Deliver*가 MQRROUTE\_DELIVER\_NO로 지정되는 경우, 알고리즘이 65 페이지의 『4』 단계에서 계속됩니다.
4. *TraceRoute* 그룹의 *Forward* 매개변수가 MQRROUTE\_FORWARDING\_REJ\_UNSUP\_MASK 비트 마스크에 인식되지 않는 전달 옵션을 포함하는지 판별합니다.
  - 인식되지 않는 전달 옵션이 발견된 경우 라우트 추적 메시지는 MQFB\_UNSUPPORTED\_FORWARDING 피드백과 함께 거부됩니다.
  - 인식되지 않는 전달 옵션이 발견되지 않는 경우 알고리즘은 65 페이지의 『5』 단계에서 계속됩니다.
5. 라우트 추적 메시지의 *TraceRoute* PCF 그룹에 있는 *Forward* 매개변수의 값을 판별합니다.
  - *Forward*가 MQRROUTE\_FORWARD\_IF\_SUPPORTED로 지정되는 경우, 라우트 추적 메시지는 MQFB\_NOT\_FORWARDED 피드백과 함께 거부됩니다.
  - *Forward*가 MQRROUTE\_FORWARD\_ALL로 지정되는 경우, 라우트 추적 메시지는 리모트 큐 관리자로 전달될 수 있습니다.

## Deliver

라우트 추적 메시지가 그의 의도된 목적지에 도달하는 경우 취할 조치를 지정합니다. 사용자 작성 애플리케이션이 라우트 추적 메시지를 그의 대상 큐에 위치시키기 전에 이 속성을 확인해야 합니다. 가능한 값은 다음과 같습니다.

### **MQROUTE\_DELIVER\_YES**

라우트 추적 메시지가 도달하면 대상 큐에 메시지를 넣습니다. 대상 큐에 Get 조작을 수행하는 모든 애플리케이션이 라우트 추적 메시지를 검색할 수 있습니다.

### **MQROUTE\_DELIVER\_NO**

도착 시, 라우트 추적 메시지가 대상 큐로 전달되지 않습니다. 해당 보고서 옵션에 따라 메시지가 처리됩니다.

## 라우트 추적 응답 메시지에 대한 공용 큐 설정

보고서가 로컬 시스템 큐로 전달될 때 특정 메시지와 관련된 라우트 추적 응답 메시지의 위치를 판별하려면 단일 노드에서 공용 큐를 사용하는 것이 더 효율적입니다.

### 시작하기 전에

큐 관리자를 라우트 추적 메시징에 대해 사용 가능하게 하고 생성된 모든 라우트 추적 응답 메시지가 로컬 시스템 큐 SYSTEM.ADMIN.TRACE.ROUTE.QUEUE로 전달되도록 지정하려면 ROUTEREC 매개변수를 설정하십시오.

### 이 태스크 정보

큐 관리자 네트워크의 많은 큐 관리자가 로컬 시스템 큐로 라우트 추적 응답 메시지를 전달하도록 설정되는 경우, 특정 메시지와 관련된 라우트 추적 응답 메시지의 위치를 판별하는 것은 시간 소비적일 수 있습니다. 그렇지 않으면 공용 큐를 호스트하는 큐 관리자인 단일 노드를 사용하십시오. 큐 관리자 네트워크의 모든 큐 관리자가 이 공용 큐로 라우트 추적 응답 메시지를 전달할 수 있습니다. 공용 큐 사용의 이점은 큐 관리자가 메시지에 지정된 응답 대상 큐로 라우트 추적 응답 메시지를 전달할 필요가 없고 메시지와 관련된 라우트 추적 응답 메시지의 위치를 판별할 때 하나의 큐만 조회한다는 점입니다.

공용 큐를 설정하려면 다음 단계를 수행하십시오.

### 프로시저

1. 큐 관리자를 단일 노드로 선택 또는 정의하십시오.
2. 단일 노드에서 공용 큐에 사용할 큐를 선택 또는 정의하십시오.
3. 라우트 추적 응답 메시지를 공용 큐로 전달하는 모든 큐 관리자에서 로컬 시스템 큐 SYSTEM.ADMIN.TRACE.ROUTE.QUEUE를 리모트 큐 정의로 다시 정의하십시오.
  - a) 단일 노드의 이름을 리모트 큐 관리자 이름으로 지정하십시오.
  - b) 공용 큐의 이름을 리모트 큐 이름으로 지정하십시오.

### 기록된 정보 획득 및 사용

라우트 추적 메시지에 대한 기록된 활동 정보를 획득하려면 다음 기술 중 하나를 사용하십시오.

활동 정보가 획득되지 않는 상황이 라우트 추적 응답 메시지에도 적용됨을 참고하십시오.

라우트 추적 메시지가 활동 기록 및 라우트 추적 메시징 모두에 대해 사용 불가능한 큐 관리자에 의해 처리될 때는 활동 정보가 기록되지 않습니다.

### 라우트 추적 응답 메시지에서 정보 획득

활동 정보를 획득하려면 라우트 추적 응답 메시지를 찾습니다. 그런 다음 메시지를 검색하고 활동 정보를 분석하십시오.

### 이 태스크 정보

라우트 추적 응답 메시지의 위치를 아는 경우에만 라우트 추적 응답 메시지에서부터 활동 정보를 획득할 수 있습니다. 다음과 같이 메시지를 찾고 활동 정보를 처리하십시오.

## 프로시저

1. 라우트 추적 메시지의 메시지 디스크립터에서 지정된 응답 대상 큐를 확인하십시오. 라우트 추적 응답 메시지가 응답 대상 큐에 없는 경우 다음 위치를 검사하십시오.
  - 라우트 추적 메시지의 대상 큐 관리자에 있는 로컬 시스템 큐 SYSTEM.ADMIN.TRACE.ROUTE.QUEUE
  - 라우트 추적 응답 메시지에 대한 공용 큐를 설정한 경우, 해당 공용 큐
  - 큐 관리자 네트워크에 있는 다른 모든 큐 관리자의 로컬 시스템 큐 SYSTEM.ADMIN.TRACE.ROUTE.QUEUE. 이것은 라우트 추적 메시지가 데드-레터 큐에 넣어졌거나 최대 활동 수를 초과한 경우에 발생할 수 있습니다.
2. 라우트 추적 응답 메시지를 검색하십시오.
3. WebSphere MQ 라우트 표시 애플리케이션을 사용하여 기록된 활동 정보를 표시하십시오.
4. 활동 정보를 연구하여 필요한 정보를 얻으십시오.

## 라우트 추적 메시지에서 정보 획득

활동 정보를 획득하려면 라우트 추적 메시지를 찾는데, 이것은 *TraceRoute* PCF 그룹에서 적절한 매개변수를 가져야 합니다. 그런 다음 메시지를 검색하고 활동 정보를 분석하십시오.

## 이 태스크 정보

라우트 추적 메시지의 위치를 알고 있고 *TraceRoute* PCF 그룹의 *Accumulate* 매개변수가 `MQRROUTE_ACCUMULATE_IN_MSG` 또는 `MQRROUTE_ACCUMULATE_AND_REPLY`로 지정된 경우에만 라우트 추적 메시지에서 활동 정보를 획득할 수 있습니다.

라우트 추적 메시지가 대상 큐로 전달되기 위해서는 *TraceRoute* PCF 그룹의 *Deliver* 매개변수가 `MQRROUTE_DELIVER_YES`로 지정되어야 합니다.

## 프로시저

1. 대상 큐를 검사하십시오. 라우트 추적 메시지가 대상 큐에 없는 경우, 활동 기록에 대해 사용 가능한 라우트 추적 메시지를 사용하여 라우트 추적 메시지를 찾으려고 시도할 수 있습니다. 생성된 활동 보고서를 사용하여 라우트 추적 메시지의 마지막으로 알려진 위치를 판별하십시오.
2. 라우트 추적 메시지를 검색하십시오.
3. WebSphere MQ 라우트 표시 애플리케이션을 사용하여 기록된 활동 정보를 표시하십시오.
4. 활동 정보를 연구하여 필요한 정보를 얻으십시오.

## 활동 보고서에서 정보 획득

활동 정보를 획득하려면 활동 보고서를 찾습니다. 이 활동 보고서는 메시지 디스크립터에서 지정되는 보고서 옵션을 가져야 합니다. 그런 다음 활동 보고서를 검색하고 활동 정보를 분석하십시오.

## 이 태스크 정보

활동 보고서의 위치를 알고 있고 보고서 옵션 `MQRO_ACTIVITY`가 라우트 추적 메시지의 메시지 디스크립터에서 지정된 경우에만 활동 보고서에서 활동 정보를 획득할 수 있습니다.

## 프로시저

1. 라우트 추적 메시지에 대해 생성된 활동 보고서를 찾아서 정렬하십시오.
  - 활동 보고서의 위치를 찾으면 수동으로 또는 WebSphere MQ 라우트 표시 애플리케이션을 사용하여 순서를 지정하고 활동 정보를 자동으로 표시할 수 있습니다.
2. 활동 정보를 연구하여 필요한 정보를 얻으십시오.

## 추가 활동 정보

라우트 추적 메시지가 큐 관리자 네트워크를 통해 라우트될 때, 사용자 애플리케이션은 라우트 추적 메시지 또는 활동 보고서의 메시지 데이터에 *Activity* 그룹을 쓸 때 하나 이상의 추가 PCF 매개변수를 포함하여 추가 정보를 기록할 수 있습니다.

추가 활동 정보는 시스템 관리자가 라우트 추적 메시지가 취한 라우트 또는 해당 라우트를 사용한 이유를 식별하는 데 도움이 될 수 있습니다.

IBM WebSphere MQ 표시 라우트 애플리케이션을 사용하여 라우트 추적 메시지에 대해 기록된 정보를 표시하는 경우, 각 매개변수의 매개변수 ID가 IBM WebSphere MQ 표시 라우트 애플리케이션에 의해 인식되지 않는 경우 모든 추가 PCF 매개변수는 숫자 ID만을 갖고 표시될 수 있습니다. 매개변수 ID를 인식하려면 추가 정보가 다음 PCF 매개변수를 사용하여 기록되어야 합니다. 이들 PCF 매개변수를 *Activity* PCF 그룹의 적절한 위치에 포함하십시오.

#### GroupName

표 13. 그룹 이름	
설명	추가 정보를 지정하는 그룹화 매개변수
ID	MQGACF_VALUE_NAMING.
데이터 유형	MQCFGR
그룹의 매개변수	<i>ParameterName</i> <i>ParameterValue</i>

#### ParameterName

표 14. 매개변수 이름	
설명	<i>ParameterValue</i> 의 값을 컨텍스트로 배치하는 IBM WebSphere MQ 표시 라우트 애플리케이션이 표시할 이름을 포함합니다.
ID	MQCA_VALUE_NAME.
데이터 유형	MQCFST
PCF 그룹에 포함:	<i>GroupName</i> .
가치:	표시될 이름.

#### ParameterValue

표 15. 매개변수 값	
설명	IBM WebSphere MQ 표시 라우트 애플리케이션이 표시할 값을 포함합니다.
ID:	추가 정보에 대한 PCF 구조 ID.
데이터 유형:	추가 정보에 대한 PCF 구조 데이터 유형.
PCF 그룹에 포함:	<i>GroupName</i> .
가치:	표시될 값.

### 추가 활동 정보 기록의 예

다음 예는 사용자 애플리케이션이 라우트 추적 메시지를 위해 활동을 수행할 때 추가 정보를 기록할 수 있는 방법을 설명합니다. 두 예 모두에서, IBM WebSphere MQ 표시 라우트 애플리케이션이 라우트 추적 메시지를 생성하고 리턴되는 활동 정보를 표시하는 데 사용됩니다.

#### 예제 1

추가 활동 정보는 WebSphere MQ 표시 라우트 애플리케이션에서 인식하는 매개변수 ID 그렇지 않다의 형식으로 사용자 애플리케이션에 의해 기록됩니다.

1. WebSphere MQ 라우트 표시 애플리케이션은 라우트 추적 메시지를 생성하여 큐 관리자 네트워크로 넣기 위해 사용됩니다. 필요한 옵션이 다음을 요청하도록 설정됩니다.
  - 라우트 추적 메시지의 메시지 데이터에 활동 정보가 누적됩니다.

- 도착 시 대상 큐에서 라우트 추적 메시지가 제거되고, 라우트 추적 응답 메시지가 생성되어 지정된 응답 대상 큐로 전달됩니다.
- 라우트 추적 응답 메시지를 수신할 때 WebSphere MQ 라우트 표시 애플리케이션이 추적된 활동 정보를 표시합니다.

라우트 추적 메시지는 큐 관리자 네트워크에 넣어집니다.

2. 라우트 추적 메시지가 큐 관리자 네트워크를 통해 라우팅되므로 라우트 추적 메시징에 사용되는 사용자 애플리케이션은 메시지에 관한 낮은 상세 활동을 수행합니다. 라우트 추적 메시지에 표준 활동 정보를 쓰는 것 외에, 사용자 애플리케이션은 Activity 그룹의 끝에 다음 PCF 매개변수를 씁니다.

**ColorValue**

```

ID
  65536
데이터 유형
  MQCFST
가치
  'Red'

```

이 추가 PCF 매개변수는 수행된 활동에 대한 추가 정보를 제공하지만 WebSphere MQ 표시 라우트 애플리케이션에서 인식하는 매개변수 ID 그렇지 않다 의 형식으로 작성됩니다.

3. 라우트 추적 메시지가 대상 큐에 도달하고 라우트 추적 응답 메시지가 WebSphere MQ 라우트 표시 애플리케이션에 리턴됩니다. 추가 활동 정보는 다음과 같이 표시됩니다.

```
65536: 'Red'
```

WebSphere MQ 라우트 표시 애플리케이션이 PCF 매개변수의 매개변수 ID를 인식하지 않고 숫자 값으로 표시합니다. 추가 정보의 컨텍스트가 명확하지 않습니다.

WebSphere MQ 라우트 표시 애플리케이션이 PCF 매개변수의 매개변수 ID를 인식하는 경우에는 [69 페이지](#)의 『예제 2』의 내용을 참조하십시오.

**예제 2**

추가 활동 정보는 IBM WebSphere MQ 표시 라우트 애플리케이션이 인식하는 매개변수 ID가 있는 형식으로 사용자 애플리케이션에 의해 기록됩니다.

1. IBM WebSphere MQ 표시 라우트 애플리케이션은 [68 페이지](#)의 『예제 1』에서와 동일한 방식으로 라우트 추적 메시지를 생성하여 큐 관리자 네트워크에 이 메시지를 넣는 데 사용됩니다.
2. 라우트 추적 메시지가 큐 관리자 네트워크를 통해 라우팅되므로 라우트 추적 메시징에 사용되는 사용자 애플리케이션은 메시지에 관한 낮은 상세 활동을 수행합니다. 사용자 애플리케이션은 라우트 추적 메시지에 표준 활동 정보를 쓰는 것 외에 활동 그룹의 끝에 다음 PCF 매개변수를 씁니다.

**ColorInfo**

표 16. 색상 정보	
설명	색상에 대한 정보를 지정하는 그룹화된 매개변수.
ID:	MQGACF_VALUE_NAMING.
데이터 유형:	MQCFGR.
그룹의 매개변수:	ColorName ColorValue

## ColorName

표 17. 색상 이름	
설명	<b>ColorValue</b> 값을 컨텍스트로 넣은 <b>IBM WebSphere MQ</b> 표시 라우트 애플리케이션이 표시할 이름을 포함합니다.
ID:	MQCA_VALUE_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>ColorInfo</i> .
가치:	'Color'

## ColorValue

표 18. 색상 값	
설명	<b>IBM WebSphere MQ</b> 표시 라우트 애플리케이션이 표시할 값을 포함합니다.
ID:	65536.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>ColorInfo</i> .
가치:	'Red'

이러한 추가 PCF 매개변수는 수행된 활동에 대한 추가 정보를 제공합니다. 이러한 PCF 매개변수는 IBM WebSphere MQ 표시 라우트 애플리케이션이 인식하는 매개변수 ID가 있는 형식으로 작성됩니다.

3. 라우트 추적 메시지가 대상 큐에 도달하고 라우트 추적 응답 메시지가 IBM WebSphere MQ 표시 라우트 애플리케이션에 리턴됩니다. 추가 활동 정보는 다음과 같이 표시됩니다.

```
Color: 'Red'
```

IBM WebSphere MQ 표시 라우트 애플리케이션은 추가 활동 정보의 값을 포함한 PCF 구조의 매개변수 ID에 해당 이름이 있음을 인식합니다. 숫자 값 대신 해당 이름이 표시됩니다.

## WebSphere MQ 표시 라우트 애플리케이션

명령행 인터페이스를 통해 라우트 추적 메시지 및 라우트 추적 메시지와 연관된 활동 정보에 대해 작업하려면 WebSphere MQ 라우트 표시 애플리케이션(**dspmqrte**)을 사용하십시오.

**참고:** 큐 관리자에 대해 클라이언트 애플리케이션을 실행하려면 클라이언트 첨부 기능이 설치되어 있어야 합니다.

다음과 같은 목적으로 WebSphere MQ 라우트 표시 애플리케이션을 사용할 수 있습니다.

- 라우트 추적 메시지를 구성 및 생성하여 큐 관리자 네트워크에 넣으려는 경우.  
라우트 추적 메시지를 큐 관리자 네트워크에 넣어서 활동 정보를 수집하고 이를 사용하여 라우트 추적 메시지가 이용한 라우트를 판별할 수 있습니다. 다음과 같이 라우트 추적 메시지의 특성을 지정할 수 있습니다.
  - 라우트 추적 메시지의 목적지.
  - 라우트 추적 메시지가 다른 메시지를 가장하는 방식.
  - 큐 관리자 네트워크를 통해 라우트할 때 라우트 추적 메시지의 핸들링 방식.
  - 활동 정보를 기록하기 위해 활동 기록 또는 라우트 추적 메시징을 사용하는지 여부.
- 라우트 추적 메시지에 관련된 활동 정보의 순서화 및 표시.

WebSphere MQ 라우트 표시 애플리케이션이 라우트 추적 메시지를 큐 관리자 네트워크에 넣었으며 관련된 활동 정보가 리턴된 후에 정보의 순서가 지정되고 즉시 표시될 수 있습니다. 또는 WebSphere MQ 라우트 표시 애플리케이션이

플리케이션을 이전에 생성된 라우트 추적 메시지와 연관된 활동 정보의 순서를 지정하고 표시하는 데 사용할 수 있습니다.

#### 관련 참조

[dspmqrte](#)

### 라우트 추적 메시지에 대한 매개변수

라우트 추적 메시지가 큐 관리자 네트워크를 통해 라우트될 때 처리되는 방법을 포함하여 라우트 추적 메시지의 특성을 판별하려면 WebSphere MQ 라우트 표시 애플리케이션인 **dspmqrte**에 의해 제공되는 매개변수에 대한 개요를 보려면 이 페이지를 사용하십시오.

#### 관련 참조

[dspmqrte](#)

### 큐 관리자 연결

WebSphere MQ 라우트 표시 애플리케이션이 연결되는 큐 관리자를 지정하려면 이 페이지를 사용하십시오.

#### -c

WebSphere MQ 라우트 표시 애플리케이션이 클라이언트 애플리케이션으로서 연결되도록 지정합니다.

이 매개변수를 지정하지 않으면 WebSphere MQ 라우트 표시 애플리케이션이 클라이언트 애플리케이션으로 연결되지 않습니다.

#### -m *QMgrName*

WebSphere MQ 표시 라우트 애플리케이션이 연결하는 큐 관리자의 이름. 이름은 최대 48자입니다.

이 매개변수를 지정하지 않으면 기본 큐 관리자가 사용됩니다.

### 대상 목적지

이 페이지를 사용하여 라우트 추적 메시지의 대상 목적지를 지정합니다.

#### -q *TargetQName*

WebSphere MQ 라우트 표시 애플리케이션이 라우트 추적 메시지를 큐 관리자 네트워크로 송신하는 데 사용되는 경우, *TargetQName*이 대상 큐의 이름을 지정합니다.

#### -ts *TargetTopicString*

토픽 문자열을 지정합니다.

#### -qm *TargetQMGr*

대상 목적지를 규정하며 그러면 일반 큐 관리자 이름 분석이 적용됩니다. 대상 목적지는 *-q TargetQName* 또는 *-ts TargetTopicString*로 지정됩니다.

이 매개변수를 지정하지 않으면 WebSphere MQ 라우트 표시 애플리케이션이 연결되는 큐 관리자가 대상 큐 관리자로 사용됩니다.

#### -o

대상 목적지가 지정된 목적지로 바인딩되지 않음을 지정합니다. 일반적으로 이 매개변수는 라우트 추적 메시지를 클러스터에 넣을 때 사용합니다. 대상 목적지는 MQOO\_BIND\_NOT\_FIXED 옵션으로 열립니다.

이 매개변수를 지정하지 않을 경우 대상 목적지가 특정 목적지로 바인딩됩니다.

### 발행 토픽

발행/구독 애플리케이션의 경우, 메시지를 발행할 WebSphere MQ 라우트 표시 애플리케이션에 대한 라우트 추적 메시지의 토픽 문자열을 지정하려면 이 페이지를 사용하십시오.

#### -ts *TopicName*

WebSphere MQ 라우트 표시 애플리케이션이 라우트 추적 메시지를 발행할 때 연결될 토픽 문자열을 지정하고 이 애플리케이션을 토픽 모드로 설정합니다. 이 모드에서 애플리케이션은 발행 요청에서 생성된 모든 메시지를 추적합니다.

또한 WebSphere MQ 라우트 표시 애플리케이션을 사용하여 발행 메시지에 대해 생성된 활동 보고서로부터 결과를 표시할 수 있습니다.

## 메시지 가장

이 페이지를 사용하여 예를 들어, 원래 메시지가 의도한 목적지에 도달하지 않았을 때 메시지를 가장하도록 라우트 추적 메시지를 구성합니다.

라우트 추적 메시징의 한 가지 용도는 의도한 목적지에 도달하지 않은 메시지의 마지막 알려진 위치를 판별하도록 돕는 것입니다. IBM WebSphere MQ 표시 라우트 애플리케이션은 원래 메시지를 가장하도록 라우트 추적 메시지를 구성하는 데 유용한 매개변수를 제공합니다. 메시지를 가장할 때 다음 매개변수를 사용할 수 있습니다.

### -l **Persistence**

생성된 라우트 추적 메시지의 지속성을 지정합니다. *Persistence*에 가능한 값은 다음과 같습니다.

#### **yes**

생성된 라우트 추적 메시지가 지속적입니다. (MQPER\_PERSISTENT).

#### **아니오**

생성된 라우트 추적 메시지가 지속적이지 **않습니다**. (MQPER\_NOT\_PERSISTENT).

#### **q**

생성된 라우트 추적 메시지가 *-q TargetQName* 또는 *-ts TargetTopicString*에 지정된 목적지로부터 지속 값을 상속합니다. (MQPER\_PERSISTENCE\_AS\_Q\_DEF).

리턴된 라우트 추적 응답 메시지 또는 보고 메시지도 원래 라우트 추적 메시지와 동일한 지속 값을 공유합니다.

*Persistence*가 **yes**로 지정되면 *-rq ReplyToQ* 매개변수를 지정해야 합니다. 응답 대상 큐를 임시 동적 큐로 해석하지 않아야 합니다.

이 매개변수를 지정하지 않을 경우 생성된 라우트 추적 메시지는 지속적이지 **않습니다**.

### -p **Priority**

라우트 추적 메시지의 우선순위를 지정합니다. *Priority*의 값은 0 이상이거나 MQPRI\_PRIORITY\_AS\_Q\_DEF입니다. MQPRI\_PRIORITY\_AS\_Q\_DEF는 *-q TargetQName* 또는 *-ts TargetTopicString*에 지정된 목적지에서 우선순위 값을 가져옴을 지정합니다.

이 매개변수를 지정하지 않으면 *-q TargetQName* 또는 *-ts TargetTopicString*에 지정된 목적지에서 우선순위 값을 가져옵니다.

### -xs **Expiry**

라우트 추적 메시지의 만료 시간을 초 단위로 지정합니다.

이 매개변수를 지정하지 않으면 만기 시간이 60초로 지정됩니다.

### -ro none | **ReportOption**

#### **none**

보고서 옵션이 설정되지 않도록 지정합니다.

#### **ReportOption**

라우트 추적 메시지에 대한 보고서 옵션을 지정합니다. 심표를 구분 기호로 사용하여 다중 보고서 옵션을 지정할 수 있습니다. *ReportOption*에 가능한 값은 다음과 같습니다.

#### **activity**

보고서 옵션 MQRO\_ACTIVITY가 설정됩니다.

#### **coa**

보고서 옵션 MQRO\_COA\_WITH\_FULL\_DATA가 설정됩니다.

#### **cod**

보고서 옵션 MQRO\_COD\_WITH\_FULL\_DATA가 설정됩니다.

#### **exception**

보고서 옵션 MQRO\_EXCEPTION\_WITH\_FULL\_DATA가 설정됩니다.

#### **expiration**

보고서 옵션 MQRO\_EXPIRATION\_WITH\_FULL\_DATA가 설정됩니다.

#### **discard**

보고서 옵션 MQRO\_DISCARD\_MSG가 설정됩니다.

-ro ReportOption 또는 -ro none 중 어느 것도 지정하지 않으면 MQRO\_ACTIVITY 및 MQRO\_DISCARD\_MSG 보고서 옵션이 지정됩니다.

IBM WebSphere MQ 표시 라우트 애플리케이션은 라우트 추적 메시지에 사용자 데이터를 추가하도록 허용하지 않습니다. 라우트 추적 메시지에 사용자 데이터를 추가해야 하는 경우에는 라우트 추적 메시지를 수동으로 생성해야 합니다.

### 기록된 활동 정보

이 페이지를 사용하여 기록된 활동 정보를 리턴하는 데 사용되는 메소드를 지정한 후 이 메소드를 사용하여 라우트 추적 메시지가 이용한 라우트를 판별할 수 있습니다.

기록된 활동 정보는 다음과 같이 리턴될 수 있습니다.

- 활동 보고서에
- 라우트 추적 응답 메시지에
- 라우트 추적 메시지 자체에(대상 큐에 넣음)

**dspmqrte**를 사용할 때에는 기록된 활동 정보를 리턴하는 데 사용된 메소드가 다음 매개변수를 사용하여 판별됩니다.

#### -ro 를 사용하여 지정된 activity 보고서 옵션

활동 정보가 활동 보고서를 사용하여 리턴됨을 지정합니다. 기본적으로 활동 기록이 사용으로 설정됩니다.

#### -ac -ar

활동 정보가 라우트 추적 메시지에 누적되고 라우트 추적 응답 메시지가 생성됨을 지정합니다.

#### -ac

활동 정보가 라우트 추적 메시지 내에 누적되도록 지정합니다.

이 매개변수를 지정하지 않으면 활동 정보가 라우트 추적 메시지 내에 누적되지 **않습니다**.

#### -ar

다음 상황에서 모든 누적 활동 정보가 포함된 라우트 추적 응답 메시지가 생성되도록 요청합니다.

- IBM WebSphere MQ 큐 관리자가 라우트 추적 메시지를 제거합니다.
- IBM WebSphere MQ 큐 관리자가 라우트 추적 메시지를 로컬 큐(때상 큐 또는 데드-레터 큐)에 넣습니다.
- 라우트 추적 메시지에 수행한 활동의 수가 -s Activities에 지정된 값을 초과합니다.

#### -ac -d yes

활동 정보가 라우트 추적 메시지에 누적되고 라우트 추적 메시지가 도달하면 대상 큐에 메시지를 넣음을 지정합니다.

#### -ac

활동 정보가 라우트 추적 메시지 내에 누적되도록 지정합니다.

이 매개변수를 지정하지 않으면 활동 정보가 라우트 추적 메시지 내에 누적되지 **않습니다**.

#### -d yes

큐 관리자가 라우트 추적 메시지를 지원하지 않는 경우에도 도달 시, 라우트 추적 메시지가 대상 큐로 들어갑니다.

이 매개변수를 지정하지 않으면 라우트 추적 메시지를 대상 큐에 넣지 **않습니다**.

그런 다음 대상 큐에서 라우트 추적 메시지를 검색하여 기록된 활동 정보를 얻을 수 있습니다.

필요에 따라 다음 메소드를 결합할 수 있습니다.

또한 다음 매개변수를 사용하여 기록된 활동 정보의 세부사항 레벨을 지정할 수 있습니다.

#### -t Detail

기록한 활동을 지정합니다. *Detail*에 가능한 값은 다음과 같습니다.

#### 낮음

사용자 정의 애플리케이션이 수행한 활동만 기록됩니다.

## medium

**low**에 지정된 활동이 기록됩니다. 또한 발행 활동과 MCA가 수행한 활동이 기록됩니다.

## 높음

**low** 및 **medium**에 지정된 활동이 기록됩니다. MCA는 이 레벨의 세부사항에서 추가 활동 정보를 표시하지 않습니다. 이 옵션은 추가 활동 정보만 표시할 사용자 정의 애플리케이션에 사용 가능합니다. 예를 들어, 사용자 정의 애플리케이션이 특정 메시지 특성을 고려하여 메시지가 이용하는 라우트를 판별하면 라우팅 논리가 이 세부사항 레벨에 포함될 수 있습니다.

이 매개변수를 지정하지 않으면 중간 레벨 활동이 기록됩니다.

기본적으로 IBM WebSphere MQ 표시 라우트 애플리케이션은 임시 동적 큐를 사용하여 리턴된 메시지를 저장합니다. IBM WebSphere MQ 표시 라우트 애플리케이션이 종료되면 임시 동적 큐가 닫히고 메시지가 제거됩니다. IBM WebSphere MQ 표시 라우트 애플리케이션의 현재 실행이 종료되어도 리턴된 메시지가 필요하다면 다음 매개변수를 사용하여 영구적 큐를 지정해야 합니다.

### -rq ReplyToQ

라우트 추적 메시지에 대해 모든 응답이 송신되는 응답 대상 큐의 이름을 지정합니다. 라우트 추적 메시지가 지속적인 경우나 -n 매개변수가 지정된 경우에는 임시 동적 큐가 **아닌** 응답 대상 큐를 지정해야 합니다.

이 매개변수를 지정하지 않으면 시스템 기본 모델 큐인 SYSTEM.DEFAULT.MODEL.QUEUE를 사용하여 동적 응답 대상 큐가 작성됩니다.

### -rqm ReplyToQMgr

응답 대상 큐가 상주하는 큐 관리자의 이름을 지정합니다. 이름은 최대 48자입니다.

이 매개변수를 지정하지 않으면 IBM WebSphere MQ 표시 라우트 애플리케이션이 연결된 큐 관리자를 응답 대상 큐 관리자로 사용합니다.

## 라우트 추적 메시지의 핸들링 방식

이 페이지를 사용하여 큐 관리자 네트워크를 통해 라우팅되는 라우트 추적 메시지를 어떻게 핸들링할지 제어합니다.

다음 매개변수는 큐 관리자 네트워크에서 라우트 추적 메시지를 라우팅할 수 있는 위치를 제한할 수 있습니다.

### -d Deliver

라우트 추적 메시지가 도착 시 대상 큐로 전달되는지 여부를 지정합니다. *Deliver*에 가능한 값은 다음과 같습니다.

<b>yes</b>	큐 관리자가 라우트 추적 메시징을 지원하지 않는 경우에도 도달 시, 라우트 추적 메시지가 대상 큐로 들어갑니다.
<b>no</b>	도달 시, 라우트 추적 메시지를 대상 큐로 넣지 <b>않습니다</b> .

이 매개변수를 지정하지 않으면 라우트 추적 메시지를 대상 큐에 넣지 **않습니다**.

### -f Forward

라우트 추적 메시지가 전달될 수 있는 큐 관리자의 유형을 지정합니다. 큐 관리자가 메시지를 리모트 큐 관리자로 전달할 것인지 여부를 판별하는 데 사용할 수 있는 알고리즘에 대한 자세한 내용은 [63 페이지의 『TraceRoute PCF 그룹』](#)의 내용을 참조하십시오. *Forward*에 가능한 값은 다음과 같습니다.

#### 모두

라우트 추적 메시지가 모든 큐 관리자에 전달됩니다.

**경고:** 버전 6.0 미만의 IBM WebSphere MQ 큐 관리자에 전달된 경우, 라우트 추적 메시징은 인식되지 못하고 -d *Deliver* 매개변수 값에도 불구하고 로컬 큐로 전달될 수 있습니다.

#### 지원됨

*TraceRoute* PCF 그룹의 *Deliver* 매개변수를 존중할 큐 관리자에만 라우트 추적 메시지가 전달됩니다.

이 매개변수를 지정하지 않으면 *Deliver* 매개변수를 존중할 큐 관리자에만 라우트 추적 메시지가 전달됩니다.

다음 매개변수는 라우트 추적 메시지가 큐 관리자 네트워크에 무제한으로 남아 있지 못하게 할 수 있습니다.

#### **-s Activities**

라우트 추적 메시지가 제거되기 전에 이를 대신하여 수행될 수 있는 기록된 최대 활동 수를 지정합니다. 이는 라우트 추적 메시지가 무한 루프에 빠졌을 때 무제한으로 전달되지 않게 합니다. *Activities*의 값이 1 이상이거나 *MQROUTE\_UNLIMITED\_ACTIVITIES*입니다. *MQROUTE\_UNLIMITED\_ACTIVITIES*는 라우트 추적 메시지를 대신하여 무제한 수의 활동이 수행될 수 있음을 지정합니다.

이 매개변수를 지정하지 않으면 라우트 추적 메시지를 대신하여 무제한 수의 활동이 수행될 수 있습니다.

#### **-xs Expiry**

라우트 추적 메시지의 만료 시간을 초 단위로 지정합니다.

이 매개변수를 지정하지 않으면 만기 시간이 60초로 지정됩니다.

#### **-xp PassExpiry**

라우트 추적 메시지의 만기 시간의 라우트 추적 응답 메시지에 전달되는지 여부를 지정합니다. *PassExpiry*에 가능한 값은 다음과 같습니다.

##### **yes**

보고서 옵션 *MQRO\_PASS\_DISCARD\_AND\_EXPIRY*는 라우트 추적 메시지의 메시지 디스크립터에 지정할 수 있습니다.

라우트 추적 메시지에 대해 라우트 추적 응답 메시지나 활동 보고서가 생성되면 *MQRO\_DISCARD* 보고서 옵션(지정된 경우) 및 남은 만기 시간이 전달됩니다.

이 값은 기본값입니다.

##### **아니오**

보고서 옵션 *MQRO\_PASS\_DISCARD\_AND\_EXPIRY*가 지정되지 않습니다.

라우트 추적 메시지에 대해 라우트 추적 응답 메시지가 생성된 경우 라우트 추적 메시지의 만기 시간 및 제거 옵션이 전달되지 **않습니다**.

이 매개변수를 지정하지 않으면 *MQRO\_PASS\_DISCARD\_AND\_EXPIRY*가 지정되지 않습니다.

#### **-ro 를 사용하여 지정된 discard 보고서 옵션**

*MQRO\_DISCARD\_MSG* 보고서 옵션을 지정합니다. 이 옵션은 라우트 추적 메시지가 큐 관리자 네트워크에 무제한으로 남아 있지 못하게 할 수 있습니다.

## **활동 정보 표시**

IBM WebSphere MQ 표시 라우트 애플리케이션은 큐 관리자 네트워크에 방금 넣은 라우트 추적 메시지에 대한 활동 정보를 표시하거나 이전에 생성한 라우트 추적 메시지에 대한 활동 정보를 표시할 수 있습니다. 사용자 작성 애플리케이션이 기록한 추가 정보를 표시할 수도 있습니다.

라우트 추적 메시지에 대해 리턴된 활동 정보의 표시 여부를 지정하려면 다음 매개변수를 지정하십시오.

#### **-n**

라우트 추적 메시지에 대해 리턴되는 활동 정보가 표시되지 않도록 지정합니다.

이 매개변수가 라우트 추적 응답 메시지, (*-ar*) 또는 (*-ro ReportOption*)의 보고서 생성 옵션에 대한 요청을 동반하는 경우 *-rq ReplyToQ*를 사용하여 특정(비모델) 응답 대상 큐를 지정해야 합니다. 기본적으로 활동 보고서 메시지만 요청됩니다.

라우트 추적 메시지를 지정된 대상 큐에 넣고 나면 라우트 추적 메시지의 메시지 ID를 포함하여 48자의 16진 문자열이 표시됩니다. 메시지 ID는 IBM WebSphere MQ 표시 라우트 애플리케이션이 *-i CorrelId* 매개변수를 사용하여 나중에 라우트 추적 메시징에 대한 활동 정보를 표시할 때 사용할 수 있습니다.

이 매개변수를 지정하지 않으면 라우트 추적 메시지에 리턴된 활동 정보가 *-v* 매개변수로 지정된 양식에 표시됩니다.

큐 관리자 네트워크에 방금 넣은 라우트 추적 메시지에 대한 활동 정보를 표시할 때에는 다음 매개변수가 지정될 수 있습니다.

### **-w WaitTime**

활동 보고서 또는 라우트 추적 응답 메시지가 지정된 응답 대상 큐에 리턴할 때까지 IBM WebSphere MQ 표시 라우트 애플리케이션이 대기할 시간(초)을 지정합니다.

이 매개변수를 지정하지 않으면 대기 시간이 라우트 추적 메시지의 만료 시간에 60초를 더한 시간으로 지정됩니다.

이전에 누적된 활동 정보를 표시할 때에는 다음 매개변수를 설정해야 합니다.

### **-q TargetQName**

IBM WebSphere MQ 표시 라우트 애플리케이션을 사용하여 이전에 수집한 활동 정보를 보는 경우 *TargetQName*이 활동 정보가 저장된 큐의 이름을 지정합니다.

### **-i CorrelId**

IBM WebSphere MQ 표시 라우트 애플리케이션을 사용하여 이전에 누적된 활동 정보만 표시하는 경우 이 매개변수를 사용합니다. *-q TargetQName*에 지정된 큐에는 많은 활동 보고서와 라우트 추적 응답 메시지가 있을 수 있습니다. *CorrelId*는 라우트 추적 메시지에 관련된 활동 보고서, 라우트 추적 응답 메시지를 식별하는데 사용됩니다. *CorrelId*에서 원래 라우트 추적 메시지의 메시지 ID를 지정하십시오.

*CorrelId*의 형식은 48자의 16진 문자열입니다.

다음 매개변수는 이전에 누적된 활동 정보를 표시할 때에나 라우트 추적 메시지에 대한 현재 활동 정보를 표시할 때 사용할 수 있습니다.

### **-b**

IBM WebSphere MQ 표시 라우트 애플리케이션이 메시지에 관련된 라우트 추적 응답 메시지나 활동 보고서를 찾아보기만 함을 지정합니다. 이는 나중에 활동 정보를 다시 표시할 수 있도록 합니다.

이 매개변수를 지정하지 않을 경우 IBM WebSphere MQ 표시 라우트 애플리케이션은 메시지에 관련된 라우트 추적 응답 메시지나 활동 보고서를 파괴적으로 가져옵니다.

### **-v summary | all | none | outline DisplayOption**

#### **summary**

라우트 추적 메시지가 라우트된 큐가 표시됩니다.

#### **모두**

사용 가능한 모든 정보가 표시됩니다.

#### **없음**

정보가 표시되지 않습니다.

#### **outline DisplayOption**

라우트 추적 메시지의 표시 옵션을 지정합니다. 쉼표를 구분 기호로 사용하여 다중 표시 옵션을 지정할 수 있습니다.

값을 제공하지 않으면 다음 사항이 표시됩니다.

- 애플리케이션 이름
- 각 조작의 유형
- 조작별 매개변수

*DisplayOption*에 가능한 값은 다음과 같습니다.

#### **activity**

*Activity* PCF 그룹의 모든 비PCF 그룹 매개변수가 표시됩니다.

#### **identifiers**

매개변수 ID MQBACF\_MSG\_ID 또는 MQBACF\_CORREL\_ID가 있는 값이 표시됩니다. 이 값은 *msgdelta*를 대체합니다.

#### **메시지**

*Message* PCF 그룹의 모든 비PCF 그룹 매개변수가 표시됩니다. 이 값이 지정된 경우 *msgdelta*를 지정할 수 없습니다.

#### **msgdelta**

마지막 조작 이후 변경된 *Message* PCF 그룹의 모든 비PCF 그룹 매개변수가 표시됩니다. 이 값이 지정된 경우 *message*를 지정할 수 없습니다.

### operation

Operation PCF 그룹의 모든 비PCF 그룹 매개변수가 표시됩니다.

### traceroute

TraceRoute PCF 그룹의 모든 비PCF 그룹 매개변수가 표시됩니다.

이 매개변수를 지정하지 않으면 메시지 라우트의 요약이 표시됩니다.

## 추가 정보 표시

라우트 추적 메시지는 큐 관리자 네트워크를 통해 라우팅되므로 사용자 작성 애플리케이션이 하나 이상의 추가 PCF 매개변수를 라우트 추적 메시지의 메시지 데이터나 활동 보고서의 메시지 데이터에 씌으로써 추가 정보를 기록할 수 있습니다. IBM WebSphere MQ 표시 라우트 애플리케이션이 추가 정보를 읽기 쉬운 양식으로 표시하게 하려면 67 페이지의 『추가 활동 정보』에 설명된 대로 추가 정보를 특정 형식으로 기록해야 합니다.

## WebSphere MQ 표시 라우트 애플리케이션 예

다음 예에서는 WebSphere MQ 표시 라우트 애플리케이션 사용 방법을 보여줍니다. 각 예마다 두 개의 큐 관리자(QM1 및 QM2)가 두 가지 채널(QM2.TO.QM1 및 QM1.TO.QM2)을 통해 서로 연결되어 있습니다.

### 예 1 - 활동 보고서 요청

대상 큐에 전달된 라우트 추적 메시지에서 활동 정보를 표시합니다.

이 예제에서는 WebSphere MQ 라우트 표시 애플리케이션이 큐 관리자 QM1에 연결되어 리모트 큐 관리자 QM2에 있는 대상 큐 TARGET.Q에 대해 라우트 추적 메시지를 생성하고 배달하는 데 사용됩니다. 라우트 추적 응답 메시지가 라우팅될 때 활동 보고서가 요청되도록 필요한 보고서 옵션이 지정됩니다. 대상 큐에 도달하면 라우트 추적 메시지가 제거됩니다. 활동 보고서를 사용하여 WebSphere MQ 라우트 표시 애플리케이션에 리턴되는 활동 정보는 순서대로 들어가고 표시됩니다.

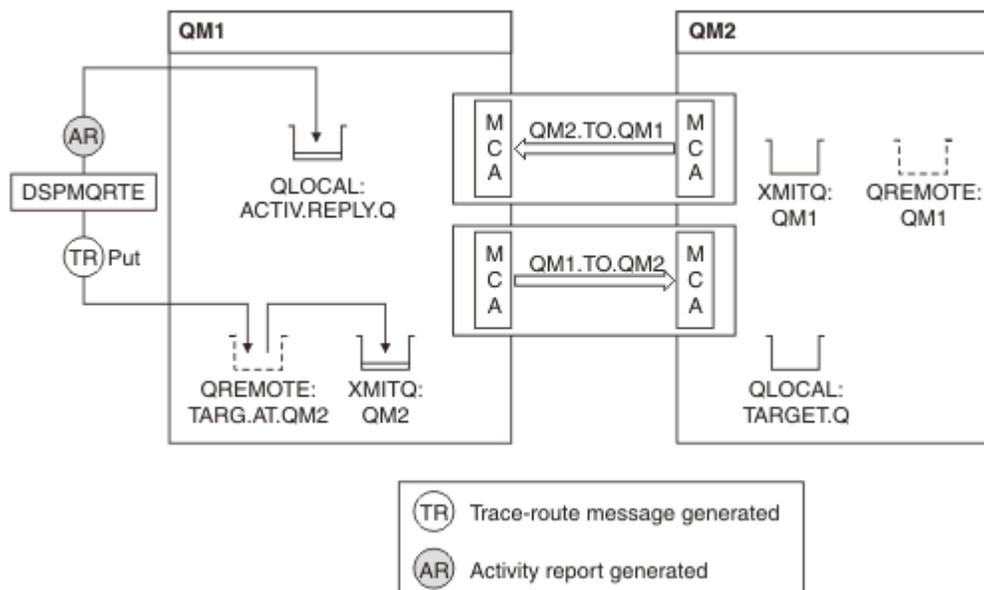


그림 9. 활동 보고서 요청, 다이어그램 1

- 각 큐 관리자(QM1 및 QM2)의 ACTIVREC 속성은 MSG로 설정됩니다.
- 다음 명령이 발행됩니다.

```
dspmqrite -m QM1 -q TARG.AT.QM2 -rq ACTIV.REPLY.Q
```

QM1은 WebSphere MQ 라우트 표시 애플리케이션이 연결되는 큐 관리자의 이름이며 TARG.AT.QM2는 대상 큐의 이름이며 ACTIV.REPLY.Q는 라우트 추적 메시지에 대한 모든 응답을 송신하도록 요청되는 큐의 이름입니다.

지정되지 않은 모든 옵션은 기본값이 가정되지만 특히, -f 옵션(TraceRoute PCF 그룹의 Deliver 매개변수를 존중하는 큐 관리자에만 라우트 추적 메시지가 전달됨), -d 옵션(라우트 추적 메시지가 도달하면 대상 큐에 메시지를 넣지 않음), -ro 옵션(MQRO\_ACTIVITY 및 MQRO\_DISCARD\_MSG 보고서 옵션이 지정됨), -t 옵션(중간 세부사항 레벨 활동이 기록됨)에 유의하십시오.

- DSPMQRTE는 라우트 추적 메시지를 생성하여 리모트 큐 TARG.AT.QM2에 넣습니다.
- DSPMQRTE는 그런 다음 큐 관리자 QM1의 ACTIVREC 속성 값을 봅니다. 값이 MSG이므로 DSPMQRTE는 활동 보고서를 생성하여 응답 큐 ACTIV.REPLY.Q에 넣습니다.

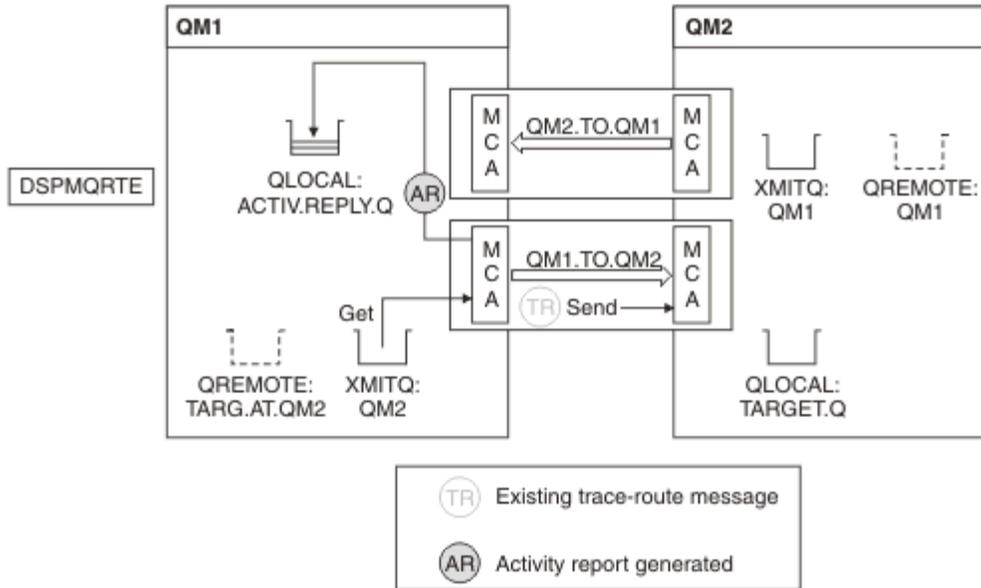


그림 10. 활동 보고서 요청, 다이어그램 2

- 송신 메시지 채널 에이전트(MCA)는 전송 큐에서 라우트 추적 메시지를 가져옵니다. 메시지가 라우트 추적 메시지이므로 MCA는 활동 정보를 기록하기 시작합니다.
- 큐 관리자(QM1)의 ACTIVREC 속성은 MSG이고 MQRO\_ACTIVITY 옵션이 메시지 디스크립터의 Report 필드에 지정되어 있으므로 MCA는 활동 보고서를 나중에 생성합니다. TraceRoute PCF 그룹의 RecordedActivities 매개변수 값은 1씩 증분됩니다.
- MCA는 TraceRoute PCF 그룹의 MaxActivities 값이 초과되지 않았는지 검사합니다.
- 메시지가 QM2에 전달되기 전에 MCA는 전달(단계 65 페이지의 『1』, 65 페이지의 『4』, 65 페이지의 『5』)에 설명된 알고리즘에 따르며 메시지를 송신하도록 선택합니다.
- 그런 다음 MCA는 활동 보고서를 생성하여 응답 큐(ACTIV.REPLY.Q)에 넣습니다.

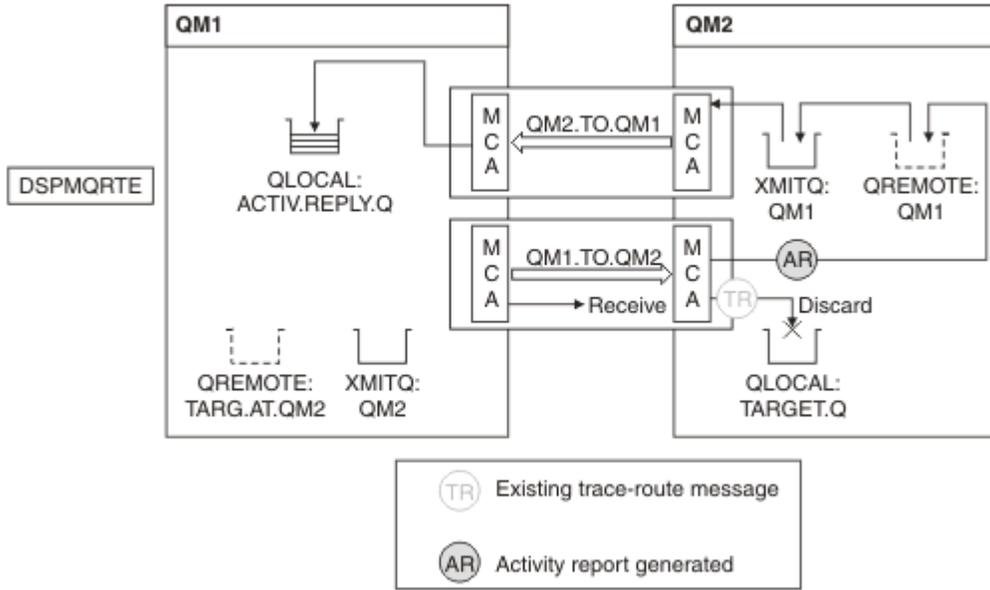


그림 11. 활동 보고서 요청, 다이어그램 3

- 수신 MCA는 채널에서 라우트 추적 메시지를 수신합니다. 메시지가 라우트 추적 메시지이므로 MCA는 활동에 대한 정보를 기록하기 시작합니다.
- 라우트 추적 메시지가 나온 큐 관리자가 5.3.1 이전 버전인 경우 MCA는 TraceRoute PCF의 DiscontinuityCount 매개변수를 1씩 증분시킵니다. 이 예는 여기에 해당하지 않습니다.
- 큐 관리자(QM2)의 ACTIVREC 속성은 MSG이고 MQRO\_ACTIVITY 옵션이 지정되어 있으므로 MCA는 활동 보고서를 생성합니다. RecordedActivities 매개변수 값은 1씩 증분됩니다.
- 대상 큐가 로컬 큐이므로 TraceRoute PCF 그룹의 Deliver 매개변수 값에 따라 피드백 MQFB\_NOT\_DELIVERED와 함께 메시지가 제거됩니다.
- 그런 다음 MCA는 최종 활동 보고서를 생성하여 응답 큐에 넣습니다. 이는 큐 관리자 QM1과 연관된 전송 큐로 분석되며 큐 관리자 QM1(ACTIV.REPLY.Q)에 활동 보고서가 리턴됩니다.

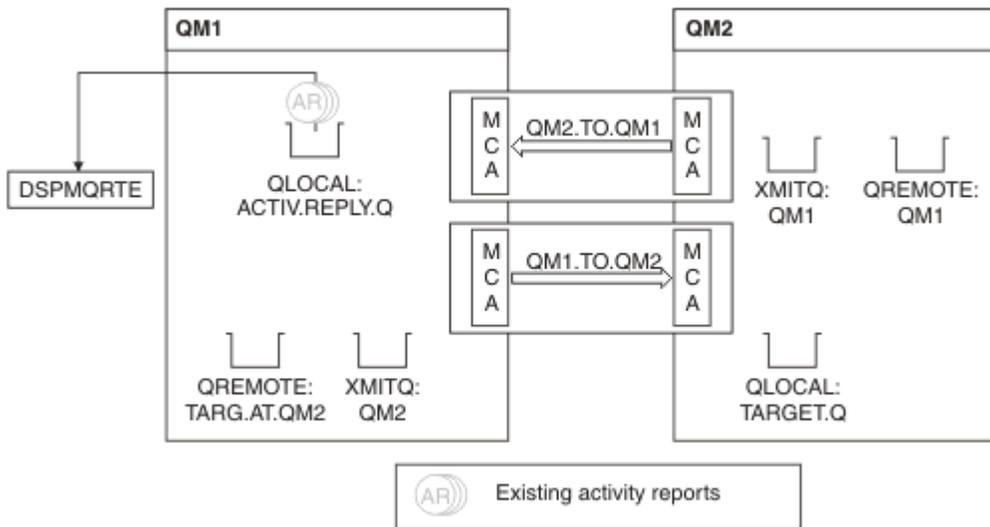


그림 12. 활동 보고서 요청, 다이어그램 4

- 그 동안, DSPMQRTE는 활동 보고서를 대기하며 응답 큐(ACTIV.REPLY.Q)에서 계속하여 MQGET을 수행했습니다. DSPMQRTE가 시작할 때 -w가 지정되지 않았으므로 최대 120초(라우트 추적 메시지의 만기 시간보다 60초 더) 동안 대기합니다.

- DSPMQRTE는 응답 큐에서 3개의 활동 보고서를 가져옵니다.
- 활동 보고서는 각 활동에 대한 TraceRoute PCF 그룹의 RecordedActivities, UnrecordedActivities, DiscontinuityCount 매개변수를 사용하여 순서화됩니다. 이 예에서 0이 아닌 유일한 값은 RecordedActivities 이므로 이 값이 실제로 사용되는 유일한 매개변수입니다.
- 제거 조작이 표시되면 프로그램이 즉시 종료합니다. 최종 조작이 제거였어도 피드백이 MQFB\_NOT\_DELIVERED이므로 넣기가 발생한 것처럼 처리됩니다.

표시되는 출력은 다음과 같습니다.

```
AMQ8653: DSPMQRTE command started with options '-m QM1 -q TARG.AT.QM2
-rq ACTIV.REPLY.Q'.
AMQ8659: DSPMQRTE command successfully put a message on queue 'QM2',
queue manager 'QM1'.
AMQ8674: DSPMQRTE command is now waiting for information to display.
AMQ8666: Queue 'QM2' on queue manager 'QM1'.
AMQ8666: Queue 'TARGET.Q' on queue manager 'QM2'.
AMQ8652: DSPMQRTE command has finished.
```

## 예 2 - 라우트 추적 응답 메시지 요청

라우트 추적 메시지를 생성하여 대상 큐에 전달합니다.

이 예제에서는 WebSphere MQ 라우트 표시 애플리케이션이 큐 관리자 QM1에 연결되어 리모트 큐 관리자 QM2에 있는 대상 큐 TARGET.Q에 대해 라우트 추적 메시지를 생성하고 배달하는 데 사용됩니다. 라우트 추적 메시지에 활동 정보가 누적되도록 필요한 옵션이 지정됩니다. 대상 큐에 도달하면 라우트 추적 응답 메시지가 요청되고 라우트 추적 메시지는 제거됩니다.

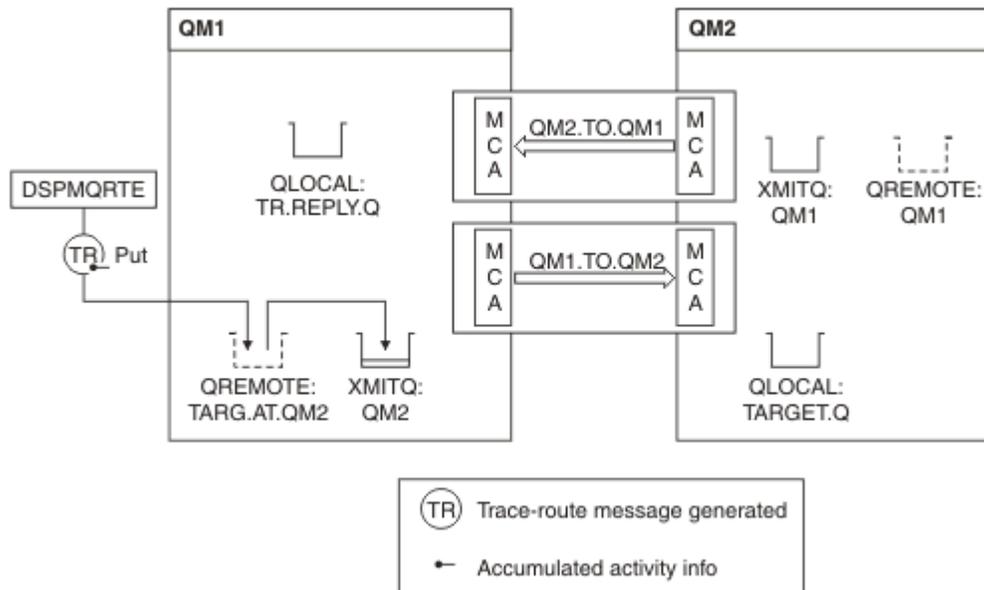


그림 13. 라우트 추적 응답 메시지 요청, 다이어그램 1

- 각 큐 관리자(QM1 및 QM2)의 ROUTEREC 속성은 MSG로 설정됩니다.
- 다음 명령이 발행됩니다.

```
dspmqrite -m QM1 -q TARG.AT.QM2 -rq TR.REPLY.Q -ac -ar -ro discard
```

QM1은 WebSphere MQ 라우트 표시 애플리케이션이 연결되는 큐 관리자의 이름이며 TARG.AT.QM2는 대상 큐의 이름이며 ACTIV.REPLY.Q는 라우트 추적 메시지에 대한 모든 응답을 송신하도록 요청되는 큐의 이름입니다. -ac 옵션은 라우트 추적 메시지에 활동 정보가 누적되도록 지정하고, -ar 옵션은 -rq 옵션에 지정된 응답 대상 큐(즉, TR.REPLY.Q)로 누적된 모든 활동을 송신하도록 지정합니다. -ro 옵션은 보고서 옵션 MQRO\_DISCARD\_MSG가 설정되도록 지정하며 이는 이 예에서 활동 보고서가 생성되지 않음을 의미합니다.

- DSPMQRTE는 대상 라우트에 메시지를 넣기 전에 라우트 추적 메시지에 활동 정보를 누적시킵니다. 이를 위해서는 큐 관리자 속성 ROUTEREC가 DISABLED이면 안됩니다.

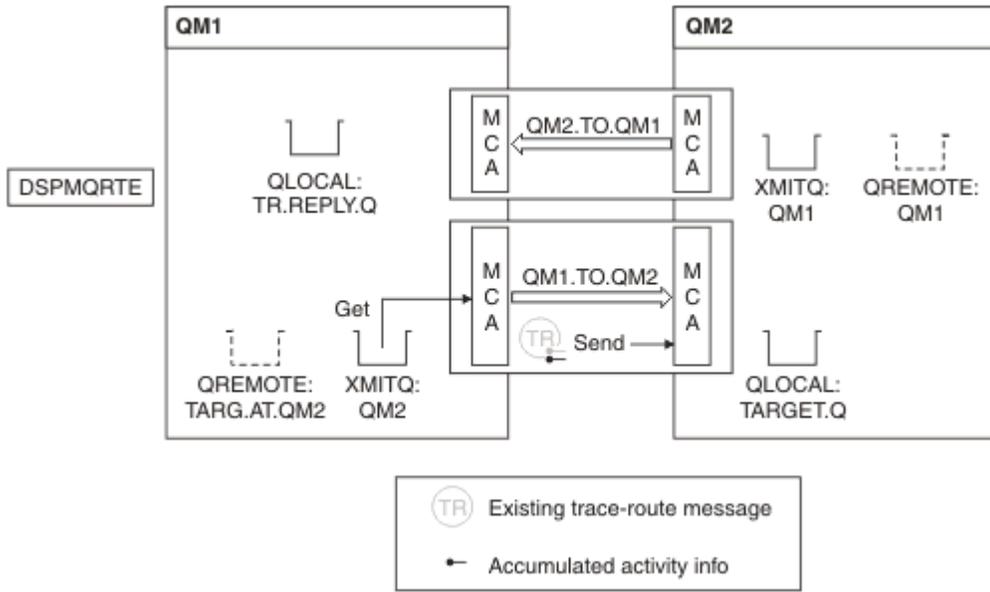


그림 14. 라우트 추적 응답 메시지 요청, 다이어그램 2

- 메시지가 라우트 추적 메시지이므로 송신 MCA는 활동에 대한 정보를 기록하기 시작합니다.
- QM1의 큐 관리자 속성 ROUTEREC가 DISABLED가 아니므로 MCA는 큐 관리자 QM2로 메시지가 전달되기 전에 활동 정보를 메시지 내에 누적시킵니다.

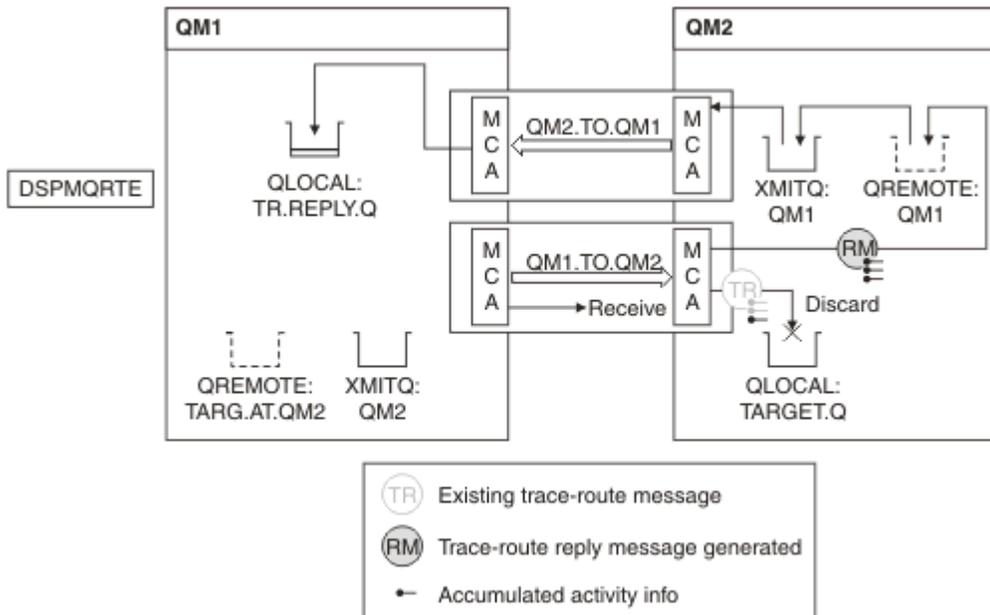


그림 15. 라우트 추적 응답 메시지 요청, 다이어그램 3

- 메시지가 라우트 추적 메시지이므로 수신 MCA는 활동에 대한 정보를 기록하기 시작합니다.
- QM2의 큐 관리자 속성 ROUTEREC가 DISABLED가 아니므로 MCA는 정보를 메시지 내에 누적시킵니다.
- 대상 큐가 로컬 큐이므로 TraceRoute PCF 그룹의 Deliver 매개변수 값에 따라 피드백 MQFB\_NOT\_DELIVERED와 함께 메시지가 제거됩니다.

- 이는 메시지에 대해 발생할 마지막 활동이며 QM1의 큐 관리자 속성 ROUTEREC가 DISABLED가 아니므로 MCA가 Accumulate 값에 따라 라우트 추적 응답 메시지를 생성합니다. ROUTEREC의 값이 MSG이므로 응답 메시지를 응답 큐에 넣습니다. 응답 메시지에는 라우트 추적 메시지의 누적된 모든 활동 정보가 포함됩니다.

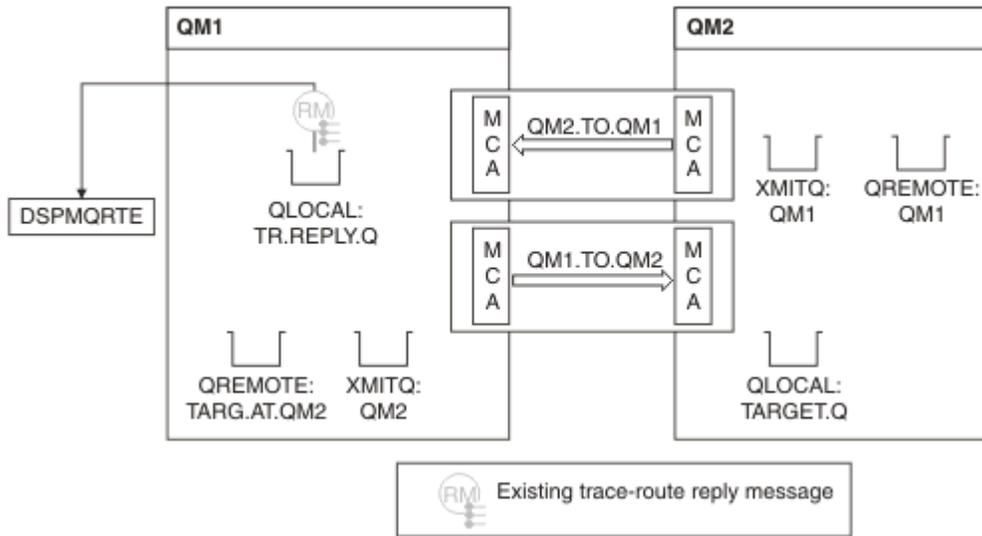


그림 16. 라우트 추적 응답 메시지 요청, 다이어그램 4

- 그 동안 DSPMQRTE는 라우트 추적 응답 메시지가 응답 큐에 리턴할 때까지 대기합니다. 메시지가 리턴되면 DSPMQRTE는 포함된 각 활동을 구문 분석하고 인쇄합니다. 최종 조작은 제거 조작입니다. 인쇄가 완료되면 DSPMQRTE가 종료합니다.

표시되는 출력은 다음과 같습니다.

```
AMQ8653: DSPMQRTE command started with options '-m QM1 -q TARG.AT.QM2 -rq
TR.REPLY.Q'.
AMQ8659: DSPMQRTE command successfully put a message on queue 'QM2', queue
manager 'QM1'.
AMQ8674: DSPMQRTE command is now waiting for information to display.
AMQ8666: Queue 'QM2' on queue manager 'QM1'.
AMQ8666: Queue 'TARGET.Q' on queue manager 'QM2'.
AMQ8652: DSPMQRTE command has finished.
```

### 예 3 - 시스템 큐에 활동 보고서 전달

활동 보고서가 응답 대상 큐 외의 기타 큐에 배달되는 경우를 발견하고 WebSphere MQ 라우트 표시 애플리케이션을 사용하여 기타 큐로부터 활동 보고서를 읽습니다.

이 예는 QM2의 ACTIVREC 큐 관리자 속성 값이 이제 QUEUE로 설정된 점을 제외하면 77 페이지의 『예 1 - 활동 보고서 요청』과 동일합니다. 이를 적용하기 위해서는 채널 QM1.TO.QM2를 재시작해야 합니다.

이 예는 응답 대상 큐가 아닌 다른 큐에 활동 보고서가 전달되는 시기를 감지할 방법을 설명합니다. 일단 발견되면 다른 큐로부터 활동 보고서를 읽기 위해 WebSphere MQ 라우트 표시 애플리케이션이 사용됩니다.

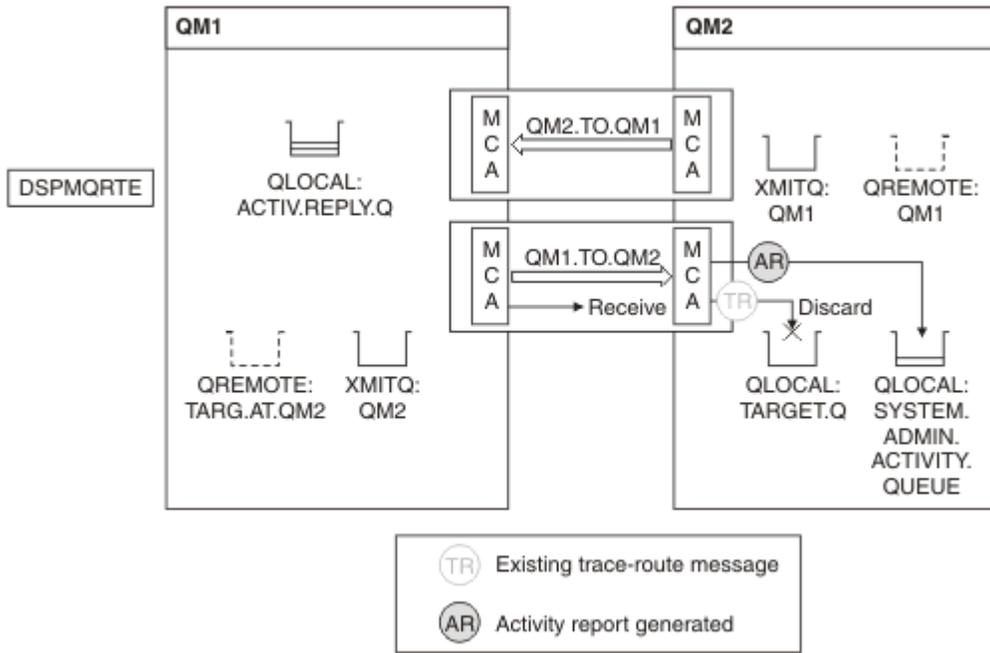


그림 17. 시스템 큐에 활동 보고서 전달, 다이어그램 1

- 메시지가 라우트 추적 메시지가므로 수신 MCA는 활동에 대한 정보를 기록하기 시작합니다.
- QM2의 ACTIVREC 큐 관리자 속성 값이 이제 QUEUE이므로 MCA는 활동 보고서를 생성하지만 응답 큐 (ACTIV.REPLY.Q)가 아닌 시스템 큐(SYSTEM.ADMIN.ACTIVITY.QUEUE)에 보고서를 넣습니다.

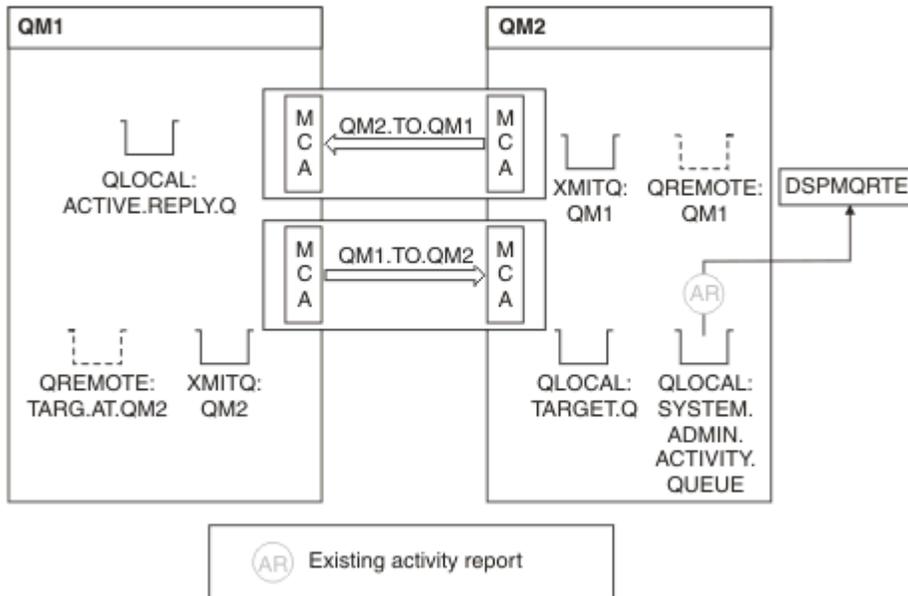


그림 18. 시스템 큐에 활동 보고서 전달, 다이어그램 2

- 그 동안 DSPMQRTE는 활동 보고서가 ACTIV.REPLY.Q에 도달하기를 대기했습니다. 두 개의 보고서만 도달합니다. DSPMQRTE는 라우트가 아직 완료되지 않은 것 같아서 계속하여 120초 동안 대기합니다.

표시되는 출력은 다음과 같습니다.

```
AMQ8653: DSPMQRTE command started with options '-m QM1 -q TARG.AT.QM2 -rq
ACTIV.REPLY.Q -v outline identifiers'.
AMQ8659: DSPMQRTE command successfully put a message on queue 'QM2', queue
manager 'QM1'.
AMQ8674: DSPMQRTE command is now waiting for information to display.
```

```
-----
Activity:
  ApplName: 'cann\output\bin\dspmqrte.exe'
```

```
Operation:
  OperationType: Put
```

```
Message:
```

```
MQMD:
  MsgId: X'414D51204C41524745512020202020A3C9154220001502'
  CorrelId: X'414D51204C41524745512020202020A3C9154220001503'
  QMgrName: 'QM1'
  QName: 'TARG.AT.QM2'
  ResolvedQName: 'QM2'
  RemoteQName: 'TARGET.Q'
  RemoteQMgrName: 'QM2'
```

```
-----
Activity:
  ApplName: 'cann\output\bin\runmqchl.EXE'
```

```
Operation:
  OperationType: Get
```

```
Message:
```

```
MQMD:
  MsgId: X'414D51204C41524745512020202020A3C9154220001505'
  CorrelId: X'414D51204C41524745512020202020A3C9154220001502'
```

```
EmbeddedMQMD:
  MsgId: X'414D51204C41524745512020202020A3C9154220001502'
  CorrelId: X'414D51204C41524745512020202020A3C9154220001503'
  QMgrName: 'QM1'
  QName: 'QM2'
  ResolvedQName: 'QM2'
```

```
Operation:
  OperationType: Send
```

```
Message:
```

```
MQMD:
  MsgId: X'414D51204C41524745512020202020A3C9154220001502'
  CorrelId: X'414D51204C41524745512020202020A3C9154220001503'
  QMgrName: 'QM1'
  RemoteQMgrName: 'QM2'
  ChannelName: 'QM1.TO.QM2'
  ChannelType: Sender
  XmitQName: 'QM2'
```

```
-----
AMQ8652: DSPMQRTE command has finished.
```

- DSPMQRTE가 관찰한 마지막 조작이 송신이었으므로 채널은 실행 중입니다. 이제 우리는 큐 관리자 QM2(RemoteQMgrName에 식별된)에서 더 많은 활동 보고서를 수신하지 못한 이유를 파악해야 합니다.
- 시스템 큐에 활동 정보가 있는지 여부를 확인하기 위해 QM2에서 DSPMQRTE를 시작해서 더 많은 활동 보고서를 시도 및 수집하십시오. 다음 명령을 사용하여 DSPMQRTE를 시작하십시오.

```
dspmqrte -m QM2 -q SYSTEM.ADMIN.ACTIVITY.QUEUE
-i 414D51204C41524745512020202020A3C9154220001502 -v outline
```

여기서, 414D51204C41524745512020202020A3C9154220001502는 넣은 라우트 추적 메시지의 MsgId입니다.

- DSPMQRTE는 그런 다음 일련의 MQGET를 다시 수행하고 지정된 ID의 라우트 추적 메시지에 관련된 시스템 활동 큐의 응답을 대기합니다.
- DSPMQRTE는 표시된 활동 보고서를 하나 더 가져옵니다. DSPMQRTE는 이전 활동 보고서가 누락되었는지 판별하고 이를 알리는 메시지를 표시합니다. 하지만 우리는 이 라우트 파트에 대해 이미 알고 있습니다.

표시되는 출력은 다음과 같습니다.

```

AMQ8653: DSPMQRTE command started with options '-m QM2
        -q SYSTEM.ADMIN.ACTIVITY.QUEUE
        -i 414D51204C41524745512020202020A3C915420001502 -v outline'.
AMQ8674: DSPMQRTE command is now waiting for information to display.
-----
Activity:
  Activity information unavailable.
-----
Activity:
  ApplName: 'cann\output\bin\AMQRMPPA.EXE'

  Operation:
    OperationType: Receive
    QMgrName: 'QM2'
    RemoteQMGrName: 'QM1'
    ChannelName: 'QM1.TO.QM2'
    ChannelType: Receiver

  Operation:
    OperationType: Discard
    QMgrName: 'QM2'
    QName: 'TARGET.Q'
    Feedback: NotDelivered
-----
AMQ8652: DSPMQRTE command has finished.

```

- 이 활동 보고서는 라우트 정보가 이제 완료되었다고 표시합니다. 문제가 발생하지 않았습니다.
- 단지 라우트 정보가 사용 불가능하거나 DSPMQRTE가 모든 라우트를 표시할 수 없다는 이유에서, 이 사실이 메시지가 전달되지 않았음을 의미하는 것은 아닙니다. 예를 들어, 다른 큐 관리자의 큐 관리자 속성이 차이가 있거나 응답 큐가 응답을 다시 가져오도록 정의되지 않았을 수 있습니다.

#### 예 4 - 채널 문제점 진단

라우트 추적 메시지가 대상 큐에 도달하지 않는 문제점을 진단합니다.

이 예에서 WebSphere MQ 라우트 표시 애플리케이션이 큐 관리자 QM1에 연결되고 라우트 추적 메시지를 생성한 다음 리모트 큐 관리자 QM2에 있는 대상 큐 TARGET.Q에 이를 배달하려고 시도합니다. 이 예에서는 라우트 추적 메시지가 대상 큐에 도달하지 않습니다. 사용 가능한 활동 보고서를 사용하여 문제점을 진단합니다.

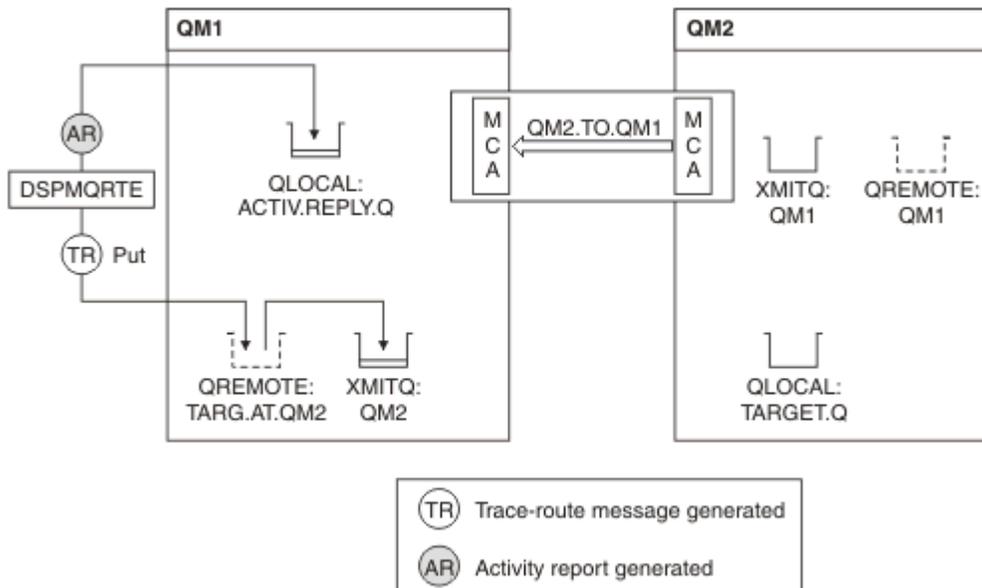


그림 19. 채널 문제점 진단

- 이 예에서는 채널 QM1.TO.QM2가 실행 중이지 않습니다.

- DSPMQRTE는 라우트 추적 메시지를 대상 큐에 넣고(예 1에서처럼) 활동 보고서를 생성합니다.
- 전송 큐(QM2)에서 메시지를 가져올 MCA가 없으므로 DSPMQRTE가 응답 큐에서 다시 가져오는 유일한 활동 보고서는 이 보고서 뿐입니다. 이번에는 라우트가 완료되지 않았다는 사실이 문제점을 표시합니다. 관리자는 ResolvedQName에 있는 전송 큐를 사용하여 전송 큐가 서비스를 제공하지 않은 원인을 조사할 수 있습니다. 표시되는 출력은 다음과 같습니다.

```
AMQ8653: DSPMQRTE command started with options '-m QM1 -q TARG.AT.QM2
-rq ACTIV.REPLY.Q -v outline'.
AMQ8659: DSPMQRTE command successfully put a message on queue 'QM2',
queue manager 'QM1'.
AMQ8674: DSPMQRTE command is now waiting for information to display.
```

```
-----
Activity:
  ApplName: 'cann\output\bin\dspmqrte.exe'
```

```
Operation:
  OperationType: Put
  QMgrName: 'QM1'
  QName: 'TARG.AT.QM2'
  ResolvedQName: 'QM2'
  RemoteQName: 'TARGET.Q'
  RemoteQMGrName: 'QM2'
```

```
-----
AMQ8652: DSPMQRTE command has finished.
```

## 활동 보고서 참조

이 페이지를 사용하여 활동 보고서 메시지 형식을 개괄합니다. 활동 보고서 메시지 데이터는 활동을 설명하는 매개변수를 포함합니다.

### 활동 보고서 형식

활동 보고서는 메시지 디스크립터 및 메시지 데이터를 포함한 표준 IBM WebSphere MQ 보고 메시지입니다. 활동 보고서는 메시지가 큐 관리자 네트워크를 통해 라우팅되었으므로 메시지에 관하여 활동을 수행한 애플리케이션이 생성하는 PCF 메시지입니다.

활동 보고서는 다음 정보를 포함합니다.

#### 메시지 디스크립터

MQMD 구조

#### 메시지 데이터

다음으로 구성되어 있습니다.

- 임베드된 PCF 헤더(MQEPH).
- 활동 보고서 메시지 데이터.

활동 보고서 메시지 데이터는 Activity PCF 그룹 및 라우트 추적 메시지에 대해 생성된 경우 *TraceRoute* PCF 그룹으로 이루어집니다.

[87 페이지의 표 19](#)는 특정 조건 하에서만 리턴되는 매개변수를 포함하여 이러한 보고서의 구조를 보여줍니다.

표 19. 활동 보고서 형식		
MQMD 구조	임베드된 PCF 헤더 MQEPH 구조	활동 보고서 메시지 데이터
구조 ID 구조 버전 보고서 옵션 메시지 유형 만기 시간 Feedback Encoding 코드화 문자 세트 ID 메시지 형식 Priority 지속 메시지 ID 상관 ID 백아웃 수 응답 대상 큐 응답 대상 큐 관리자 사용자 ID 계정 토큰 애플리케이션 ID 데이터 애플리케이션 유형 애플리케이션 이름 넣은 날짜 넣은 시간 애플리케이션 원본 데이터 그룹 ID 메시지 순서 번호 오프셋 메시지 플래그 원래 길이	구조 ID 구조 버전 구조 길이 Encoding 코드화 문자 세트 ID 메시지 형식 플래그 PCF 헤더(MQCFH) 구조 유형 구조 길이 구조 버전 명령 ID 메시지 순서 번호 제어 옵션 완료 코드 이유 코드 매개변수 수	활동 활동 애플리케이션 이름 활동 애플리케이션 유형 활동 설명 Operation 조작 유형 조작 날짜 조작 시간 메시지 메시지 길이 MQMD <sup>8</sup> EmbeddedMQMD 큐 관리자 이름 큐 공유 그룹 이름 큐 이름 <sup>1</sup> <sup>2 3</sup> <sup>7</sup> 해석된 큐 이름 <sup>1</sup> <sup>3</sup> <sup>7</sup> 리모트 큐 이름 <sup>3</sup> <sup>7</sup> 리모트 큐 관리자 이름 <sup>2</sup> <sup>3 4</sup> <sup>5</sup> <sup>7</sup> 구독 레벨 <sup>2</sup> Subscription ID <sup>9</sup> 피드백 <sup>2</sup> <sup>10</sup> 채널 이름 <sup>4</sup> <sup>5</sup> 채널 유형 <sup>4</sup> <sup>5</sup> 전송 큐 이름 <sup>5</sup> TraceRoute <sup>6</sup> Detail 기록된 활동 기록되지 않은 활동 중단 수 최대 활동 누적 전달

**참고사항:**

1. 가져오기 및 찾아보기 조작에 대해 리턴됩니다.
2. 제거 조작에 대해 리턴됩니다.
3. 넣기, 응답 넣기, 보고서 넣기 조작에 대해 리턴됩니다.
4. 수신 조작에 대해 리턴됩니다.

5. 송신 조작에 대해 리턴됩니다.
6. 라우트 추적 메시지에 대해 리턴됩니다.
7. 발행 활동 내에 포함된 토픽에 넣기 조작에 대해 리턴되지 않습니다.
8. 제외된 발행 조작에 대해 리턴되지 않습니다. 발행 및 제거된 발행 조작의 경우 매개변수의 서브세트를 포함하여 리턴됩니다.
9. 발행, 제거된 발행, 제외된 발행 조작에 대해 리턴됩니다.
10. 제거된 발행 및 제외된 발행 조작에 대해 리턴됩니다.

## 활동 보고서 MQMD(메시지 디스크립터)

이 페이지를 사용하여 활동 보고서의 MQMD 구조에 따라 포함된 값을 봅니다.

### **StrucId**

구조 ID:

데이터 유형

MQCHAR4

가치

MQMD\_STRUC\_ID.

### **Version**

구조 버전 번호

데이터 유형

MQLONG

값

원래 메시지 디스크립터에서 복사됩니다. 가능한 값은 다음과 같습니다.

#### **MQMD\_VERSION\_1**

모든 환경에서 지원되는 버전 1 메시지 디스크립터 구조

#### **MQMD\_VERSION\_2**

Version-2 메시지 디스크립터 구조, AIX®, HP-UX, z/OS, IBM i, Solaris, Linux, 윈도우 및 이 시스템에 연결된 모든 WebSphere MQ MQI 클라이언트에서 지원됩니다.

### **Report**

추가 보고 메시지에 대한 옵션

데이터 유형

MQLONG

가치

MQRO\_PASS\_DISCARD\_AND\_EXPIRY 또는 MQRO\_DISCARD\_MSG가 원래 메시지 디스크립터의 Report 필드에 지정된 경우:

#### **MQRO\_DISCARD**

보고서를 목적지 큐에 전달할 수 없는 경우 보고서가 제거됩니다.

그렇지 않으면 다음을 수행하십시오.

#### **MQRO\_NONE**

보고서가 필요하지 않습니다.

### **MsgType**

메시지의 유형을 표시함

데이터 유형

MQLONG

가치

MQMT\_REPORT

### **Expiry**

보고 메시지 수명

**데이터 유형**  
MQLONG

**가치**

원래 메시지 디스크립터의 *Report* 필드가 MQRO\_PASS\_DISCARD\_AND\_EXPIRY로 지정된 경우 원래 메시지의 남은 만기 시간이 사용됩니다.

그렇지 않으면 다음을 수행하십시오.

**MQEI\_UNLIMITED**

보고서의 만기 시간이 없습니다.

**Feedback**

설명: 피드백 또는 이유 코드.

데이터 유형: MQLONG.

가치: **MQFB\_ACTIVITY**  
활동 보고서.

**Encoding**

설명: 보고 메시지 데이터의 숫자 인코딩.

데이터 유형: MQLONG.

가치: MQENC\_NATIVE.

**CodedCharSetId**

설명: 보고 메시지 데이터의 문자 세트 ID.

데이터 유형: MQLONG.

가치: 해당하는 값으로 설정됩니다.

**Format**

설명: 보고 메시지 데이터의 형식 이름.

데이터 유형: MQCHAR8.

가치: **MQFMT\_EMBEDDED\_PCF**  
임베드된 PCF 메시지.

**Priority**

설명: 보고 메시지 우선순위.

데이터 유형: MQLONG.

가치: 원래 메시지 디스크립터에서 복사됩니다.

**Persistence**

설명: 보고 메시지 지속.

데이터 유형: MQLONG.

가치: 원래 메시지 디스크립터에서 복사됩니다.

**MsgId**

설명: 메시지 ID.

데이터 유형: MQBYTE24.

값: 원래 메시지 디스크립터의 *Report* 필드가 MQRO\_PASS\_MSG\_ID로 지정된 경우 원래 메시지의 메시지 ID가 사용됩니다.  
그렇지 않은 경우에는 큐 관리자가 고유 값을 생성합니다.

### **CorrelId**

설명: 상관 ID.  
데이터 유형: MQBYTE24.  
가치: 원래 메시지 디스크립터의 *Report* 필드가 MQRO\_PASS\_CORREL\_ID로 지정된 경우 원래 메시지의 상관 ID가 사용됩니다.  
그렇지 않은 경우에는 원래 메시지에서 메시지 ID가 복사됩니다.

### **BackoutCount**

설명: 백아웃 카운터입니다.  
데이터 유형: MQLONG.  
가치: 0.

### **ReplyToQ**

설명: 응답 큐의 이름입니다.  
데이터 유형: MQCHAR48.  
값: 공백입니다.

### **ReplyToQMGr**

설명: 응답 큐 관리자의 이름입니다.  
데이터 유형: MQCHAR48.  
가치: 보고 메시지를 생성한 큐 관리자 이름.

### **UserIdentifier**

설명: 보고 메시지를 생성한 애플리케이션의 사용자 ID.  
데이터 유형: MQCHAR12.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

### **AccountingToken**

설명: 애플리케이션이 메시지의 결과로 완료된 작업에 청구할 수 있는 계정 토큰.  
데이터 유형: MQBYTE32.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

### **ApplIdentityData**

설명: ID와 관련된 애플리케이션 데이터.  
데이터 유형: MQCHAR32.  
값: 원래 메시지 디스크립터에서 복사됩니다.

### **PutApplType**

설명: 보고 메시지를 넣은 애플리케이션의 유형.

데이터 유형: MQLONG.  
가치: **MQAT\_QMGR**  
메시지를 생성한 큐 관리자.

#### **PutApplName**

설명: 보고 메시지를 넣은 애플리케이션의 이름.  
데이터 유형: MQCHAR28.  
가치: 큐 관리자 이름 또는 보고 메시지를 생성한 MCA 이름의 처음 28바이트.

#### **PutDate**

설명: 메시지를 넣은 날짜입니다.  
데이터 유형: MQCHAR8.  
가치: 큐 관리자가 생성한 그대로의 값입니다.

#### **PutTime**

설명: 메시지를 넣은 시간입니다.  
데이터 유형: MQCHAR8.  
가치: 큐 관리자가 생성한 그대로의 값입니다.

#### **ApplOriginData**

설명: 원본과 관련된 애플리케이션 데이터입니다.  
데이터 유형: MQCHAR4.  
가치: 공백입니다.

버전이 MQMD\_VERSION\_2이면 다음과 같은 추가 필드가 있습니다.

#### **GroupId**

설명: 실제 메시지가 속한 메시지 그룹 또는 논리 메시지를 식별합니다.  
데이터 유형: MQBYTE24.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

#### **MsgSeqNumber**

설명: 그룹 내의 논리 메시지의 순서 번호.  
데이터 유형: MQLONG.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

#### **Offset**

설명: 논리 메시지의 시작으로부터 실제 메시지의 데이터 오프셋입니다.  
데이터 유형: MQLONG.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

#### **MsgFlags**

설명: 메시지의 속성을 지정하거나 처리를 제어하는 메시지 플래그입니다.

데이터 유형: MQLONG.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

### **OriginalLength**

설명: 원래 메시지의 길이.  
데이터 유형: MQLONG.  
가치: 원래 메시지 디스크립터에서 복사됩니다.

## **활동 보고서 MQEPH(임베드된 PCF 헤더)**

이 페이지를 사용하여 활동 보고서의 MQEPH 구조에 따라 포함된 값을 봅니다.

MQEPH 구조는 활동 보고서의 메시지 데이터를 수반한 PCF 정보와 이를 따르는 애플리케이션 메시지 데이터 모두에 대한 설명을 포함합니다.

활동 보고서의 경우 MQEPH 구조가 다음 값을 포함합니다.

### **StrucId**

설명: 구조 ID.  
데이터 유형: MQCHAR4.  
가치: MQEPH\_STRUC\_ID.

### **Version**

설명: 구조 버전 번호입니다.  
데이터 유형: MQLONG.  
값: MQEPH\_VERSION\_1.

### **StrucLength**

설명: 구조 길이.  
데이터 유형: MQLONG.  
가치: 뒤따르는 PCF 매개변수 구조를 포함한 구조의 총 길이.

### **Encoding**

설명: 마지막 PCF 매개변수 구조 다음에 오는 메시지 데이터의 숫자 인코딩.  
데이터 유형: MQLONG.  
가치: 원래 애플리케이션 메시지 데이터의 데이터가 보고 메시지에 포함된 경우 원래 메시지 디스크립터의 *Encoding* 필드에서 값이 복사됩니다.  
그렇지 않은 경우에는 0입니다.

### **CodedCharSetId**

설명: 마지막 PCF 매개변수 구조 다음에 오는 메시지 데이터의 문자 세트 ID.  
데이터 유형: MQLONG.  
가치: 원래 애플리케이션 메시지 데이터의 데이터가 보고 메시지에 포함된 경우 원래 메시지 디스크립터의 *CodedCharSetId* 필드에서 값이 복사됩니다.  
그렇지 않은 경우에는 MQCCSI\_UNDEFINED입니다.

### Format

- 설명: 마지막 PCF 매개변수 구조 다음에 오는 메시지 데이터의 형식 이름.
- 데이터 유형: MQCHAR8.
- 가치: 원래 애플리케이션 메시지 데이터의 데이터가 보고 메시지에 포함된 경우 원래 메시지 디스크립터의 *Format* 필드에서 값이 복사됩니다.  
그렇지 않은 경우에는 MQFMT\_NONE입니다.

### Flags

- 설명: 구조의 속성을 지정하거나 처리를 제어하는 플래그.
- 데이터 유형: MQLONG.
- 가치: **MQEPH\_CCSID\_EMBEDDED**  
문자 데이터를 포함한 매개변수의 문자 세트가 각 구조의 *CodedCharSetId* 필드 내에 개별적으로 지정됨을 지정합니다.

### PCFHeader

- 설명: 프로그래밍 가능 명령 형식(PCF) 헤더
- 데이터 유형: MQCFH.
- 가치: [93 페이지의 『활동 보고서 MQCFH\(PCF 헤더\)』](#)의 내용을 참조하십시오.

## 활동 보고서 MQCFH(PCF 헤더)

이 페이지를 사용하여 활동 보고서의 MQCFH 구조에 따라 포함된 PCF 값을 봅니다.

활동 보고서의 경우 MQCFH 구조가 다음 값을 포함합니다.

### Type

- 설명: 보고서 메시지의 콘텐츠를 식별하는 구조 유형.
- 데이터 유형: MQLONG.
- 가치: **MQCFT\_REPORT**  
메시지가 보고서입니다.

### StrucLength

- 설명: 구조 길이.
- 데이터 유형: MQLONG.
- 가치: **MQCFH\_STRUC\_LENGTH**  
MQCFH 구조의 길이(바이트 단위).

### Version

- 설명: 구조 버전 번호입니다.
- 데이터 유형: MQLONG.
- 값: MQCFH\_VERSION\_3

### Command

- 설명: 명령 ID입니다. 메시지의 범주를 식별합니다.
- 데이터 유형: MQLONG.

값: **MQCMD\_ACTIVITY\_MSG**  
메시지 활동.

### **MsgSeqNumber**

설명: 메시지 순서 번호입니다. 관련 메시지 그룹 내의 메시지의 순서 번호입니다.  
데이터 유형: MQLONG.  
값: 1.

### **Control**

설명: 제어 옵션입니다.  
데이터 유형: MQLONG.  
값: MQCFC\_LAST.

### **CompCode**

설명: 완료 코드.  
데이터 유형: MQLONG.  
값: MQCC\_OK.

### **Reason**

설명: 완료 코드를 규정하는 이유 코드입니다.  
데이터 유형: MQLONG.  
값: MQRC\_NONE.

### **ParameterCount**

설명: 매개변수 구조 수입니다. MQCFH 구조 뒤에 나오는 매개변수 구조의 수입니다. 그룹 구조(MQCFGR) 및 포함된 해당 매개변수 구조는 하나의 구조로만 계산됩니다.  
데이터 유형: MQLONG.  
값: 1 이상.

## **활동 보고서 메시지 데이터**

이 페이지를 사용하여 활동 보고서 메시지의 *Activity* PCF 그룹에 포함된 매개변수를 봅니다. 일부 매개변수는 특정 조작이 수행되었을 때에만 리턴됩니다.

활동 보고서 메시지 데이터는 *Activity* PCF 그룹 및 라우트 추적 메시지에 대해 생성된 경우 *TraceRoute* PCF 그룹으로 이루어집니다. *Activity* PCF 그룹을 이 주제에서 자세히 설명합니다.

조작별 활동 보고서 메시지 데이터로 설명되는 일부 매개변수는 특정 조작이 수행되었을 때에만 리턴됩니다.

활동 보고서의 경우 활동 보고서 메시지 데이터에 다음 매개변수가 포함됩니다.

### **Activity**

설명: 활동을 설명하는 그룹화된 매개변수.  
ID: MQGACF\_ACTIVITY.  
데이터 유형: MQCFGR.  
PCF 그룹에 포함: 없음

PCF 그룹의 매개변수: *ActivityApplName*  
*ActivityApplType*  
*ActivityDescription*  
*Operation*  
*TraceRoute*

리턴됨: 항상.

### ***ActivityApplName***

설명: 활동을 수행한 애플리케이션의 이름.  
ID: MQCACF\_APPL\_NAME.  
데이터 유형: MQCFST.  
PCF 그룹에 포함: *Activity*.  
최대 길이: MQ\_APPL\_NAME\_LENGTH.  
리턴됨: 항상.

### ***ActivityApplType***

설명: 활동을 수행한 애플리케이션의 유형.  
ID: MQIA\_APPL\_TYPE.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: *Activity*.  
리턴됨: 항상.

### ***ActivityDescription***

설명: 애플리케이션이 수행한 활동에 대한 설명.  
ID: MQCACF\_ACTIVITY\_DESCRIPTION.  
데이터 유형: MQCFST.  
PCF 그룹에 포함: *Activity*.  
최대 길이: 64  
리턴됨: 항상.

### ***Operation***

설명: 활동의 조작을 설명하는 그룹화된 매개변수.  
ID: MQGACF\_OPERATION.  
데이터 유형: MQCFGR.  
PCF 그룹에 포함: *Activity*.

PCF 그룹의 매개변수:

- OperationType*
- OperationDate*
- OperationTime*
- Message*
- QMgrName*
- QSGName*

**참고:** 조작 유형에 따라 이 그룹의 추가 매개변수가 리턴됩니다. 이 추가 매개변수는 [조작별 활동 보고서 메시지 데이터](#)로 설명됩니다.

리턴됨: 활동의 조작별로 하나의 *Operation* PCF 그룹.

### ***OperationType***

설명: 수행한 조작의 유형.  
 ID: MQIACF\_OPERATION\_TYPE.  
 데이터 유형: MQCFIN.  
 PCF 그룹에 포함: *Operation*.  
 값: MQOPER\_\*.  
 리턴됨: 항상.

### ***OperationDate***

설명: 조작이 수행된 날짜.  
 ID: MQCACF\_OPERATION\_DATE.  
 데이터 유형: MQCFST.  
 PCF 그룹에 포함: *Operation*.  
 최대 길이: MQ\_DATE\_LENGTH.  
 리턴됨: 항상.

### ***OperationTime***

설명: 조작이 수행된 시간.  
 ID: MQCACF\_OPERATION\_TIME.  
 데이터 유형: MQCFST.  
 PCF 그룹에 포함: *Operation*.  
 최대 길이: MQ\_TIME\_LENGTH.  
 리턴됨: 항상.

### ***Message***

설명: 활동을 유발한 메시지를 설명하는 그룹화된 매개변수.  
 ID: MQGACF\_MESSAGE.  
 데이터 유형: MQCFGR.  
 PCF 그룹에 포함: *Operation*.  
 그룹의 매개변수: *MsgLength*  
*MQMD*  
*EmbeddedMQMD*

리턴됨: 항상, 제외된 발행 조작은 예외입니다.

### **MsgLength**

설명: 활동이 발생하기 전, 활동을 유발한 메시지의 길이.  
ID: MQIACF\_MSG\_LENGTH.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: Message.  
리턴됨: 항상.

### **MQMD**

설명: 활동을 유발한 메시지의 메시지 디스크립터에 관련된 그룹화된 매개변수.  
ID: MQGACF\_MQMD.  
데이터 유형: MQCFGR.  
PCF 그룹에 포함: Message.

그룹의 매개변수:

- StrucId*
- Version*
- Report*
- MsgType*
- Expiry*
- Feedback*
- Encoding*
- CodedCharSetId*
- Format*
- Priority*
- Persistence*
- MsgId*
- CorrelId*
- BackoutCount*
- ReplyToQ*
- ReplyToQMgr*
- UserIdentifier*
- AccountingToken*
- ApplIdentityData*
- PutApplType*
- PutApplName*
- PutDate*
- PutTime*
- ApplOriginData*
- GroupId*
- MsgSeqNumber*
- Offset*
- MsgFlags*
- OriginalLength*

리턴됨: 항상, 제외된 발행 조작은 예외입니다.

## **EmbeddedMQMD**

설명: 전송 큐의 메시지 내에 임베드된 메시지 디스크립터를 설명하는 그룹화된 매개변수.

ID: MQGACF\_EMBEDDED\_MQMD.

데이터 유형: MQCFGR.

PCF 그룹에 포함: *Message*.

그룹의 매개변수:

- StrucId*
- Version*
- Report*
- MsgType*
- Expiry*
- Feedback*
- Encoding*
- CodedCharSetId*
- Format*
- Priority*
- Persistence*
- MsgId*
- CorrelId*
- BackoutCount*
- ReplyToQ*
- ReplyToQMgr*
- UserIdentifier*
- AccountingToken*
- ApplIdentityData*
- PutApplType*
- PutApplName*
- PutDate*
- PutTime*
- ApplOriginData*
- GroupId*
- MsgSeqNumber*
- Offset*
- MsgFlags*
- OriginalLength*

리턴됨: 큐가 전송 큐로 분석되는 가져오기 조작의 경우.

## **StrucId**

설명: 구조 ID

ID: MQCACF\_STRUC\_ID.

데이터 유형: MQCFST.

PCF 그룹에 포함: *MQMD* 또는 *EmbeddedMQMD*.

최대 길이: 4.

리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

## **Version**

설명: 구조 버전 번호입니다.

ID: MQIACF\_VERSION.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **Report**

설명: 보고 메시지의 옵션입니다.  
ID: MQIACF\_REPORT.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **MsgType**

설명: 메시지 유형을 표시합니다.  
ID: MQIACF\_MSG\_TYPE.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **Expiry**

설명: 메시지 수명입니다.  
ID: MQIACF\_EXPIRY.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **Feedback**

설명: 피드백 또는 이유 코드.  
ID: MQIACF\_FEEDBACK.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **Encoding**

설명: 메시지 데이터의 숫자 인코딩입니다.  
ID: MQIACF\_ENCODING.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **CodedCharSetId**

설명:	메시지 데이터의 문자 세트 ID.
ID:	MQIA_CODED_CHAR_SET_ID.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **Format**

설명:	메시지 데이터의 형식 이름
ID:	MQCACH_FORMAT_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
최대 길이:	MQ_FORMAT_LENGTH.
리턴됨:	항상, 제외된 발행 조작은 예외입니다.

### **Priority**

설명:	메시지 우선순위입니다.
ID:	MQIACF_PRIORITY.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	항상, 제외된 발행 조작은 예외입니다.

### **Persistence**

설명:	메시지 지속성.
ID:	MQIACF_PERSISTENCE.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	항상, 제외된 발행 조작은 예외입니다.

### **MsgId**

설명:	메시지 ID.
ID:	MQBACF_MSG_ID.
데이터 유형:	MQCFBS.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
최대 길이:	MQ_MSG_ID_LENGTH.
리턴됨:	항상, 제외된 발행 조작은 예외입니다.

### **CorrelId**

설명:	상관 ID.
ID:	MQBACF_CORREL_ID.
데이터 유형:	MQCFBS.

PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
최대 길이: MQ\_CORREL\_ID\_LENGTH.  
리턴됨: 항상, 제외된 발행 조작은 예외입니다.

### **BackoutCount**

설명: 백아웃 카운터입니다.  
ID: MQIACF\_BACKOUT\_COUNT.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **ReplyToQ**

설명: 응답 큐의 이름입니다.  
ID: MQCACF\_REPLY\_TO\_QUEUE.  
데이터 유형: MQCFST.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
최대 길이: MQ\_Q\_NAME\_LENGTH.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **ReplyToQMgr**

설명: 응답 큐 관리자의 이름입니다.  
ID: MQCACF\_REPLY\_TO\_Q\_MGR.  
데이터 유형: MQCFST.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH.  
리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **UserIdentifier**

설명: 메시지를 생성한 애플리케이션의 사용자 ID.  
ID: MQCACF\_USER\_IDENTIFIER.  
데이터 유형: MQCFST.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
최대 길이: MQ\_USER\_ID\_LENGTH.  
리턴됨: 항상, 제외된 발행 조작은 예외입니다.

### **AccountingToken**

설명: 애플리케이션이 메시지의 결과로 완료된 작업에 청구할 수 있는 계정 토큰.  
ID: MQBACF\_ACCOUNTING\_TOKEN.  
데이터 유형: MQCFBS.  
PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.  
최대 길이: MQ\_ACCOUNTING\_TOKEN\_LENGTH.

리턴됨: 항상, 제외된 발행 조작은 예외입니다.

### ***ApplIdentityData***

설명: ID와 관련된 애플리케이션 데이터.

ID: MQCACF\_APPL\_IDENTITY\_DATA.

데이터 유형: MQCFST.

PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.

최대 길이: MQ\_APPL\_IDENTITY\_DATA\_LENGTH.

리턴됨: 항상, 제외된 발행 조작은 예외입니다.

### ***PutApplType***

설명: 메시지를 넣는 애플리케이션의 유형입니다.

ID: MQIA\_APPL\_TYPE.

데이터 유형: MQCFIN.

PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.

리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### ***PutApplName***

설명: 메시지를 넣는 애플리케이션의 이름입니다.

ID: MQCACF\_APPL\_NAME.

데이터 유형: MQCFST.

PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.

최대 길이: MQ\_APPL\_NAME\_LENGTH.

리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### ***PutDate***

설명: 메시지를 넣은 날짜입니다.

ID: MQCACF\_PUT\_DATE.

데이터 유형: MQCFST.

PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.

최대 길이: MQ\_PUT\_DATE\_LENGTH.

리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### ***PutTime***

설명: 메시지를 넣은 시간입니다.

ID: MQCACF\_PUT\_TIME.

데이터 유형: MQCFST.

PCF 그룹에 포함: MQMD 또는 *EmbeddedMQMD*.

최대 길이: MQ\_PUT\_TIME\_LENGTH.

리턴됨: 항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **ApplOriginData**

설명:	원본과 관련된 애플리케이션 데이터입니다.
ID:	MQCACF_APPL_ORIGIN_DATA.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
최대 길이:	MQ_APPL_ORIGIN_DATA_LENGTH.
리턴됨:	항상, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작은 예외입니다.

### **GroupId**

설명:	실제 메시지가 속한 메시지 그룹 또는 논리 메시지를 식별합니다.
ID:	MQBACF_GROUP_ID.
데이터 유형:	MQCFBS.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
최대 길이:	MQ_GROUP_ID_LENGTH.
리턴됨:	<i>Version</i> 이 MQMD_VERSION_2로 지정된 경우, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작의 경우 리턴되지 않습니다.

### **MsgSeqNumber**

설명:	그룹 내의 논리 메시지의 순서 번호.
ID:	MQIACH_MSG_SEQUENCE_NUMBER.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	<i>Version</i> 이 MQMD_VERSION_2로 지정된 경우, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작의 경우 리턴되지 않습니다.

### **Offset**

설명:	논리 메시지의 시작으로부터 실제 메시지의 데이터 오프셋입니다.
ID:	MQIACF_OFFSET.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	<i>Version</i> 이 MQMD_VERSION_2로 지정된 경우, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작의 경우 리턴되지 않습니다.

### **MsgFlags**

설명:	메시지의 속성을 지정하거나 처리를 제어하는 메시지 플래그입니다.
ID:	MQIACF_MSG_FLAGS.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	<i>Version</i> 이 MQMD_VERSION_2로 지정된 경우, 제외된 발행 조작과 MQMD에서 발행 및 제거된 발행 조작의 경우 리턴되지 않습니다.

### **OriginalLength**

설명:	원래 메시지의 길이.
ID:	MQIACF_ORIGINAL_LENGTH.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>MQMD</i> 또는 <i>EmbeddedMQMD</i> .
리턴됨:	<i>Version</i> 이 <i>MQMD_VERSION_2</i> 로 지정된 경우. 제외된 발행 조작과 <i>MQMD</i> 에서 발행 및 제거된 발행 조작의 경우 리턴되지 않습니다.

### **QMgrName**

설명:	활동이 수행된 큐 관리자의 이름.
ID:	MQCA_Q_MGR_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_Q_MGR_NAME_LENGTH
리턴됨:	항상.

### **QSGName**

설명:	활동이 수행된 큐 관리자가 속한 큐 공유 그룹의 이름.
ID:	MQCA_QSG_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_QSG_NAME_LENGTH
리턴됨:	If the activity was performed on a WebSphere MQ for z/OS queue manager.

### **TraceRoute**

설명:	라우트 추적 메시지의 속성을 지정하는 그룹화된 매개변수.
ID:	MQGACF_TRACE_ROUTE.
데이터 유형:	MQCFGR.
PCF 그룹에 포함:	<i>Activity</i> .
그룹의 매개변수:	<i>Detail</i> <i>RecordedActivities</i> <i>UnrecordedActivities</i> <i>DiscontinuityCount</i> <i>MaxActivities</i> <i>Accumulate</i> <i>Forward</i> <i>Deliver</i>
리턴됨:	라우트 추적 메시지에 관하여 활동이 수행된 경우.

*TraceRoute* PCF 그룹의 매개변수 값은 활동 보고서가 생성되었을 때 라우트 추적 메시지의 값입니다.

## 조작별 활동 보고서 메시지 데이터

이 페이지를 사용하여 *OperationType* 매개변수의 값에 따라 활동 보고서의 PCF 그룹 *Operation*에 리턴될 수 있는 추가 PCF 매개변수를 봅니다.

추가 매개변수는 다음 조작 유형에 따라 다릅니다.

### 가져오기/찾아보기(MQOPER\_GET/MQOPER\_BROWSE)

가져오기/찾아보기(MQOPER\_GET/MQOPER\_BROWSE) 조작 유형(큐의 메시지를 가져오거나 찾아봄)에 대해 PCF 그룹 *Operation*에 리턴되는 추가 활동 보고서 메시지 데이터 매개변수.

#### QName

설명:	열린 큐의 이름.
ID:	MQCA_Q_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_Q_NAME_LENGTH
리턴됨:	항상.

#### ResolvedQName

설명:	열린 큐를 분석한 이름.
ID:	MQCACF_RESOLVED_Q_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_Q_NAME_LENGTH
리턴됨:	항상.

### 제거(MQOPER\_DISCARD)

제거(MQOPER\_DISCARD) 조작 유형(메시지가 제거됨)에 대해 PCF 그룹 *Operation*에 리턴되는 추가 활동 보고서 메시지 데이터 매개변수.

#### Feedback

설명:	메시지를 제거하는 이유.
ID:	MQIACF_FEEDBACK.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>Operation</i> .
리턴됨:	항상.

#### QName

설명:	열린 큐의 이름.
ID:	MQCA_Q_NAME.
데이터 유형:	MQCFST.
최대 길이:	MQ_Q_NAME_LENGTH
PCF 그룹에 포함:	<i>Operation</i> .
리턴됨:	큐에 넣는 데 실패해서 메시지가 제거된 경우.

### **RemoteQMgrName**

설명:	메시지가 예정된 큐 관리자의 이름.
ID:	MQCA_REMOTE_Q_MGR_NAME.
데이터 유형:	MQCFST.
최대 길이:	MQ_Q_MGR_NAME_LENGTH
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	<i>Feedback</i> 의 값이 MQFB_NOT_FORWARDED인 경우.

### **발행/제거된 발행/제외된 발행(MQOPER\_PUBLISH/MQOPER\_DISCARDED\_PUBLISH/MQOPER\_EXCLUDED\_PUBLISH)**

발행/제거된 발행/제외된 발행(MQOPER\_PUBLISH/MQOPER\_DISCARDED\_PUBLISH/MQOPER\_EXCLUDED\_PUBLISH) 조작 유형(발행/구독 메시지가 전달, 제거 또는 제외됨)에 대해 PCF 그룹 *Operation*에 리턴되는 추가 활동 보고서 메시지 데이터 매개변수.

### **SubId**

설명:	구독 ID.
ID:	MQBACF_SUB_ID.
데이터 유형:	MQCFBS.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	항상.

### **SubLevel**

설명:	구독 레벨.
ID:	MQIACF_SUB_LEVEL.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	항상.

### **Feedback**

설명:	메시지를 제거하는 이유.
ID:	MQIACF_FEEDBACK.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	메시지가 구독자에게 전달되지 않았거나 구독자가 제외되었기 때문에 메시지가 전달되지 않아서 메시지가 제거된 경우.

발행 조작 MQOPER\_PUBLISH는 특정 구독자에게 전달된 메시지에 대한 정보를 제공합니다. 이 조작은 연관된 넣기 조작에 설명된 메시지에서 변경되었을 수 있는 향후 메시지의 요소를 설명합니다. 넣기 조작과 마찬가지로, 메시지 그룹 MQGACF\_MESSAGE 및 그룹 내 MQMD 그룹 MQGACF\_MQMD를 포함합니다. 그러나 이 MQMD 그룹에는 구독자가 대체할 수 있는 *Format, Priority, Persistence, MsgId, CorrelId, UserIdentifier, AccountingToken, ApplIdentityData* 필드만 포함됩니다.

구독자의 *SubId* 및 *SubLevel*은 조작 정보에 포함됩니다. *SubID*를 MQCMD\_INQUIRE\_SUBSCRIBER PCF 명령과 함께 사용하여 구독자에 대한 다른 모든 속성을 검색할 수 있습니다.

제거된 발행 조작 MQOPER\_DISCARDED\_PUBLISH는 메시지가 포인트-투-포인트 메시징으로 전달되지 않을 때 사용되는 제거 조작과 유사합니다. 로컬 목적지에 메시지를 전달하지 않도록 명시적으로 요청되었고 이 구독자가 로컬 목적지를 지정하는 경우 구독자에게 메시지가 전달되지 않습니다. 큐가 가득 찼을 때처럼 목적지 큐에 메시지가 도달하는 데 문제점이 있는 경우에도 메시지가 전달되지 않는 것으로 고려합니다.

제거된 발행 조작의 정보는 발행 조작에 대해 동일하며 메시지가 전달되지 않은 이유를 제공하는 *Feedback* 필드가 추가됩니다. 이 피드백 필드는 MQOPER\_DISCARD 조작에 공통된 MQFB\_\* 또는 MQRC\_\* 값을 포함합니다. 발행 제외와 대조적으로, 발행을 제거하는 이유는 넣기를 제거하는 이유와 동일합니다.

구독자가 구독 중인 토픽이 연관된 넣기 조작의 토픽과 일치하지만 토픽에 넣고 있는 메시지와 다른 선택 기준이 일치하지 않아서 구독자에게 메시지가 전달되지 않았기 때문에, 제외된 발행 조작 MQOPER\_EXCLUDED\_PUBLISH는 메시지 전달을 고려한 구독자에 대한 정보를 제공합니다. 제거된 발행 조작에서처럼, *Feedback* 필드는 이 구독이 제외된 원인에 대한 정보를 제공합니다. 하지만 제거된 발행 조작과 달리, 이 구독자에 대한 메시지가 생성되지 않아서 메시지 관련 정보가 제공되지 않습니다.

### 넣기/응답 넣기/보고서 넣기(MQOPER\_PUT/MQOPER\_PUT\_REPLY/MQOPER\_PUT\_REPORT)

넣기/응답 넣기/보고서 넣기(MQOPER\_PUT/MQOPER\_PUT\_REPLY/MQOPER\_PUT\_REPORT) 조작 유형(메시지, 응답 메시지 또는 보고서 메시지를 큐에 넣음)에 대해 PCF 그룹 *Operation*에 리턴되는 추가 활동 보고서 메시지 데이터 매개변수.

#### QName

설명:	열린 큐의 이름.
ID:	MQCA_Q_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_Q_NAME_LENGTH
리턴됨:	항상, 발행 활동 내에 포함된 토픽에 대한 넣기 조작인 경우에 리턴되지 않는 한 가지 예외가 있습니다.

#### ResolvedQName

설명:	열린 큐를 분석한 이름.
ID:	MQCACF_RESOLVED_Q_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_Q_NAME_LENGTH
리턴됨:	열린 큐를 분석할 수 있을 때, 발행 활동 내에 포함된 토픽에 대한 넣기 조작인 경우 리턴되지 않습니다.

#### RemoteQName

설명:	리모트 큐 관리자에 알려진 열린 큐의 이름.
ID:	MQCA_REMOTE_Q_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation</i> .
최대 길이:	MQ_Q_NAME_LENGTH
리턴됨:	열린 큐가 리모트 큐인 경우, 발행 활동 내에 포함된 토픽에 대한 넣기 조작인 경우 리턴되지 않습니다.

### **RemoteQMgrName**

설명:	리모트 큐가 정의된 리모트 큐 관리자의 이름.
ID:	MQCA_REMOTE_Q_MGR_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
최대 길이:	MQ_Q_MGR_NAME_LENGTH
리턴됨:	열린 큐가 리모트 큐인 경우. 발행 활동 내에 포함된 토픽에 대한 넣기 조작인 경우 리턴되지 않습니다.

### **TopicString**

설명:	메시지를 넣고 있는 전체 토픽 문자열.
ID:	MQCA_TOPIC_STRING.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	발행 활동 내에 포함된 토픽에 대한 넣기 조작인 경우.

### **Feedback**

설명:	데드-레터 큐에 메시지를 넣는 이유.
ID:	MQIACF_FEEDBACK.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	데드-레터 큐에 메시지를 넣은 경우.

### **수신(MQOPER\_RECEIVE)**

수신(MQOPER\_RECEIVE) 조작 유형(채널에 메시지가 수신됨)에 대해 PCF 그룹 *Operation*에 리턴되는 추가 활동 보고서 메시지 데이터 매개변수.

### **ChannelName**

설명:	메시지가 수신된 채널의 이름.
ID:	MQCACH_CHANNEL_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
최대 길이:	MQ_CHANNEL_NAME_LENGTH
리턴됨:	항상.

### **ChannelType**

설명:	메시지가 수신된 채널의 유형.
ID:	MQIACH_CHANNEL_TYPE.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	항상.

### **RemoteQMgrName**

설명:	메시지가 수신된 큐 관리자의 이름.
ID:	MQCA_REMOTE_Q_MGR_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
최대 길이:	MQ_Q_MGR_NAME_LENGTH
리턴됨:	항상.

### **송신(MQOPER\_SEND)**

송신(MQOPER\_SEND) 조작 유형(채널에서 메시지가 송신됨)에 대해 PCF 그룹 *Operation*에 리턴되는 추가 활동 보고서 메시지 데이터 매개변수.

### **ChannelName**

설명:	메시지가 송신된 채널의 이름.
ID:	MQCACH_CHANNEL_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
최대 길이:	MQ_CHANNEL_NAME_LENGTH.
리턴됨:	항상.

### **ChannelType**

설명:	메시지가 송신된 채널의 유형.
ID:	MQIACH_CHANNEL_TYPE.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>Operation.</i>
리턴됨:	항상.

### **XmitQName**

설명:	메시지를 검색한 전송 큐.
ID:	MQCACH_XMIT_Q_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
최대 길이:	MQ_Q_NAME_LENGTH.
리턴됨:	항상.

### **RemoteQMgrName**

설명:	메시지가 송신된 리모트 큐 관리자의 이름.
ID:	MQCA_REMOTE_Q_MGR_NAME.
데이터 유형:	MQCFST.
PCF 그룹에 포함:	<i>Operation.</i>
최대 길이:	MQ_Q_MGR_NAME_LENGTH
리턴됨:	항상.

## 라우트 추적 메시지 참조

이 페이지를 사용하여 라우트 추적 메시지 형식을 개괄합니다. 라우트 추적 메시지 데이터는 라우트 추적 메시지가 유발한 활동을 설명하는 매개변수를 포함합니다.

### 라우트 추적 메시지 형식

라우트 추적 메시지는 메시지 디스크립터 및 메시지 데이터를 포함하는 표준 WebSphere MQ 메시지입니다. 메시지 데이터는 큐 관리자 네트워크를 통해 라우팅되었기 때문에 라우트 추적 메시지에 대해 수행된 활동에 대한 정보를 포함합니다.

라우트 추적 메시지는 다음 정보를 포함합니다.

#### 메시지 디스크립터

MQFMT\_ADMIN 또는 MQFMT\_EMBEDDED\_PCF로 설정된 *Format* 필드를 갖는 MQMD 구조입니다.

#### 메시지 데이터

다음 중 하나로 이루어집니다.

- *Format*이 MQFMT\_ADMIN으로 설정된 경우 PCF 헤더(MQCFH)와 라우트 추적 메시지 데이터 또는
- *Format*이 MQFMT\_EMBEDDED\_PCF로 설정된 경우 임베디드 PCF 헤더 (MQEPH), 라우트 추적 메시지 데이터 및 추가 사용자 지정 메시지 데이터.

WebSphere MQ 라우트 표시 애플리케이션을 사용하여 라우트 추적 메시지를 생성하는 경우, *Format*이 MQFMT\_ADMIN으로 설정됩니다.

라우트 추적 메시지 데이터의 콘텐츠는 다음과 같이 *TraceRoute* PCF 그룹의 *Accumulate* 매개변수로 판별됩니다.

- *Accumulate*가 MQROUTE\_ACCUMULATE\_NONE으로 설정된 경우 라우트 추적 메시지 데이터에 *TraceRoute* PCF 그룹이 포함됩니다.
- *Accumulate*가 MQROUTE\_ACCUMULATE\_IN\_MSG 또는 MQROUTE\_ACCUMULATE\_AND\_REPLY로 설정된 경우에는 라우트 추적 메시지 데이터에 *TraceRoute* PCF 그룹과 0개 이상의 *Activity* PCF 그룹이 포함됩니다.

111 페이지의 표 20은 라우트 추적 메시지의 구조를 보여줍니다.

표 20. 라우트 추적 메시지 형식		
MQMD 구조	임베드된 PCF 헤더 MQEPH 구조	라우트 추적 메시지 데이터
구조 ID 구조 버전 보고서 옵션 메시지 유형 만기 시간 Feedback Encoding 코드화 문자 세트 ID 메시지 형식 Priority 지속 메시지 ID 상관 ID 백아웃 수 응답 대상 큐 응답 대상 큐 관리자 사용자 ID 계정 토큰 애플리케이션 ID 데이터 애플리케이션 유형 애플리케이션 이름 넣은 날짜 넣은 시간 애플리케이션 원본 데이터 그룹 ID 메시지 순서 번호 오프셋 메시지 플래그 원래 길이	구조 ID 구조 버전 구조 길이 Encoding 코드화 문자 세트 ID 메시지 형식 플래그 PCF 헤더(MQCFH) 구조 유형 구조 길이 구조 버전 명령 ID 메시지 순서 번호 제어 옵션 완료 코드 이유 코드 매개변수 수	TraceRoute Detail 기록된 활동 기록되지 않은 활동 중단 수 최대 활동 누적 전달

### 라우트 추적 메시지 MQMD(메시지 디스크립터)

이 페이지를 사용하여 라우트 추적 메시지의 MQMD 구조에 따라 포함된 값을 봅니다.

#### **StrucId**

설명: 구조 ID.  
 데이터 유형: MQCHAR4.  
 가치: MQMD\_STRUC\_ID.

#### **Version**

설명: 구조 버전 번호입니다.  
 데이터 유형: MQLONG.  
 값: **MQMD\_VERSION\_1.**

#### **Report**

설명: 보고 메시지의 옵션입니다.  
 데이터 유형: MQLONG.

가치: 요구사항에 따라 설정됩니다. 공통 보고서 옵션은 다음과 같습니다.

#### **MQRO\_DISCARD\_MSG**

메시지가 로컬 큐에 도달하면 제거됩니다.

#### **MQRO\_PASS\_DISCARD\_AND\_EXPIRY**

모든 응답(활동 보고서 또는 라우트 추적 응답 메시지)에 보고서 옵션 MQRO\_DISCARD\_MSG가 설정되고 남은 만기가 전달됩니다. 이를 통해 응답이 큐 관리자 네트워크에 무제한으로 남아 있지 않습니다.

### **MsgType**

설명: 메시지의 유형.

데이터 유형: MQLONG.

가치: TraceRoute 그룹의 *Accumulate* 매개변수가 MQROUTE\_ACCUMULATE\_AND\_REPLY로 지정되어 있으면 메시지 유형이 MQMT\_REQUEST입니다.

그렇지 않으면 다음을 수행하십시오.

#### **MQMT\_DATAGRAM.**

### **Expiry**

설명: 메시지 수명입니다.

데이터 유형: MQLONG.

가치: 요구사항에 따라 설정됩니다. 이 매개변수는 라우트 추적 메시지가 큐 관리자 네트워크에 무제한으로 남지 않도록 하는 데 사용할 수 있습니다.

### **Feedback**

설명: 피드백 또는 이유 코드.

데이터 유형: MQLONG.

가치: **MQFB\_NONE.**

### **Encoding**

설명: 메시지 데이터의 숫자 인코딩입니다.

데이터 유형: MQLONG.

가치: 해당하는 값으로 설정됩니다.

### **CodedCharSetId**

설명: 메시지 데이터의 문자 세트 ID.

데이터 유형: MQLONG.

가치: 해당하는 값으로 설정됩니다.

### **Format**

설명: 메시지 데이터의 형식 이름

데이터 유형: MQCHAR8.

가치: **MQFMT\_ADMIN**  
관리 메시지. *TraceRoute* PCF 그룹 다음에 사용자 데이터가 없습니다.

**MQFMT\_EMBEDDED\_PCF**  
임베드된 PCF 메시지. *TraceRoute* PCF 그룹 다음에 사용자 데이터가 있습니다.

### **Priority**

설명: 메시지 우선순위입니다.

데이터 유형: MQLONG.

가치: 요구사항에 따라 설정됩니다.

### **Persistence**

설명: 메시지 지속성.

데이터 유형: MQLONG.

가치: 요구사항에 따라 설정됩니다.

### **MsgId**

설명: 메시지 ID.

데이터 유형: MQBYTE24.

가치: 요구사항에 따라 설정됩니다.

### **CorrelId**

설명: 상관 ID.

데이터 유형: MQBYTE24.

가치: 요구사항에 따라 설정됩니다.

### **BackoutCount**

설명: 백아웃 카운터입니다.

데이터 유형: MQLONG.

가치: 0.

### **ReplyToQ**

설명: 응답 큐의 이름입니다.

데이터 유형: MQCHAR48.

값: 요구사항에 따라 설정됩니다.

*MsgType*이 MQMT\_REQUEST로 설정되거나 *Report*에 보고서 생성 옵션이 설정된 경우 이 매개변수가 공백이 아니어야 합니다.

### **ReplyToQMgr**

설명: 응답 큐 관리자의 이름입니다.

데이터 유형: MQCHAR48.

가치: 요구사항에 따라 설정됩니다.

### ***UserIdentifier***

설명: 메시지를 생성한 애플리케이션의 사용자 ID.  
데이터 유형: MQCHAR12.  
가치: 일반적으로 설정됩니다.

### ***AccountingToken***

설명: 애플리케이션이 메시지의 결과로 완료된 작업에 청구할 수 있는 계정 토큰.  
데이터 유형: MQBYTE32.  
가치: 일반적으로 설정됩니다.

### ***ApplIdentityData***

설명: ID와 관련된 애플리케이션 데이터.  
데이터 유형: MQCHAR32.  
값: 일반적으로 설정됩니다.

### ***PutApplType***

설명: 메시지를 넣는 애플리케이션의 유형입니다.  
데이터 유형: MQLONG.  
가치: 일반적으로 설정됩니다.

### ***PutApplName***

설명: 메시지를 넣는 애플리케이션의 이름입니다.  
데이터 유형: MQCHAR28.  
가치: 일반적으로 설정됩니다.

### ***PutDate***

설명: 메시지를 넣은 날짜입니다.  
데이터 유형: MQCHAR8.  
가치: 일반적으로 설정됩니다.

### ***PutTime***

설명: 메시지를 넣은 시간입니다.  
데이터 유형: MQCHAR8.  
가치: 일반적으로 설정됩니다.

### ***ApplOriginData***

설명: 원본과 관련된 애플리케이션 데이터입니다.  
데이터 유형: MQCHAR4.  
가치: 일반적으로 설정됩니다.

## **라우트 추적 메시지 MQEPH(임베드된 PCF 헤더)**

이 페이지를 사용하여 라우트 추적 메시지의 MQEPH 구조에 따라 포함된 값을 봅니다.

MQEPH 구조는 라우트 추적 메시지의 메시지 데이터를 수반한 PCF 정보와 이를 따르는 애플리케이션 메시지 데이터 모두에 대한 설명을 포함합니다. MQEPH 구조는 TraceRoute PCF 그룹 다음에 추가 사용자 메시지 데이터가 있는 경우에만 사용됩니다.

라우트 추적 메시지의 경우 MQEPH 구조가 다음 값을 포함합니다.

#### **StrucId**

설명: 구조 ID.  
데이터 유형: MQCHAR4.  
가치: MQEPH\_STRUC\_ID.

#### **Version**

설명: 구조 버전 번호입니다.  
데이터 유형: MQLONG.  
값: MQEPH\_VERSION\_1.

#### **StrucLength**

설명: 구조 길이.  
데이터 유형: MQLONG.  
가치: 뒤따르는 PCF 매개변수 구조를 포함한 구조의 총 길이.

#### **Encoding**

설명: 마지막 PCF 매개변수 구조 다음에 오는 메시지 데이터의 숫자 인코딩.  
데이터 유형: MQLONG.  
가치: 메시지 데이터의 인코딩.

#### **CodedCharSetId**

설명: 마지막 PCF 매개변수 구조 다음에 오는 메시지 데이터의 문자 세트 ID.  
데이터 유형: MQLONG.  
가치: 메시지 데이터의 문자 세트.

#### **Format**

설명: 마지막 PCF 매개변수 구조 다음에 오는 메시지 데이터의 형식 이름.  
데이터 유형: MQCHAR8.  
가치: 메시지 데이터의 형식 이름.

#### **Flags**

설명: 구조의 속성을 지정하거나 처리를 제어하는 플래그.  
데이터 유형: MQLONG.  
가치: **MQEPH\_NONE**

플래그가 지정되지 않습니다.

#### **MQEPH\_CCsid\_EMBEDDED**

문자 데이터를 포함한 매개변수의 문자 세트가 각 구조의 *CodedCharSetId* 필드 내에 개별적으로 지정됨을 지정합니다.

### **PCFHeader**

설명:	프로그래밍 가능 명령 형식(PCF) 헤더
데이터 유형:	MQCFH.
가치:	<a href="#">116 페이지의 『라우트 추적 메시지 MQCFH(PCF 헤더)』의 내용을 참조하십시오.</a>

### **라우트 추적 메시지 MQCFH(PCF 헤더)**

이 페이지를 사용하여 라우트 추적 메시지의 MQCFH 구조에 따라 포함된 PCF 값을 봅니다.

라우트 추적 메시지의 경우 MQCFH 구조가 다음 값을 포함합니다.

#### **Type**

설명:	메시지의 콘텐츠를 식별하는 구조 유형입니다.
데이터 유형:	MQLONG.
가치:	<b>MQCFT_TRACE_ROUTE</b> 메시지가 라우트 추적 메시지입니다.

#### **StrucLength**

설명:	구조 길이.
데이터 유형:	MQLONG.
가치:	<b>MQCFH_STRUC_LENGTH</b> MQCFH 구조의 길이(바이트 단위).

#### **Version**

설명:	구조 버전 번호입니다.
데이터 유형:	MQLONG.
값:	MQCFH_VERSION_3

#### **Command**

설명:	명령 ID입니다. 메시지의 범주를 식별합니다.
데이터 유형:	MQLONG.
값:	<b>MQCMD_TRACE_ROUTE</b> 라우트 추적 메시지.

#### **MsgSeqNumber**

설명:	메시지 순서 번호입니다. 관련 메시지 그룹 내의 메시지의 순서 번호입니다.
데이터 유형:	MQLONG.
값:	1.

#### **Control**

설명:	제어 옵션입니다.
데이터 유형:	MQLONG.
값:	MQCFC_LAST.

### **CompCode**

설명: 완료 코드.  
데이터 유형: MQLONG.  
값: MQCC\_OK.

### **Reason**

설명: 완료 코드를 규정하는 이유 코드입니다.  
데이터 유형: MQLONG.  
값: MQRC\_NONE.

### **ParameterCount**

설명: 매개변수 구조 수입니다. MQCFH 구조 뒤에 나오는 매개변수 구조의 수입니다. 그룹 구조(MQCFGR) 및 포함된 해당 매개변수 구조는 하나의 구조로만 계산됩니다.  
데이터 유형: MQLONG.  
값: 1 이상.

## **라우트 추적 메시지 데이터**

이 페이지를 사용하여 라우트 추적 메시지 데이터의 *TraceRoute* PCF 그룹을 구성하는 매개변수를 봅니다.

라우트 추적 메시지 데이터의 콘텐츠는 *TraceRoute* PCF 그룹의 *Accumulate* 매개변수에 따라 다릅니다. 라우트 추적 메시지 데이터는 *TraceRoute* PCF 그룹과 0개 이상의 *Activity* PCF 그룹으로 구성됩니다. *TraceRoute* PCF 그룹을 이 주제에서 자세히 설명합니다. *Activity* PCF 그룹의 세부사항은 관련 정보를 참조하십시오.

라우트 추적 메시지 데이터는 다음 매개변수를 포함합니다.

### **TraceRoute**

설명: 라우트 추적 메시지의 속성을 지정하는 그룹화된 매개변수. 라우트 추적 메시지의 경우 일부 매개변수를 변경하여 처리 방식을 제어할 수 있습니다.  
ID: MQGACF\_TRACE\_ROUTE.  
데이터 유형: MQCFGR.  
PCF 그룹에 포함: 없음  
그룹의 매개변수: *Detail*  
*RecordedActivities*  
*UnrecordedActivities*  
*DiscontinuityCount*  
*MaxActivities*  
*Accumulate*  
*Forward*  
*Deliver*

### **Detail**

설명: 활동에 대해 기록할 세부사항 레벨.  
ID: MQIACF\_ROUTE\_DETAIL.  
데이터 유형: MQCFIN.  
PCF 그룹에 포함: *TraceRoute*.

값:

**MQRROUTE\_DETAIL\_LOW**

사용자 작성 애플리케이션이 수행한 활동이 기록됩니다.

**MQRROUTE\_DETAIL\_MEDIUM**

MQRROUTE\_DETAIL\_LOW에 지정된 활동이 기록됩니다. 또한 MCA가 수행한 활동이 기록됩니다.

**MQRROUTE\_DETAIL\_HIGH**

MQRROUTE\_DETAIL\_LOW 및 MQRROUTE\_DETAIL\_MEDIUM에 지정된 활동이 기록됩니다. MCA는 이 세부사항 레벨에서 어떤 추가 활동 정보도 기록하지 않습니다. 이 옵션은 추가 활동 정보를 기록할 사용자 작성 애플리케이션에만 사용 가능합니다.

**RecordedActivities**

설명: 정보가 기록된 라우트 추적 메시지가 유발한 활동의 수.

ID: MQIACF\_RECORDED\_ACTIVITIES.

데이터 유형: MQCFIN.

PCF 그룹에 포함: *TraceRoute*.

**UnrecordedActivities**

설명: 정보가 기록되지 않은 라우트 추적 메시지가 유발한 활동의 수.

ID: MQIACF\_UNRECORDED\_ACTIVITIES.

데이터 유형: MQCFIN.

PCF 그룹에 포함: *TraceRoute*.

**DiscontinuityCount**

설명: 라우트 추적 메시지를 지원하지 않는 큐 관리자에서 라우트 추적 메시지가 수신된 횟수.

ID: MQIACF\_DISCONTINUITY\_COUNT.

데이터 유형: MQCFIN.

PCF 그룹에 포함: *TraceRoute*.

**MaxActivities**

설명: 처리가 중지되기 전 라우트 추적 메시지에 포함시킬 수 있는 활동의 최대 수입니다.

ID: MQIACF\_MAX\_ACTIVITIES.

데이터 유형: MQCFIN.

PCF 그룹에 포함: *TraceRoute*.

가치: 양수  
활동의 최대 수입니다.

**MQRROUTE\_UNLIMITED\_ACTIVITIES**

무제한 수의 활동입니다.

**Accumulate**

설명: 활동 정보가 라우트 추적 메시지 내에 누적되는지 여부 및 라우트 추적 메시지를 제거하거나 비전송 큐에 넣기 전에 누적된 활동 정보를 포함한 응답 메시지가 생성되는지 여부를 지정합니다.

ID: MQIACF\_ROUTE\_ACCUMULATION.

데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>TraceRoute</i> .
가치:	<p><b>MQROUTE_ACCUMULATE_NONE</b> 라우트 추적 메시지의 메시지 데이터에 활동 정보가 누적되지 않습니다.</p> <p><b>MQROUTE_ACCUMULATE_IN_MSG</b> 라우트 추적 메시지의 메시지 데이터에 활동 정보가 누적됩니다.</p> <p><b>MQROUTE_ACCUMULATE_AND_REPLY</b> 활동 정보가 라우트 추적 메시지의 메시지 데이터에 누적되고 라우트 추적 응답 메시지가 생성됩니다.</p>

### **Forward**

설명:	라우트 추적 메시지를 전달할 수 있는 큐 관리자를 지정합니다. 리모트 큐 관리자에 메시지를 전달할지 여부를 판별할 때 큐 관리자는 <u>전달</u> 에 설명된 알고리즘을 사용합니다.
ID:	MQIACF_ROUTE_FORWARDING.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>TraceRoute</i> .
가치:	<p><b>MQROUTE_FORWARD_IF_SUPPORTED</b> 라우트 추적 메시지는 <i>TraceRoute</i> 그룹의 <i>Deliver</i> 매개변수의 값을 존중하는 큐 관리자로만 전달됩니다.</p> <p><b>MQROUTE_FORWARD_ALL</b> 라우트 추적 메시지가 <i>Deliver</i> 매개변수의 값이 존중되는지 여부와 상관없이 모든 큐 관리자로 전달됩니다.</p>

### **Deliver**

설명:	라우트 추적 메시지가 목적지 큐에 정상적으로 도달하는 경우에 수행할 조치를 지정합니다.
ID:	MQIACF_ROUTE_DELIVERY.
데이터 유형:	MQCFIN.
PCF 그룹에 포함:	<i>TraceRoute</i> .
가치:	<p><b>MQROUTE_DELIVER_YES</b> 라우트 추적 메시지가 도달하면 대상 큐에 메시지를 넣습니다. 대상 큐에서 파괴적 가져오기를 수행하는 애플리케이션은 라우트 추적 메시지를 수신할 수 있습니다.</p> <p><b>MQROUTE_DELIVER_NO</b> 라우트 추적 메시지가 도달하면 메시지를 제거합니다.</p>

## **라우트 추적 응답 메시지 참조**

이 페이지를 사용하여 라우트 추적 응답 메시지 형식을 개괄합니다. 라우트 추적 응답 메시지 데이터는 데이터가 생성된 라우트 추적 메시지의 라우트 추적 메시지 데이터 복제입니다.

### **라우트 추적 응답 메시지 형식**

라우트 추적 응답 메시지는 메시지 디스크립터 및 메시지 데이터를 포함하는 표준 WebSphere MQ 메시지입니다. 메시지 데이터는 큐 관리자 네트워크를 통해 라우팅되었기 때문에 라우트 추적 메시지에 대해 수행된 활동에 대한 정보를 포함합니다.

라우트 추적 응답 메시지는 다음 정보를 포함합니다.

## 메시지 디스크립터

MQMD 구조

## 메시지 데이터

PCF 헤더(MQCFH) 및 라우트 추적 메시지 데이터

라우트 추적 응답 메시지 데이터는 하나 이상의 Activity PCF 그룹으로 구성됩니다.

라우트 추적 메시지가 대상 큐에 도달하면 라우트 추적 메시지의 활동 정보 사본을 포함한 라우트 추적 응답 메시지가 생성될 수 있습니다. 라우트 추적 응답 메시지는 응답 대상 큐나 시스템 큐로 전달됩니다.

120 페이지의 표 21은 특정 조건 하에서만 리턴되는 매개변수를 포함하여, 라우트 추적 응답 메시지의 구조를 보여줍니다.

표 21. 라우트 추적 응답 메시지 형식		
MQMD 구조	PCF 헤더 MQCFH 구조	라우트 추적 응답 메시지 데이터
구조 ID 구조 버전 보고서 옵션 메시지 유형 만기 시간 Feedback Encoding 코드화 문자 세트 ID 메시지 형식 Priority 지속 메시지 ID 상관 ID 백아웃 수 응답 대상 큐 응답 대상 큐 관리자 사용자 ID 계정 토큰 애플리케이션 ID 데이터 애플리케이션 유형 애플리케이션 이름 넣은 날짜 넣은 시간 애플리케이션 원본 데이터 그룹 ID 메시지 순서 번호 오프셋 메시지 플래그 원래 길이	PCF 헤더(MQCFH) 구조 유형 구조 길이 구조 버전 명령 ID 메시지 순서 번호 제어 옵션 완료 코드 이유 코드 매개변수 수	활동 활동 애플리케이션 이름 활동 애플리케이션 유형 활동 설명 Operation 조작 유형 조작 날짜 조작 시간 메시지 메시지 길이 MQMD EmbeddedMQMD 큐 관리자 이름 큐 공유 그룹 이름 큐 이름 <sup>1 2 3</sup> 분석된 큐 이름 <sup>1 3</sup> 리모트 큐 이름 <sup>3</sup> 리모트 큐 관리자 이름 <sup>2 3 4 5</sup> 피드백 <sup>2</sup> 채널 이름 <sup>4 5</sup> 채널 유형 <sup>4 5</sup> 전송 큐 이름 <sup>5</sup> TraceRoute Detail 기록된 활동 기록되지 않은 활동 중단 수 최대 활동 누적 전달
<b>참고:</b> 1. 가져오기 및 찾아보기 조작에 대해 리턴됩니다. 2. 제거 조작에 대해 리턴됩니다. 3. 넣기, 응답 넣기, 보고서 넣기 조작에 대해 리턴됩니다. 4. 수신 조작에 대해 리턴됩니다. 5. 송신 조작에 대해 리턴됩니다.		

## 라우트 추적 응답 메시지 MQMD(메시지 디스크립터)

이 페이지를 사용하여 라우트 추적 응답 메시지의 MQMD 구조에 따라 포함된 값을 봅니다.

라우트 추적 응답 메시지의 경우 MQMD 구조는 [활동 보고서 메시지 디스크립터](#)에 설명된 매개변수를 포함합니다. 라우트 추적 응답 메시지 디스크립터의 일부 매개변수 값은 다음과 같이 활동 보고서 메시지 디스크립터와는 차이가 있습니다.

### **MsgType**

설명:	메시지의 유형.
데이터 유형:	MQLONG.
가치:	<b>MQMT_REPLY</b>

### **Feedback**

설명:	피드백 또는 이유 코드.
데이터 유형:	MQLONG.
가치:	<b>MQFB_NONE</b>

### **Encoding**

설명:	메시지 데이터의 숫자 인코딩입니다.
데이터 유형:	MQLONG.
가치:	라우트 추적 메시지 디스크립터에서 복사됩니다.

### **CodedCharSetId**

설명:	메시지 데이터의 문자 세트 ID.
데이터 유형:	MQLONG.
가치:	라우트 추적 메시지 디스크립터에서 복사됩니다.

### **Format**

설명:	메시지 데이터의 형식 이름
데이터 유형:	MQCHAR8.
가치:	<b>MQFMT_ADMIN</b> 관리 메시지.

## 라우트 추적 응답 메시지 MQCFH(PCF 헤더)

이 페이지를 사용하여 라우트 추적 응답 메시지의 MQCFH 구조에 따라 포함된 PCF 값을 봅니다.

라우트 추적 응답 메시지의 PCF 헤더(MQCFH)는 라우트 추적 메시지의 경우와 동일합니다.

## 라우트 추적 응답 메시지 데이터

라우트 추적 응답 메시지 데이터는 데이터가 생성된 라우트 추적 메시지의 라우트 추적 메시지 데이터 복제입니다.

라우트 추적 응답 메시지 데이터는 하나 이상의 *Activity* 그룹을 포함합니다. 매개변수는 [94 페이지의 『활동 보고서 메시지 데이터』](#)에 설명되어 있습니다.

## 회계 및 통계 메시지

큐 관리자는 IBM WebSphere MQ 애플리케이션이 수행하는 MQI 조작에 대한 정보를 기록하거나 IBM WebSphere MQ 시스템에서 발생하는 활동에 대한 정보를 기록하기 위해 계정 및 통계 메시지를 생성합니다.

### 회계 메시지

회계 메시지는 IBM WebSphere MQ 애플리케이션으로 수행된 MQI 조작에 관한 정보를 기록하는 데 사용됩니다. [122 페이지의 『회계 메시지』](#)의 내용을 참조하십시오.

### 통계 메시지

통계 메시지는 IBM WebSphere MQ 시스템에서 발생하는 활동에 대한 정보를 기록하는 데 사용됩니다 ([125 페이지의 『통계 메시지』](#) 참조). 통계 메시지에 기록된 일부 활동은 내부 큐 관리자 조작과 관련됩니다.

회계 및 통계 메시지는 두 개의 시스템 큐 중 하나에 전달됩니다. 사용자 애플리케이션은 이 시스템 큐에서 메시지를 검색하여 다음과 같은 다양한 용도로 기록된 정보를 사용할 수 있습니다.

- 애플리케이션 자원 사용을 설명합니다.
- 애플리케이션 활동을 기록합니다.
- 용량 계획.
- 큐 관리자 네트워크에서 문제점을 감지합니다.
- 큐 관리자 네트워크에서 문제점의 원인을 판별하는 데 도움을 줍니다.
- 큐 관리자 네트워크의 효율성을 향상시킵니다.
- 큐 관리자 네트워크의 실행에 익숙해집니다.
- 큐 관리자 네트워크가 올바르게 실행 중인지 확인합니다.

## 회계 메시지

계정 메시지는 WebSphere MQ 애플리케이션에 의해 수행되는 MQI 조작에 대한 정보를 기록합니다. 회계 메시지는 많은 PCF 구조를 포함한 PCF 메시지입니다.

애플리케이션이 큐 관리자로부터 연결이 끊어지면 회계 메시지가 생성되어 시스템 회계 큐 (SYSTEM.ADMIN.ACCOUNTING.QUEUE)로 전달됩니다. 장기 실행 중인 WebSphere MQ 애플리케이션의 경우, 다음과 같이 중간 계정 메시지가 생성됩니다.

- 연결이 설정된 이후의 시간이 구성된 간격을 초과할 때.
- 마지막 중간 회계 메시지 이후의 시간이 구성된 간격을 초과할 때.

회계 메시지는 다음 범주에 있습니다.

### MQI 회계 메시지

MQI 회계 메시지는 큐 관리자에 대한 연결을 사용하여 수행된 MQI 호출에 관련한 정보를 포함합니다.

### 큐 회계 메시지

큐 회계 메시지는 큐 관리자에 대한 연결을 사용하여 수행된 MQI 호출 수에 관련한 큐별로 그룹화된 정보를 포함합니다.

각 큐 회계 메시지는 최대 100개의 레코드를 포함할 수 있으며, 모든 레코드가 특정 큐에 대해 애플리케이션으로 수행된 활동에 관련되어 있습니다.

회계 메시지는 로컬 큐에 대해서만 기록됩니다. 애플리케이션이 알리어스 큐에 대해 MQI 호출을 수행하면 기본 큐에 대해 회계 데이터가 기록되고 리모트 큐의 경우에는 전송 큐에 대해 회계 데이터가 기록됩니다.

### 관련 참조

[138 페이지의 『MQI 회계 메시지 데이터』](#)

이 페이지를 사용하여 MQI 회계 메시지의 구조를 봅니다.

[148 페이지의 『큐 회계 메시지 데이터』](#)

이 페이지를 사용하여 큐 회계 메시지의 구조를 봅니다.

## 회계 메시지 형식

회계 메시지는 메시지 디스크립터와 메시지 데이터로 이루어진 PCF 필드 세트 구성됩니다.

### 메시지 디스크립터

- 회계 메시지 MQMD(메시지 디스크립터)

### 회계 메시지 데이터

- 회계 메시지 MQCFH(PCF 헤더)
- 항상 리턴되는 회계 메시지 데이터
- 사용 가능한 경우에 리턴되는 회계 메시지 데이터

회계 메시지 MQCFH(PCF 헤더)는 회계 데이터가 기록된 간격 및 애플리케이션에 대한 정보를 포함합니다.

회계 메시지 데이터는 회계 정보를 저장하는 PCF 매개변수로 구성됩니다. 회계 메시지의 콘텐츠는 다음과 같은 메시지 범주에 따라 다릅니다.

### MQI 회계 메시지

MQI 회계 메시지 데이터는 많은 PCF 매개변수로 구성됩니다(PCF 그룹은 없음).

### 큐 회계 메시지

큐 회계 메시지 데이터는 많은 PCF 매개변수로 구성되며 범위는 1 ~ 100 *QAccountingData* PCF 그룹입니다.

회계 데이터를 수집한 모든 큐에 *QAccountingData* PCF 그룹이 하나씩 있습니다. 애플리케이션이 100개를 초과한 큐에 액세스하면 여러 회계 메시지가 생성됩니다. 각 메시지마다 MQCFH(PCF 헤더)의 *SeqNumber* 가 그에 맞춰 업데이트되고, 순서의 마지막 메시지에는 MQCFH의 *Control* 매개변수가 MQCFC\_LAST로 지정됩니다.

## 회계 정보 콜렉션

큐와 큐 관리자 속성을 사용하여 회계 정보 콜렉션을 제어합니다. MQCONNX 옵션을 사용하여 연결 레벨에서 콜렉션을 제어할 수도 있습니다.

### MQI 회계 정보

큐 관리자 속성 ACCTMQI를 사용하여 MQI 회계 정보 콜렉션을 제어합니다.

이 속성의 값을 변경하려면 MQSC 명령 ALTER QMGR를 사용하고 ACCTMQI 매개변수를 지정하십시오. 회계 메시지는 회계가 사용된 후에 시작한 연결에 대해서만 생성됩니다. ACCTMQI 매개변수의 가능한 값은 다음과 같습니다.

#### 켜짐(ON)

모든 큐 관리자 연결에 대한 MQI 회계 정보를 수집합니다.

#### 꺼짐(OFF)

MQI 회계 정보를 수집하지 않습니다. 이 값은 기본값입니다.

예를 들어, MQI 회계 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR ACCTMQI(ON)
```

### 큐 회계 정보

큐 속성 ACCTQ와 큐 관리자 속성 ACCTQ를 사용하여 큐 회계 정보의 콜렉션을 제어합니다.

큐 속성의 값을 변경하려면 MQSC 명령 ALTER QLOCAL 를 사용하고 ACCTQ 매개변수를 지정하십시오. 회계 메시지는 회계가 사용된 후에 시작한 연결에 대해서만 생성됩니다. 큐 속성 ACCTQ의 가능한 값은 다음과 같습니다.

#### 켜짐(ON)

이 큐에 대한 큐 회계 정보는 큐를 여는 모든 큐 관리자 연결에 대해 수집됩니다.

#### 꺼짐(OFF)

이 큐에 대한 큐 회계 정보를 수집하지 않습니다.

## 큐 관리자

이 큐에 대한 큐 회계 정보 콜렉션은 큐 관리자 속성 ACCTQ의 값에 따라 제어됩니다. 이 값은 기본값입니다. 큐 관리자 속성의 값을 변경하려면 MQSC 명령, ALTER QMGR을 사용하고 매개변수 ACCTQ를 지정하십시오. 큐 관리자 속성 ACCTQ의 가능한 값은 다음과 같습니다.

### 켜짐(ON)

큐 속성 ACCTQ가 QMGR로 설정된 큐에 대해서 큐 회계 정보를 수집합니다.

### 꺼짐(OFF)

큐 속성 ACCTQ가 QMGR로 설정된 큐에 대해 큐 회계 정보를 수집하지 않습니다. 이 값은 기본값입니다.

### NONE

큐 속성 ACCTQ와 무관하게 모든 큐에 대한 큐 회계 정보 콜렉션이 사용 불가능합니다.

큐 관리자 속성, ACCTQ가 NONE으로 설정된 경우에는 큐 속성 ACCTQ와 무관하게 모든 큐에 대한 큐 회계 정보 콜렉션이 사용 안함으로 설정됩니다.

예를 들어, 큐 Q1에 대한 회계 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QLOCAL(Q1) ACCTQ(ON)
```

큐 속성 ACCTQ를 QMGR로 지정하는 모든 큐에 대한 회계 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR ACCTQ(ON)
```

## MQCONNX 옵션

MQCONNX 호출의 **ConnectOpts** 매개변수를 사용하여 큐 관리자 속성 ACCTMQI 및 ACCTQ의 유효 값을 대체해서, 연결 레벨에서 MQI와 큐 회계 정보 모두의 콜렉션을 수정합니다.

**ConnectOpts** 매개변수의 가능한 값은 다음과 같습니다.

### MQCNO\_ACCOUNTING\_MQI\_ENABLED

큐 관리자 속성 ACCTMQI의 값이 OFF로 지정된 경우 MQI 회계가 이 연결에 사용됩니다. 이는 큐 관리자 속성 ACCTMQI를 ON으로 지정한 경우와 동등합니다.

큐 관리자 속성 ACCTMQI의 값이 OFF로 지정되면 이 속성이 적용되지 않습니다.

### MQCNO\_ACCOUNTING\_MQI\_DISABLED

큐 관리자 속성 ACCTMQI의 값이 ON으로 지정된 경우 MQI 회계가 이 연결에 사용 불가능합니다. 이는 큐 관리자 속성 ACCTMQI를 OFF로 지정한 경우와 동등합니다.

큐 관리자 속성 ACCTMQI의 값이 ON으로 지정되면 이 속성이 적용되지 않습니다.

### MQCNO\_ACCOUNTING\_Q\_ENABLED

큐 관리자 속성 ACCTQ의 값이 OFF로 지정된 경우 큐 회계가 이 연결에 사용됩니다. ACCTQ가 QMGR로 지정된 모든 큐가 큐 회계에 대해 사용됩니다. 이는 큐 관리자 속성 ACCTQ를 ON으로 지정한 경우와 동등합니다.

큐 관리자 속성 ACCTQ의 값이 OFF로 지정되면 이 속성이 적용되지 않습니다.

### MQCNO\_ACCOUNTING\_Q\_DISABLED

큐 관리자 속성 ACCTQ의 값이 ON으로 지정된 경우 큐 회계가 이 연결에 사용 불가능합니다. 이는 큐 관리자 속성 ACCTQ를 OFF로 지정한 경우와 동등합니다.

큐 관리자 속성 ACCTQ의 값이 ON으로 지정되면 이 속성이 적용되지 않습니다.

이러한 대체는 기본적으로 사용 불가능합니다. 이 대체를 사용하려면 큐 관리자 속성 ACCTCONO를 ENABLED로 설정하십시오. 개별 연결에 대한 회계 대체를 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR ACCTCONO(ENABLED)
```

## 회계 메시지 생성

회계 메시지는 큐 관리자로부터의 애플리케이션 연결이 끊어질 때 생성됩니다. 중간 계정 메시지 또한 장기 실행 중인 WebSphere MQ 애플리케이션용으로 작성됩니다.

회계 메시지는 애플리케이션 연결이 끊어질 때 다음 중 한 방식으로 생성됩니다.

- 애플리케이션이 MQDISC 호출을 발행함
- 큐 관리자가 애플리케이션이 종료한 사실을 인식함

연결이 설정된 이후 또는 작성된 마지막 중간 계정 메시지 이후의 간격이 구성된 간격을 초과하는 경우, 중간 계정 메시지가 장기 실행 중인 WebSphere MQ 애플리케이션용으로 작성됩니다. 큐 관리자 속성, ACCTINT는 중간 회계 메시지가 자동으로 기록되기 시작하는 시간(초)을 지정합니다. 회계 메시지는 애플리케이션이 큐 관리자와 상호작용할 때에만 생성되므로, MQI 요청을 실행하지 않은 채로 오랜 기간 큐 관리자에 연결되어 있는 애플리케이션은 회계 간격이 완료된 후 첫 번째 MQI 요청이 실행될 때까지 회계 메시지를 생성하지 않습니다.

기본 회계 간격은 1800초(30분)입니다. 예를 들어, 회계 간격을 900초(15분)로 변경하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR ACCTINT(900)
```

## 통계 메시지

통계 메시지는 WebSphere MQ 시스템에서 발생하는 활동에 대한 정보를 기록합니다. 통계 메시지는 많은 PCF 구조를 포함한 PCF 메시지입니다.

통계 메시지는 구성된 간격으로 시스템 큐(SYSTEM.ADMIN.STATISTICS.QUEUE)에 전달됩니다.

통계 메시지는 다음 범주에 있습니다.

### MQI 통계 메시지

MQI 통계 메시지는 구성된 간격에 수행된 MQI 호출의 수에 관련한 정보를 포함합니다. 예를 들어, 큐 관리자가 실행한 MQI 호출의 수가 정보에 포함될 수 있습니다.

### 큐 통계 메시지

큐 통계 메시지는 구성된 간격 중 큐의 활동에 관련된 정보를 포함합니다. 큐에 넣고 큐에서 검색한 메시지 수와 큐에서 처리된 총 바이트 수가 정보에 포함됩니다.

각 큐 통계 메시지는 통계가 수집된 큐별 활동에 각 레코드가 관련되어 있는 최대 100개의 레코드를 포함할 수 있습니다.

통계 메시지는 로컬 큐에 대해서만 기록됩니다. 애플리케이션이 알리어스 큐에 대해 MQI 호출을 수행하면 기본 큐에 대해 통계 데이터가 기록되고 리모트 큐의 경우에는 전송 큐에 대해 통계 데이터가 기록됩니다.

### 채널 통계 메시지

채널 통계 메시지는 구성된 간격 중 채널의 활동에 관련된 정보를 포함합니다. 예를 들어, 채널에서 전송된 메시지 수나 채널에서 전송된 바이트 수와 같은 정보일 수 있습니다.

각 채널 통계 메시지는 통계가 수집된 채널별 활동에 각 레코드가 관련되어 있는 최대 100개의 레코드를 포함합니다.

### 관련 참조

#### [126 페이지의 『MQI 통계 정보』](#)

큐 관리자 속성 STATMQI를 사용하여 MQI 통계 정보 컬렉션을 제어합니다.

#### [126 페이지의 『큐 통계 정보』](#)

큐 속성 STATQ와 큐 관리자 속성 STATQ를 사용하여 큐 통계 정보의 컬렉션을 제어합니다.

#### [127 페이지의 『채널 통계 정보』](#)

채널 속성 STATCHL을 사용하여 채널 통계 정보 컬렉션을 제어합니다. 큐 관리자 속성을 사용하여 정보 컬렉션을 제어할 수도 있습니다. 이러한 속성은 분산형 플랫폼과 IBM i에서 사용할 수 있습니다.

## 통계 메시지 형식

통계 메시지는 메시지 디스크립터와 메시지 데이터로 이루어진 PCF 필드 세트 구성됩니다.

## 메시지 디스크립터

- 통계 메시지 MQMD(메시지 디스크립터)

## 회계 메시지 데이터

- 통계 메시지 MQCFH(PCF 헤더)
- 항상 리턴되는 통계 메시지 데이터
- 사용 가능한 경우에 리턴되는 통계 메시지 데이터

통계 메시지 MQCFH(PCF 헤더)는 통계 데이터가 기록된 간격에 대한 정보를 포함합니다.

통계 메시지 데이터는 통계 정보를 저장하는 PCF 매개변수로 구성됩니다. 통계 메시지의 콘텐츠는 다음과 같은 메시지 범주에 따라 다릅니다.

### MQI 통계 메시지

MQI 통계 메시지 데이터는 많은 PCF 매개변수로 구성됩니다(PCF 그룹은 없음).

### 큐 통계 메시지

큐 통계 메시지 데이터는 많은 PCF 매개변수로 구성되며 범위는 1 ~ 100 *QStatisticsData* PCF 그룹입니다.

간격의 활성화된 모든 큐에 *QStatisticsData* PCF 그룹이 하나씩 있습니다. 간격 중에 100개를 초과한 큐가 활성화되면 여러 통계 메시지가 생성됩니다. 각 메시지마다 MQCFH(PCF 헤더)의 *SeqNumber*가 그에 맞춰 업데이트되고, 순서의 마지막 메시지에는 MQCFH의 *Control* 매개변수가 MQCFH\_LAST로 지정됩니다.

### 채널 통계 메시지

채널 통계 메시지 데이터는 많은 PCF 매개변수로 구성되며 범위는 1 ~ 100 *ChlStatisticsData* PCF 그룹입니다.

간격의 활성화된 모든 채널에 *ChlStatisticsData* PCF 그룹이 하나씩 있습니다. 간격 중에 100개를 초과한 채널이 활성화되면 여러 통계 메시지가 생성됩니다. 각 메시지마다 MQCFH(PCF 헤더)의 *SeqNumber*가 그에 맞춰 업데이트되고, 순서의 마지막 메시지에는 MQCFH의 *Control* 매개변수가 MQCFH\_LAST로 지정됩니다.

## 통계 정보 콜렉션

큐, 큐 관리자, 채널 속성을 사용하여 통계 정보 콜렉션을 제어합니다.

### MQI 통계 정보

큐 관리자 속성 STATMQI를 사용하여 MQI 통계 정보 콜렉션을 제어합니다.

이 속성의 값을 변경하려면 MQSC 명령 ALTER QMGR 를 사용하고 STATMQI 매개변수를 지정하십시오. 통계 메시지는 통계 콜렉션이 사용 가능하게 된 후에 열린 큐에 대해서만 생성됩니다. STATMQI 매개변수의 가능한 값은 다음과 같습니다.

#### 켜짐(ON)

모든 큐 관리자 연결에 대한 MQI 통계 정보를 수집합니다.

#### 꺼짐(OFF)

MQI 통계 정보를 수집하지 않습니다. 이 값은 기본값입니다.

예를 들어, MQI 통계 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR STATMQI(ON)
```

### 큐 통계 정보

큐 속성 STATQ와 큐 관리자 속성 STATQ를 사용하여 큐 통계 정보의 콜렉션을 제어합니다.

개별 큐나 여러 큐에 대해서 큐 통계 정보 콜렉션을 사용 가능 또는 불가능하게 할 수 있습니다. 개별 큐를 제어하려면 큐 속성 STATQ를 설정하십시오. 큐 관리자 속성 STATQ를 사용하여 큐 관리자 레벨에서 큐 통계 정보 콜렉션을 사용 또는 사용 안함으로 설정합니다. 큐 속성 STATQ가 QMGR 값으로 지정된 모든 큐의 경우에는 큐 관리자 레벨에서 큐 통계 정보 콜렉션이 제어됩니다.

큐 통계는 통계 콜렉션이 사용 가능하게 된 후에 열린 IBM WebSphere MQ MQI 오브젝트 핸들을 사용하는 조작에 대해서만 증분됩니다.

큐 통계 메시지는 이전 기간에 통계 데이터가 수집된 큐에 대해서만 생성됩니다.

동일한 큐에는 여러 오브젝트 핸들을 통해 여러 넣기 조작과 가져오기 조작이 있을 수 있습니다. 일부 오브젝트 핸들은 통계 콜렉션이 사용 가능하게 되기 전에 열렸을 수 있지만 다른 핸들은 그 후에 열렸습니다. 따라서 큐 통계가 모든 넣기 조작 및 가져오기 조작이 아닌 이들 일부 조작의 활동을 기록할 수 있습니다.

큐 통계가 모든 애플리케이션의 활동을 기록하게 하려면 모니터링 중인 단일 또는 다중 큐의 오브젝트 핸들을 닫고 새 오브젝트 핸들을 열어야 합니다. 이를 위한 최상의 방법은 통계 콜렉션을 사용 가능하게 한 후 모든 애플리케이션을 종료하고 다시 시작하는 것입니다.

큐 속성 STATQ의 값을 변경하려면 MQSC 명령, ALTER QLOCAL을 사용하고 매개변수 STATQ를 지정하십시오. 큐 속성 STATQ의 가능한 값은 다음과 같습니다.

#### 켜짐(ON)

큐 통계 정보는 큐를 여는 모든 큐 관리자 연결에 대해 수집됩니다.

#### 꺼짐(OFF)

이 큐에 대한 큐 통계 정보를 수집하지 않습니다.

#### 큐 관리자

이 큐에 대한 큐 통계 정보 콜렉션은 큐 관리자 속성, STATQ의 값에 따라 제어됩니다. 이 값은 기본값입니다.

큐 관리자 속성 STATQ의 값을 변경하려면 MQSC 명령, ALTER QMGR을 사용하고 매개변수 STATQ를 지정하십시오. 큐 관리자 속성 STATQ의 가능한 값은 다음과 같습니다.

#### 켜짐(ON)

큐 속성 STATQ가 QMGR로 설정된 큐에 대해서 큐 통계 정보를 수집합니다.

#### 꺼짐(OFF)

큐 속성 STATQ가 QMGR로 설정된 큐에 대해 큐 통계 정보를 수집하지 않습니다. 이 값은 기본값입니다.

#### NONE

큐 속성 STATQ와 무관하게 모든 큐에 대한 큐 통계 정보 콜렉션이 사용 불가능합니다.

큐 관리자 속성 STATQ가 NONE으로 설정된 경우에는 큐 속성 STATQ와 무관하게 모든 큐에 대한 큐 통계 정보 콜렉션이 사용 안함으로 설정됩니다.

예를 들어, 큐 Q1에 대한 통계 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QLOCAL(Q1) STATQ(ON)
```

큐 속성 STATQ를 QMGR로 지정하는 모든 큐에 대한 통계 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR STATQ(ON)
```

### **distributed** 채널 통계 정보

채널 속성 STATCHL을 사용하여 채널 통계 정보 콜렉션을 제어합니다. 큐 관리자 속성을 사용하여 정보 콜렉션을 제어할 수도 있습니다. 이러한 속성은 분산형 플랫폼과 IBM i에서 사용할 수 있습니다.

개별 채널이나 여러 채널에 대해서 채널 통계 정보 콜렉션을 사용 가능 또는 불가능하게 할 수 있습니다. 개별 채널을 제어하려면 채널 통계 정보 콜렉션을 사용 또는 사용하지 않도록 채널 속성 STATCHL을 설정해야 합니다. 많은 채널을 함께 제어하려면 큐 관리자 속성 STATCHL을 사용하여 큐 관리자 레벨에서 채널 통계 정보 콜렉션을 사용 가능 또는 사용 불가능으로 설정할 수 있습니다. 채널 속성 STATCHL이 값 QMGR을 사용하여 지정된 모든 채널에 대해 채널 통계 정보 콜렉션이 큐 관리자 레벨에서 제어됩니다.

자동으로 정의된 클러스터 송신자 채널이 WebSphere MQ 오브젝트가 아니므로 채널 오브젝트와 동일한 방법으로 속성을 갖지 마십시오. 자동으로 정의된 클러스터 송신자 채널을 제어하려면 큐 관리자 속성 STACLs를 사용하십시오. 이 속성은 큐 관리자 내의 자동으로 정의된 클러스터 송신자 채널이 채널 통계 정보 콜렉션에 대해 사용 가능 또는 불가능한지 여부를 판별합니다.

채널 통계 정보 콜렉션을 세 가지 모니터링 레벨인 낮음, 중간 또는 높음 중 하나로 설정할 수 있습니다. 오브젝트 레벨 또는 큐 관리자 레벨에서 모니터링 레벨을 설정할 수 있습니다. 사용할 레벨의 선택은 시스템에 따라 다릅니다. 통계 정보 데이터를 수집하려면 계산 비용이 상대적으로 비싼 몇 가지 지시사항이 필요할 수 있으므로, 채널

통계 정보 콜렉션의 영향을 줄이기 위해 중간 및 낮음 모니터링 옵션은 항상 데이터를 수집하기 보다는 정기적으로 데이터 샘플을 측정합니다. [128 페이지의 표 22](#)에서는 채널 통계 정보 콜렉션에 사용 가능한 레벨을 요약합니다.

표 22. 채널 통계 정보 콜렉션의 세부사항 레벨		
레벨	설명	사용법
낮음	일정한 간격으로 작은 데이터 샘플을 측정합니다.	높은 볼륨의 메시지를 처리하는 오브젝트의 경우.
보통	일정한 간격으로 데이터 샘플을 측정합니다.	대부분의 오브젝트의 경우.
높음	일정한 간격으로 모든 데이터를 측정합니다.	초당 몇 개의 메시지만을 처리하는 오브젝트의 경우(최신 정보가 중요함).

채널 속성 `STATCHL`의 값을 변경하려면 `MQSC` 명령, `ALTER CHANNEL`을 사용하고 매개변수 `STATCHL`을 지정하십시오.

큐 관리자 속성 `STATCHL`의 값을 변경하려면 `MQSC` 명령, `ALTER QMGR`을 사용하고 매개변수 `STATCHL`을 지정하십시오.

큐 관리자 속성 `STATACLS`의 값을 변경하려면 `MQSC` 명령, `ALTER QMGR`을 사용하고 매개변수 `STATACLS`를 지정하십시오.

채널 속성, `STATCHL`의 가능한 값은 다음과 같습니다.

**낮음(LOW)**

채널 통계 정보가 낮은 세부사항 레벨로 수집됩니다.

**보통(MEDIUM)**

채널 통계 정보가 중간 세부사항 레벨로 수집됩니다.

**높음(HIGH)**

채널 통계 정보가 높은 세부사항 레벨로 수집됩니다.

**꺼짐(OFF)**

이 채널에 대한 채널 통계 정보가 수집되지 않습니다.

**큐 관리자**

채널 속성이 `QMGR`로 설정됩니다. 이 채널에 대한 통계 정보 콜렉션은 큐 관리자 속성, `STATCHL`의 값으로 제어됩니다.

이 값은 기본값입니다.

큐 관리자 속성, `STATCHL`의 가능한 값은 다음과 같습니다.

**낮음(LOW)**

채널 속성 `STATCHL`이 `QMGR`로 설정된 모든 채널에 대한 채널 통계 정보가 낮은 세부사항 레벨로 수집됩니다.

**보통(MEDIUM)**

채널 속성 `STATCHL`이 `QMGR`로 설정된 모든 채널에 대한 채널 통계 정보가 중간 세부사항 레벨로 수집됩니다.

**높음(HIGH)**

채널 속성 `STATCHL`이 `QMGR`로 설정된 모든 채널에 대한 채널 통계 정보가 높은 세부사항 레벨로 수집됩니다.

**꺼짐(OFF)**

채널 속성 `STATCHL`이 `QMGR`로 설정된 모든 채널에 대한 채널 통계 정보가 수집되지 않습니다.

이 값은 기본값입니다.

**NONE**

채널 속성 `STATCHL`과 무관하게 모든 채널에 대한 채널 통계 정보 콜렉션이 사용 불가능합니다.

큐 관리자 속성, STATACLS의 가능한 값은 다음과 같습니다.

#### 낮음(LOW)

자동으로 정의된 클러스터 송신자 채널에 대한 통계 정보가 낮은 세부사항 레벨로 수집됩니다.

#### 보통(MEDIUM)

자동으로 정의된 클러스터 송신자 채널에 대한 통계 정보가 중간 세부사항 레벨로 수집됩니다.

#### 높음(HIGH)

자동으로 정의된 클러스터 송신자 채널에 대한 통계 정보가 높은 세부사항 레벨로 수집됩니다.

#### 꺼짐(OFF)

통계 정보가 자동으로 정의된 클러스터 송신자 채널용이 아닙니다.

#### 큐 관리자

자동으로 정의된 클러스터 송신자 채널에 대한 통계 정보 콜렉션은 큐 관리자 속성, STATCHL의 값으로 제어됩니다.

이 값은 기본값입니다.

예를 들어, 송신자 채널 QM1.TO.QM2에 대한 통계 정보 콜렉션을 중간 세부사항 레벨로 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) STATCHL(MEDIUM)
```

채널 속성 STATCHL을 QMGR로 지정하는 모든 채널에 대한 통계 정보 콜렉션을 중간 세부사항 레벨로 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR STATCHL(MEDIUM)
```

모든 자동으로 정의된 클러스터 송신자 채널에 대한 통계 정보 콜렉션을 중간 세부사항 레벨로 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR STATACLS(MEDIUM)
```

## 통계 메시지 생성

통계 메시지는 구성된 간격에, 큐 관리자가 제어된 방식으로 시스템 종료할 때 생성됩니다.

구성된 간격은 통계 메시지 생성 사이의 간격(초)을 지정하는 STATINT 큐 관리자 속성으로 제어됩니다. 기본 통계 간격은 1800초(30분)입니다. 통계 간격을 변경하려면 MQSC 명령 ALTER QMGR 를 사용하고 STATINT 매개 변수를 지정하십시오. 예를 들어, 통계 간격을 900초(15분)로 변경하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR STATINT(900)
```

통계 콜렉션 간격이 만기되려 하기 전에 현재 수집한 통계 데이터를 통계 큐에 쓰려면 다음 MQSC 명령 RESET QMGR TYPE(STATISTICS)를 사용하십시오. 이 명령을 발행하면 통계 큐에 수집한 통계 데이터가 기록되고 새 통계 데이터 콜렉션 간격이 시작됩니다.

## 회계 및 통계 정보 표시

회계 및 통계 메시지에 기록된 정보를 사용하려면 **amqsmn** 샘플 프로그램과 같은 애플리케이션을 실행하여 기록된 정보를 적합한 형식으로 변환하십시오.

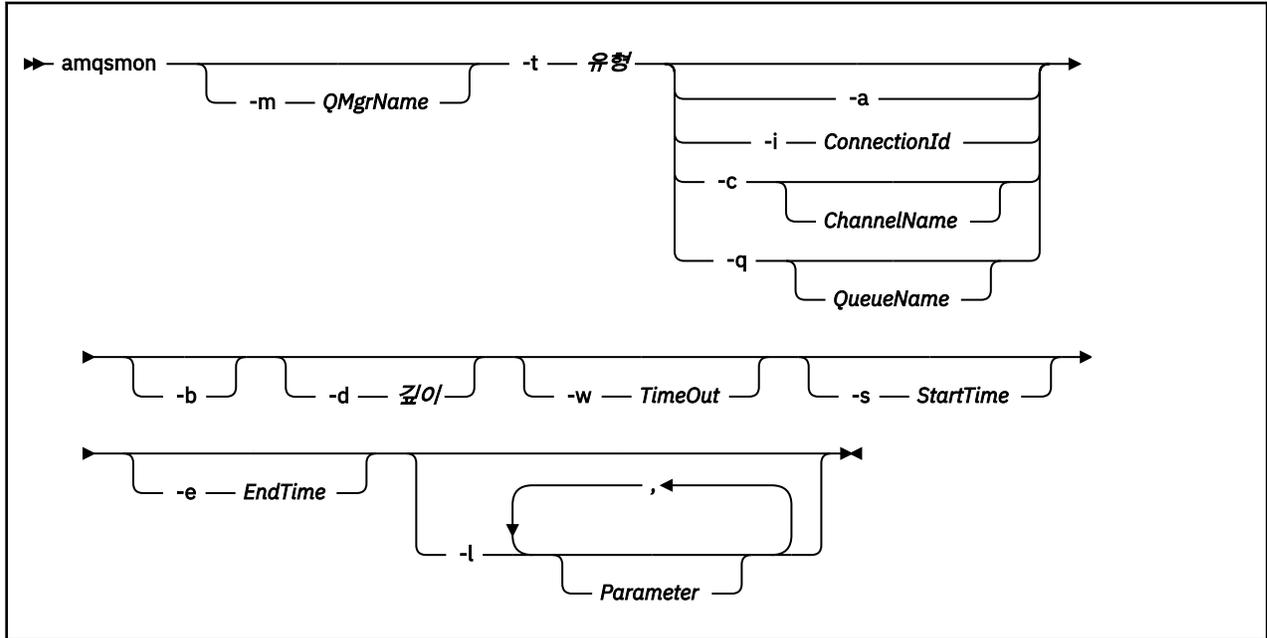
회계 및 통계 메시지는 시스템 회계 및 통계 큐에 기록됩니다. **amqsmn**은 계정 및 통계 큐로부터 메시지를 처리하고 읽을 수 있는 형식으로 화면에 정보를 표시하는 WebSphere MQ와 함께 제공되는 샘플 프로그램입니다.

**amqsmn**은 샘플 프로그램이므로 회계 또는 통계 메시지를 처리할 자신만의 애플리케이션을 쓰기 위한 템플릿으로 제공된 소스 코드를 사용하거나 **amqsmn** 소스 코드를 자신의 특정 요구사항에 맞게 수정할 수 있습니다.

## amqsmon(형식화된 모니터링 정보 표시)

**amqsmon** 샘플 프로그램을 사용하여 회계 및 통계 메시지 내에 포함된 정보를 읽기 쉬운 형식으로 표시합니다. **amqsmon** 프로그램은 계정 큐 SYSTEM.ADMIN.ACCOUNTING.QUEUE에서 계정 메시지를 읽습니다. 통계 큐 SYSTEM.ADMIN.STATISTICS.QUEUE에서 통계 메시지를 읽습니다.

### 구문



### 필수 매개변수

#### -t Type

처리할 메시지의 유형. Type을 다음 중 하나로 지정하십시오.

#### 회계

회계 레코드가 처리됩니다. 시스템 큐, SYSTEM.ADMIN.ACCOUNTING.QUEUE에서 메시지를 읽습니다.

#### 통계

통계 레코드가 처리됩니다. 시스템 큐, SYSTEM.ADMIN.STATISTICS.QUEUE에서 메시지를 읽습니다.

### 선택적 매개변수

#### -m QMgrName

회계 또는 통계 메시지를 처리할 큐 관리자의 이름.

이 매개변수를 지정하지 않으면 기본 큐 관리자가 사용됩니다.

#### -a

MQI 레코드만을 포함한 프로세스 메시지.

MQI 레코드만 표시합니다. MQI 레코드를 포함하지 않은 메시지는 메시지를 읽는 큐에 항상 남습니다.

#### -q QueueName

QueueName은 선택적 매개변수입니다.

QueueName을 제공하지 않을 경우: 큐 회계와 큐 통계 레코드만 표시합니다.

*QueueName*을 제공할 경우:

*QueueName*에 지정된 큐에 대한 큐 회계와 큐 통계 레코드만 표시합니다.

-*b*가 지정되지 않으면 레코드가 생성된 계정 및 통계 메시지가 제거됩니다. 계정 및 통계 메시지도 기타 큐로부터 레코드를 포함할 수 있으므로 -*b*가 지정되지 않으면 보이지 않는 레코드가 제거될 수 있습니다.

### **-c ChannelName**

*ChannelName*은 선택적 매개변수입니다.

*ChannelName*을 제공하지 않을 경우:

채널 통계 레코드만 표시합니다.

*ChannelName*을 제공할 경우:

*ChannelName*에 지정된 채널의 채널 통계 레코드만 표시합니다.

-*b*가 지정되지 않으면 레코드가 생성된 통계 메시지가 제거됩니다. 통계 메시지도 기타 채널로부터 레코드를 포함할 수 있으므로 -*b*가 지정되지 않으면 보이지 않는 레코드가 제거될 수 있습니다.

이 매개변수는 통계 메시지만 표시하는 경우(-*t statistics*), 사용 가능합니다.

### **-i ConnectionId**

*ConnectionId*에 지정된 연결 ID와 관련이 있는 레코드만 표시합니다.

이 매개변수는 계정 메시지만 표시하는 경우(-*t accounting*), 사용 가능합니다.

-*b*가 지정되지 않으면 레코드가 생성된 통계 메시지가 제거됩니다. 통계 메시지도 기타 채널로부터 레코드를 포함할 수 있으므로 -*b*가 지정되지 않으면 보이지 않는 레코드가 제거될 수 있습니다.

### **-b**

메시지를 찾아봅니다.

메시지는 비파괴적으로 검색됩니다.

### **-d Depth**

처리 가능한 최대 메시지 수.

이 매개변수를 지정하지 않으면 무제한 수의 메시지가 처리될 수 있습니다.

### **-w TimeOut**

메시지가 사용 가능하게 될 때까지 대기할 최대 시간(초).

이 매개변수를 지정하지 않을 경우 처리할 메시지가 더 이상 없으면 *amqsmn*이 종료됩니다.

### **-s StartTime**

지정된 *StartTime* 후에만 넣은 프로세스 메시지.

*StartTime* 은 *yyyy-mm-dd hh.mm.ss* 형식으로 지정됩니다. 시간 없이 날짜가 지정되면 시간은 기본적으로 지정된 날짜에 00.00.00으로 설정됩니다. 시간은 GMT입니다.

이 매개변수를 지정하지 않을 경우에 미치는 영향은 [참고 1](#)을 참조하십시오.

### **-e EndTime**

지정된 *EndTime* 이전에만 넣은 프로세스 메시지.

*EndTime*은 *yyyy-mm-dd hh.mm.ss* 형식으로 지정됩니다. 시간 없이 날짜가 지정되면 시간은 기본적으로 지정된 날짜에 23.59.59으로 설정됩니다. 시간은 GMT입니다.

이 매개변수를 지정하지 않을 경우에 미치는 영향은 [참고 1](#)을 참조하십시오.

## -l Parameter

처리된 레코드에서 선택한 필드만 표시합니다. *Parameter*는 쉘표로 구분된 정수 값 목록으로, 각 정수 값이 필드의 숫자 상수에 매핑됩니다(amqsmon 예 5 참조).

이 매개변수를 지정하지 않으면 사용 가능한 모든 필드가 표시됩니다.

## 참고:

1. *-s StartTime* 또는 *-e EndTime*을 지정하지 않을 경우 처리 가능한 메시지가 넣기 시간까지 제한되지 않습니다.

## amqsmon 예

이 페이지를 사용하여 amqsmon(형식화된 모니터링 정보 표시) 샘플 프로그램 실행 예를 봅니다.

1. 다음 명령은 큐 관리자 saturn.queue.manager의 모든 MQI 통계 메시지를 표시합니다.

```
amqsmon -m saturn.queue.manager -t statistics -a
```

이 명령의 출력은 다음과 같습니다.

```
RecordType: MQIStatistics
QueueManager: 'saturn.queue.manager'
IntervalStartDate: '2005-04-30'
IntervalStartTime: '15.09.02'
IntervalEndDate: '2005-04-30'
IntervalEndTime: '15.39.02'
CommandLevel: 600
ConnCount: 23
ConnFailCount: 0
ConnsMax: 8
DiscCount: [17, 0, 0]
OpenCount: [0, 80, 1, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0, 0]
OpenFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
CloseCount: [0, 73, 1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
CloseFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
InqCount: [4, 2102, 0, 0, 0, 46, 0, 0, 0, 0, 0, 0, 0, 0]
InqFailCount: [0, 31, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
SetCount: [0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
SetFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
PutCount: [26, 1]
PutFailCount: 0
Put1Count: [40, 0]
Put1FailCount: 0
PutBytes: [57064, 12320]
GetCount: [18, 1]
GetBytes: [52, 12320]
GetFailCount: 2254
BrowseCount: [18, 60]
BrowseBytes: [23784, 30760]
BrowseFailCount: 9
CommitCount: 0
CommitFailCount: 0
BackCount: 0
ExpiredMsgCount: 0
PurgeCount: 0
```

2. 다음 명령은 큐 관리자 saturn.queue.manager의 큐 LOCALQ에 대한 모든 큐 통계 메시지를 표시합니다.

```
amqsmon -m saturn.queue.manager -t statistics -q LOCALQ
```

이 명령의 출력은 다음과 같습니다.

```
RecordType: QueueStatistics
QueueManager: 'saturn.queue.manager'
IntervalStartDate: '2005-04-30'
IntervalStartTime: '15.09.02'
IntervalEndDate: '2005-04-30'
IntervalEndTime: '15.39.02'
CommandLevel: 600
ObjectCount: 3
```

```

QueueStatistics:
  QueueName: 'LOCALQ'
  CreateDate: '2005-03-08'
  CreateTime: '17.07.02'
  QueueType: Predefined
  QueueDefinitionType: Local
  QMinDepth: 0
  QMaxDepth: 18
  AverageQueueTime: [29827281, 0]
  PutCount: [26, 0]
  PutFailCount: 0
  Put1Count: [0, 0]
  Put1FailCount: 0
  PutBytes: [88, 0]
  GetCount: [18, 0]
  GetBytes: [52, 0]
  GetFailCount: 0
  BrowseCount: [0, 0]
  BrowseBytes: [0, 0]
  BrowseFailCount: 1
  NonQueuedMsgCount: 0
  ExpiredMsgCount: 0
  PurgedMsgCount: 0

```

3. 다음 명령은 큐 관리자 saturn.queue.manager에서 2005년 4월 30일 15:30분 이후에 기록된 모든 통계 메시지를 표시합니다.

```
amqsmon -m saturn.queue.manager -t statistics -s "2005-04-30 15.30.00"
```

이 명령의 출력은 다음과 같습니다.

```

RecordType: MQIStatistics
QueueManager: 'saturn.queue.manager'
IntervalStartDate: '2005-04-30'
IntervalStartTime: '15.09.02'
IntervalEndDate: '2005-04-30'
IntervalEndTime: '15.39.02'
CommandLevel: 600
ConnCount: 23
ConnFailCount: 0
ConnsMax: 8
DiscCount: [17, 0, 0]
OpenCount: [0, 80, 1, 0, 0, 3, 0, 0, 0, 0, 0, 0]
...
RecordType: QueueStatistics
QueueManager: 'saturn.queue.manager'
IntervalStartDate: '2005-04-30'
IntervalStartTime: '15.09.02'
IntervalEndDate: '2005-04-30'
IntervalEndTime: '15.39.02'
CommandLevel: 600
ObjectCount: 3
QueueStatistics: 0
  QueueName: 'LOCALQ'
  CreateDate: '2005-03-08'
  CreateTime: '17.07.02'
  QueueType: Predefined
  ...
QueueStatistics: 1
  QueueName: 'SAMPLEQ'
  CreateDate: '2005-03-08'
  CreateTime: '17.07.02'
  QueueType: Predefined
  ...

```

4. 다음 명령은 큐 관리자 saturn.queue.manager에서 2005년 4월 30일에 기록된 모든 회계 메시지를 표시합니다.

```
amqsmon -m saturn.queue.manager -t accounting -s "2005-04-30" -e "2005-04-30"
```

이 명령의 출력은 다음과 같습니다.

```
RecordType: MQIAccounting
```

```

QueueManager: 'saturn.queue.manager'
IntervalStartDate: '2005-04-30'
IntervalStartTime: '15.09.29'
IntervalEndDate: '2005-04-30'
IntervalEndTime: '15.09.30'
CommandLevel: 600
ConnectionId: x'414d514354524556312020202020208d0b3742010a0020'
SeqNumber: 0
ApplicationName: 'amqsput'
ApplicationPid: 8572
ApplicationTid: 1
UserId: 'admin'
ConnDate: '2005-03-16'
ConnTime: '15.09.29'
DiscDate: '2005-03-16'
DiscTime: '15.09.30'
DiscType: Normal
OpenCount: [0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
OpenFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
CloseCount: [0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
CloseFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
PutCount: [1, 0]
PutFailCount: 0
PutBytes: [4, 0]
GetCount: [0, 0]
GetFailCount: 0
GetBytes: [0, 0]
BrowseCount: [0, 0]
BrowseFailCount: 0
BrowseBytes: [0, 0]
CommitCount: 0
CommitFailCount: 0
BackCount: 0
InqCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
InqFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
SetCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]
SetFailCount: [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

RecordType: MQIAccounting
QueueManager: 'saturn.queue.manager'
IntervalStartDate: '2005-03-16'
IntervalStartTime: '15.16.22'
IntervalEndDate: '2005-03-16'
IntervalEndTime: '15.16.22'
CommandLevel: 600
ConnectionId: x'414d514354524556312020202020208d0b3742010c0020'
SeqNumber: 0
ApplicationName: 'runmqsc'
ApplicationPid: 8615
ApplicationTid: 1
...

```

5. 다음 명령은 회계 큐를 찾아보고 MQI 회계 정보가 사용 가능한 모든 애플리케이션의 애플리케이션 이름과 연결 ID를 표시합니다.

```
amqsmon -m saturn.queue.manager -t accounting -b -a -l 7006,3024
```

이 명령의 출력은 다음과 같습니다.

```

ConnectionId: x'414d514354524556312020202020208d0b374203090020'
ApplicationName: 'runmqsc'

ConnectionId: x'414d514354524556312020202020208d0b3742010a0020'
ApplicationName: 'amqsput'

ConnectionId: x'414d514354524556312020202020208d0b3742010c0020'
ApplicationName: 'runmqsc'

ConnectionId: x'414d514354524556312020202020208d0b3742010d0020'
ApplicationName: 'amqsput'

ConnectionId: x'414d514354524556312020202020208d0b3742150d0020'
ApplicationName: 'amqsget'

5 Records Processed.

```

## 회계 및 통계 메시지 참조

이 페이지를 사용하여 회계 및 통계 메시지의 형식과 이 메시지에 리턴되는 정보를 개괄합니다.

계정 및 통계 메시지는 메시지 디스크립터 및 메시지 데이터를 포함하는 표준 WebSphere MQ 메시지입니다. 메시지 데이터는 WebSphere MQ 애플리케이션에 의해 수행되는 MQI 조작에 대한 정보 또는 WebSphere MQ 시스템에서 발생하는 활동에 대한 정보를 포함합니다.

### 메시지 디스크립터

- MQMD 구조

### 메시지 데이터

- PCF 헤더(MQCFH)
- 항상 리턴되는 회계 또는 통계 메시지 데이터
- 사용 가능한 경우에 리턴되는 회계 또는 통계 메시지 데이터

## 회계 및 통계 메시지 형식

이 페이지를 MQI 회계 메시지의 구조 예로 사용하십시오.

표 23. MQI 회계 메시지 구조		
MQMD 구조	회계 메시지 헤더 MQCFH 구조	MQI 계정 메시지 데이터 <sup>1</sup>
구조 ID 구조 버전 보고서 옵션 메시지 유형 만기 시간 피드백 코드 Encoding 코드화 문자 세트 ID 메시지 형식 메시지 우선순위 지속 메시지 ID 상관 ID 백아웃 수 응답 대상 큐 응답 대상 큐 관리자 사용자 ID 계정 토큰 애플리케이션 ID 데이터 애플리케이션 유형 애플리케이션 이름 넣은 날짜 넣은 시간 애플리케이션 원본 데이터 그룹 ID 메시지 순서 번호 오프셋 메시지 플래그 원래 길이	구조 유형 구조 길이 구조 버전 명령 ID 메시지 순서 번호 제어 옵션 완료 코드 이유 코드 매개변수 수	큐 관리자 간격 시작 날짜 간격 시작 시간 간격 종료 날짜 간격 종료 시간 명령 레벨 연결 ID 순서 번호 애플리케이션 이름 애플리케이션 프로세스 ID 애플리케이션 스레드 ID 사용자 ID 연결 날짜 연결 시간 연결 이름 채널 이름 연결 해제 날짜 연결 해제 시간 연결 해제 유형 열기 수 열기 실패 수 닫기 수 닫기 실패 수 넣기 수 넣기 실패 수 Put1 수 Put1 실패 수 넣기 바이트 가져오기 수 가져오기 실패 수 가져오기 바이트 찾아보기 수 찾아보기 실패 수 찾아보기 바이트 커밋 수 커밋 실패 수 백아웃 수 조회 수 조회 실패 수 설정 수 설정 실패 수
<b>참고:</b> 1. 표시된 매개변수는 MQI 회계 메시지에 대해 리턴된 매개변수입니다. 실제 회계 또는 통계 메시지 데이터는 메시지 범주에 따라 다릅니다.		

### 회계 및 통계 메시지 MQMD(메시지 디스크립터)

이 페이지를 사용하여 회계 및 통계 메시지의 메시지 디스크립터와 이벤트 메시지의 메시지 디스크립터 간 차이를 이해합니다.

회계 및 통계 메시지의 메시지 디스크립터의 매개변수와 값은 다음을 제외하면, 이벤트 메시지의 메시지 디스크립터에서와 동일합니다.

#### **Format**

설명: 메시지 데이터의 형식 이름입니다.  
데이터 유형: MQCHAR8.  
가치: **MQFMT\_ADMIN**  
관리 메시지.

회계 및 통계 메시지의 메시지 디스크립터에 포함된 일부 매개변수는 메시지를 생성한 큐 관리자가 제공하는 수정된 데이터를 포함합니다.

MQCMD는 메시지를 넣는 큐 관리자의 이름(28자에서 잘림)과 회계 또는 통계 큐에 메시지를 넣은 날짜 및 시간도 지정합니다.

### **회계 및 통계 메시지의 메시지 데이터**

회계 및 통계 메시지의 메시지 데이터는 PCF 명령 조회 및 응답에 사용되는 프로그래밍 가능 명령 형식(PCF)을 기반으로 합니다. 회계 및 통계 메시지의 메시지 데이터는 PCF 헤더(MQCFH)와 회계 또는 통계 보고서로 구성됩니다.

### **회계 및 통계 메시지 MQCFH(PCF 헤더)**

회계 및 통계 메시지의 메시지 헤더는 MQCFH 구조입니다. 회계 및 통계 메시지의 메시지 헤더의 매개변수와 값은 다음을 제외하면, 이벤트 메시지의 메시지 헤더에서와 동일합니다.

#### **Command**

설명: 명령 ID입니다. 회계 또는 통계 메시지 범주를 식별합니다.  
데이터 유형: MQLONG.  
값: **MQCMD\_ACCOUNTING\_MQI**  
MQI 회계 메시지.  
**MQCMD\_ACCOUNTING\_Q**  
큐 회계 메시지.  
**MQCMD\_STATISTICS\_MQI**  
MQI 통계 메시지.  
**MQCMD\_STATISTICS\_Q**  
큐 통계 메시지.  
**MQCMD\_STATISTICS\_CHANNEL**  
채널 통계 메시지.

#### **Version**

설명: 구조 버전 번호입니다.  
데이터 유형: MQLONG.  
가치: **MQCFH\_VERSION\_3**  
회계 및 통계 메시지 버전-3.

### **회계 및 통계 메시지 데이터**

회계 및 통계 메시지 데이터의 콘텐츠는 다음과 같이 회계 또는 통계 메시지의 범주에 따라 다릅니다.

#### **MQI 회계 메시지**

MQI 회계 메시지 데이터는 많은 PCF 매개변수로 구성됩니다(PCF 그룹은 없음).

### 큐 회계 메시지

큐 회계 메시지 데이터는 많은 PCF 매개변수로 구성되며 범위는 1 ~ 100 *QAccountingData* PCF 그룹입니다.

### MQI 통계 메시지

MQI 통계 메시지 데이터는 많은 PCF 매개변수로 구성됩니다(PCF 그룹은 없음).

### 큐 통계 메시지

큐 통계 메시지 데이터는 많은 PCF 매개변수로 구성되며 범위는 1 ~ 100 *QStatisticsData* PCF 그룹입니다.

### 채널 통계 메시지

채널 통계 메시지 데이터는 많은 PCF 매개변수로 구성되며 범위는 1 ~ 100 *ChlStatisticsData* PCF 그룹입니다.

## MQI 회계 메시지 데이터

이 페이지를 사용하여 MQI 회계 메시지의 구조를 봅니다.

메시지 이름:	MQI 회계 메시지.
플랫폼:	All, except WebSphere MQ for z/OS.
시스템 큐:	SYSTEM.ADMIN.ACCOUNTING.QUEUE.

### **QueueManager**

설명:	큐 매니저의 이름
ID:	MQCA_Q_MGR_NAME
데이터 유형:	MQCFST
최대 길이:	MQ_Q_MGR_NAME_LENGTH
리턴됨:	Always

### **IntervalStartDate**

설명:	모니터링 기간의 시작 날짜
ID:	MQCAMO_START_DATE
데이터 유형:	MQCFST
최대 길이:	MQ_DATE_LENGTH
리턴됨:	Always

### **IntervalStartTime**

설명:	모니터링 기간의 시작 시간
ID:	MQCAMO_START_TIME
데이터 유형:	MQCFST
최대 길이:	MQ_TIME_LENGTH
리턴됨:	Always

### **IntervalEndDate**

설명:	모니터링 기간의 종료 날짜
ID:	MQCAMO_END_DATE
데이터 유형:	MQCFST
최대 길이:	MQ_DATE_LENGTH

리턴됨: Always

### ***IntervalEndTime***

설명: 모니터링 기간의 종료 시간  
ID: MQCAMO\_END\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: Always

### ***CommandLevel***

설명: 큐 관리자 명령 레벨  
ID: MQIA\_COMMAND\_LEVEL  
데이터 유형: MQCFIN  
리턴됨: Always

### ***ConnectionId***

설명: WebSphere MQ 연결의 연결 ID  
ID: MQBACF\_CONNECTION\_ID  
데이터 유형: MQCFBS  
최대 길이: MQ\_CONNECTION\_ID\_LENGTH  
리턴됨: Always

### ***SeqNumber***

설명: 순서 번호. 이 값은 장기 실행 연결에 대한 각 후속 레코드마다 증분됩니다.  
ID: MQIACF\_SEQUENCE\_NUMBER  
데이터 유형: MQCFIN  
리턴됨: Always

### ***ApplicationName***

설명: 애플리케이션의 이름. 이 필드의 콘텐츠는 메시지 디스크립터의 *PutApplName* 필드 콘텐츠에 해당합니다.  
ID: MQCACF\_APPL\_NAME  
데이터 유형: MQCFST  
최대 길이: MQ\_APPL\_NAME\_LENGTH  
리턴됨: Always

### ***ApplicationPid***

설명: 애플리케이션의 운영 체제 프로세스 ID  
ID: MQIACF\_PROCESS\_ID  
데이터 유형: MQCFIN  
리턴됨: Always

### ***ApplicationId***

설명:	애플리케이션 내에서 연결의 WebSphere MQ 스레드 ID입니다.
ID:	MQIACF_THREAD_ID
데이터 유형:	MQCFIN
리턴됨:	Always

### ***UserId***

설명:	애플리케이션의 사용자 ID 컨텍스트
ID:	MQCACF_USER_IDENTIFIER
데이터 유형:	MQCFST
최대 길이:	MQ_USER_ID_LENGTH
리턴됨:	Always

### ***ConnDate***

설명:	MQCONN 조작 날짜
ID:	MQCAMO_CONN_DATE
데이터 유형:	MQCFST
최대 길이:	MQ_TIME_LENGTH
리턴됨:	사용 가능할 때

### ***ConnTime***

설명:	MQCONN 조작 시간
ID:	MQCAMO_CONN_TIME
데이터 유형:	MQCFST
최대 길이:	MQ_TIME_LENGTH
리턴됨:	사용 가능할 때

### ***ConnName***

설명:	클라이언트 연결의 연결 이름
ID:	MQCACH_CONNECTION_NAME
데이터 유형:	MQCFST
최대 길이:	MQ_CONN_NAME_LENGTH
리턴됨:	사용 가능할 때

### ***ChannelName***

설명:	클라이언트 연결의 채널 이름
ID:	MQCACH_CHANNEL_NAME
데이터 유형:	MQCFST
최대 길이:	MQ_CHANNEL_NAME_LENGTH
리턴됨:	사용 가능할 때

### **DiscDate**

설명: MQDISC 조작 날짜  
ID: MQCAMO\_DISC\_DATE  
데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH  
리턴됨: 사용 가능할 때

### **DiscTime**

설명: MQDISC 조작 시간  
ID: MQCAMO\_DISC\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: 사용 가능할 때

### **DiscType**

설명: 연결 해제 유형  
ID: MQIAMO\_DISC\_TYPE  
데이터 유형: MQCFIN  
값: 가능한 값은 다음과 같습니다.  
**MQDISCONNECT\_NORMAL**  
애플리케이션의 요청으로  
**MQDISCONNECT\_IMPLICIT**  
비정상 애플리케이션 종료  
**MQDISCONNECT\_Q\_MGR**  
큐 관리자가 끊은 연결  
리턴됨: 사용 가능할 때

### **OpenCount**

설명: 열린 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다.  
[참조 노트 1](#)을 보십시오.  
ID: MQIAMO\_OPENS  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때

### **OpenFailCount**

설명: 실패한 오브젝트 열기 시도 수. 이 매개변수는 오브젝트 유형별로 색인화된 정수 목록입니다([참조 참고 1](#) 참조).  
ID: MQIAMO\_OPENS\_FAILED  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때

### **CloseCount**

설명:	닫힌 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 1</a> 을 보십시오.
ID:	MQIAMO_CLOSES
데이터 유형:	MQCFIL
리턴됨:	사용 가능할 때

### **CloseFailCount**

설명:	실패한 오브젝트 닫기 시도 수. 이 매개변수는 오브젝트 유형별로 색인화된 정수 목록입니다( <a href="#">참조 참고 1</a> 참조).
ID:	MQIAMO_CLOSES_FAILED
데이터 유형:	MQCFIL
리턴됨:	사용 가능할 때

### **PutCount**

설명:	MQPUT1 호출을 사용하여 넣은 메시지를 제외하고, 큐에 정상적으로 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_PUTS
데이터 유형:	MQCFIL
리턴됨:	사용 가능할 때

### **PutFailCount**

설명:	실패한 메시지 넣기 시도 수
ID:	MQIAMO_PUTS_FAILED
데이터 유형:	MQCFIN
리턴됨:	사용 가능할 때

### **Put1Count**

설명:	MQPUT1 호출을 사용하여 큐에 성공적으로 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_PUT1S
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **Put1FailCount**

설명:	MQPUT1 호출을 사용하여 메시지를 넣으려는 실패한 시도 수
ID:	MQIAMO_PUT1S_FAILED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **PutBytes**

설명:	지속 및 비지속 메시지에 대해 넣기 호출을 사용하여 쓴 바이트 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_PUT_BYTES
데이터 유형:	MQCFIL64
리턴됨:	사용 가능할 때

### **GetCount**

설명:	지속 및 비지속 메시지에 대한 성공한 파괴적 MQGET 호출의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_GETS
데이터 유형:	MQCFIL
리턴됨:	사용 가능할 때

### **GetFailCount**

설명:	실패한 파괴적 MQGET 호출 수
ID:	MQIAMO_GETS_FAILED
데이터 유형:	MQCFIN
리턴됨:	사용 가능할 때

### **GetBytes**

설명:	지속 및 비지속 메시지의 수신된 총 바이트 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_GET_BYTES
데이터 유형:	MQCFIL64
리턴됨:	사용 가능할 때

### **BrowseCount**

설명:	지속 및 비지속 메시지에 대한 성공한 비파괴적 MQGET 호출의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_BROWSES
데이터 유형:	MQCFIL
리턴됨:	사용 가능할 때

### **BrowseFailCount**

설명:	실패한 비파괴 MQGET 호출 수
ID:	MQIAMO_BROWSES_FAILED
데이터 유형:	MQCFIN
리턴됨:	사용 가능할 때

### **BrowseBytes**

설명:	지속 및 비지속 메시지를 찾아본 총 바이트 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
-----	--

ID: MQIAMO64\_BROWSE\_BYTES  
데이터 유형: MQCFIL64  
리턴됨: 사용 가능할 때

#### **CommitCount**

설명: 성공한 트랜잭션 수. 이 수는 연결된 애플리케이션이 내재적으로 커미트한 트랜잭션을 포함합니다. 미해결 작업이 없는 커미트 요청이 이 수에 포함됩니다.  
ID: MQIAMO\_COMMITS  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때

#### **CommitFailCount**

설명: 실패한 트랜잭션 완료 시도 수  
ID: MQIAMO\_COMMITS\_FAILED  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때

#### **BackCount**

설명: 비정상 연결 해제로 인한 암시적 백아웃을 포함하여, 처리된 백아웃 수  
ID: MQIAMO\_BACKOUTS  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때

#### **InqCount**

설명: 조회에 성공한 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.  
ID: MQIAMO\_INQS  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때

#### **InqFailCount**

설명: 실패한 오브젝트 조회 시도 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.  
ID: MQIAMO\_INQS\_FAILED  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때

#### **SetCount**

설명: 성공한 MQSET 호출 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.  
ID: MQIAMO\_SETS  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때

### **SetFailCount**

설명: 실패한 MQSET 호출 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_SETS\_FAILED

데이터 유형: MQCFIL

리턴됨: 사용 가능할 때

### **SubCountDur**

설명: 지속 가능한 구독을 작성, 대체 또는 재개한 구독 요청의 수. 조작 유형별로 색인화된 값 배열입니다.

0 = 작성된 구독 수  
1 = 변경된 구독 수  
2 = 재개한 구독 수

ID: MQIAMO\_SUBS\_DUR

데이터 유형: MQCFIL

리턴됨: 사용 가능할 때.

### **SubCountNDur**

설명: 지속 불가능한 구독을 작성, 대체 또는 재개한 구독 요청의 수. 조작 유형별로 색인화된 값 배열입니다.

0 = 작성된 구독 수  
1 = 변경된 구독 수  
2 = 재개한 구독 수

ID: MQIAMO\_SUBS\_NDUR

데이터 유형: MQCFIL

리턴됨: 사용 가능할 때.

### **SubFailCount**

설명: 실패한 구독 요청 수.

ID: MQIAMO\_SUBS\_FAILED

데이터 유형: MQCFIN

리턴됨: 사용 가능할 때.

### **UnsubCountDur**

설명: 지속 가능한 구독에 대한 성공한 구독 해제 요청 수. 조작 유형별로 색인화된 값 배열입니다.

0 - 구독이 달렸지만 제거되지 않음  
1 - 구독이 달히고 제거됨

ID: MQIAMO\_UNSUBS\_DUR

데이터 유형: MQCFIL

리턴됨: 사용 가능할 때.

### **UnsubCountNDur**

설명: 지속 가능한 구독에 대한 성공한 구독 해제 요청 수. 조작 유형별로 색인화된 값 배열입니다.  
0 - 구독이 닫혔지만 제거되지 않음  
1 - 구독이 닫히고 제거됨

ID: MQIAMO\_UNSUBS\_NDUR  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때.

### **UnsubFailCount**

설명: 실패한 구독 해제 요청 수.

ID: MQIAMO\_UNSUBS\_FAILED  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **SubRqCount**

설명: 성공한 MQSUBRQ 요청 수.

ID: MQIAMO\_SUBRQS  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **SubRqFailCount**

설명: 실패한 MQSUB 요청 수.

ID: MQIAMO\_SUBRQS\_FAILED  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **CBCount**

설명: 성공한 MQCB 요청 수. 조작 유형별로 색인화된 값 배열입니다.  
0 - 콜백이 작성 또는 변경됨  
1 - 콜백이 제거됨  
2 - 콜백이 재개됨  
3 - 콜백이 일시중단됨

ID: MQIAMO\_CBS  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **CBFailCount**

설명: 실패한 MQCB 요청 수.

ID: MQIAMO\_CBS\_FAILED  
데이터 유형: MQCFIN

리턴됨: 사용 가능할 때.

### **CtlCount**

설명: 성공한 MQCTL 요청 수. 조작 유형별로 색인화된 값 배열입니다.

- 0 - 연결이 시작됨
- 1 - 연결이 중지됨
- 2 - 연결이 재개됨
- 3 - 연결이 일시중단됨

ID: MQIAMO\_CTL5

데이터 유형: MQCFIL

리턴됨: 사용 가능할 때.

### **CtlFailCount**

설명: 실패한 MQCTL 요청 수.

ID: MQIAMO\_CTL5\_FAILED

데이터 유형: MQCFIN

리턴됨: 사용 가능할 때.

### **StatCount**

설명: 성공한 MQSTAT 요청 수.

ID: MQIAMO\_STATS.

데이터 유형: MQCFIN

리턴됨: 사용 가능할 때.

### **StatFailCount**

설명: 실패한 MQSTAT 요청 수.

ID: MQIAMO\_STATS\_FAILED

데이터 유형: MQCFIN

리턴됨: 사용 가능할 때.

### **PutTopicCount**

설명: MQPUT1 호출을 사용하여 넣은 메시지를 제외하고, 토픽에 정상적으로 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

참고: 토픽으로 분석되며 큐 알리언스를 사용하여 넣은 메시지가 이 값에 포함됩니다.

ID: MQIAMO\_TOPIC\_PUTS

데이터 유형: MQCFIL

리턴됨: 사용 가능할 때.

### **PutTopicFailCount**

설명: 토픽에 메시지를 넣으려는 실패한 시도 수.

ID: MQIAMO\_TOPIC\_PUTS\_FAILED

데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **Put1TopicCount**

설명: MQPUT1 호출을 사용하여 토픽에 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.  
참고: 토픽으로 분석되며 큐 알리어스를 사용하여 넣은 메시지가 이 값에 포함됩니다.

ID: MQIAMO\_TOPIC\_PUT1S  
데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때.

### **Put1TopicFailCount**

설명: MQPUT1 호출을 사용하여 토픽에 메시지를 넣으려는 실패한 시도 수.  
ID: MQIAMO\_TOPIC\_PUT1S\_FAILED  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **PutTopicBytes**

설명: 발행 조작으로 분석된 지속 및 비지속 메시지에 대해 넣기 호출을 사용하여 쓴 바이트 수. 이 수는 구독자에게 전달된 결과적인 바이트 수가 아니라, 애플리케이션이 넣은 바이트 수입니다. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.  
ID: MQIAMO64\_TOPIC\_PUT\_BYTES  
데이터 유형: MQCFIL64  
리턴됨: 사용 가능할 때.

## **큐 회계 메시지 데이터**

이 페이지를 사용하여 큐 회계 메시지의 구조를 봅니다.

메시지 이름:	큐 회계 메시지.
플랫폼:	All, except WebSphere MQ for z/OS.
시스템 큐:	SYSTEM.ADMIN.ACCOUNTING.QUEUE.

### **QueueManager**

설명: 큐 매니저의 이름  
ID: MQCA\_Q\_MGR\_NAME  
데이터 유형: MQCFST  
최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH  
리턴됨: Always

### **IntervalStartDate**

설명: 모니터링 기간의 시작 날짜  
ID: MQCAMO\_START\_DATE

데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH  
리턴됨: Always

### ***IntervalStartTime***

설명: 모니터링 기간의 시작 시간  
ID: MQCAMO\_START\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: Always

### ***IntervalEndDate***

설명: 모니터링 기간의 종료 날짜  
ID: MQCAMO\_END\_DATE  
데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH  
리턴됨: Always

### ***IntervalEndTime***

설명: 모니터링 기간의 종료 시간  
ID: MQCAMO\_END\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: Always

### ***CommandLevel***

설명: 큐 관리자 명령 레벨  
ID: MQIA\_COMMAND\_LEVEL  
데이터 유형: MQCFIN  
리턴됨: Always

### ***ConnectionId***

설명: WebSphere MQ 연결의 연결 ID  
ID: MQBACF\_CONNECTION\_ID  
데이터 유형: MQCFBS  
최대 길이: MQ\_CONNECTION\_ID\_LENGTH  
리턴됨: Always

### ***SeqNumber***

설명: 순서 번호. 이 값은 장기 실행 연결에 대한 각 후속 레코드마다 증분됩니다.  
ID: MQIACF\_SEQUENCE\_NUMBER

데이터 유형: MQCFIN  
리턴됨: Always

### ***ApplicationName***

설명: 애플리케이션의 이름. 이 필드의 콘텐츠는 메시지 디스크립터의 PutApplName 필드 콘텐츠에 해당합니다.  
ID: MQCACF\_APPL\_NAME  
데이터 유형: MQCFST  
최대 길이: MQ\_APPL\_NAME\_LENGTH  
리턴됨: Always

### ***ApplicationPid***

설명: 애플리케이션의 운영 체제 프로세스 ID  
ID: MQIACF\_PROCESS\_ID  
데이터 유형: MQCFIN  
리턴됨: Always

### ***ApplicationTid***

설명: 애플리케이션 내에서 연결의 WebSphere MQ 스레드 ID입니다.  
ID: MQIACF\_THREAD\_ID  
데이터 유형: MQCFIN  
리턴됨: Always

### ***UserId***

설명: 애플리케이션의 사용자 ID 컨텍스트  
ID: MQCACF\_USER\_IDENTIFIER  
데이터 유형: MQCFST  
최대 길이: MQ\_USER\_ID\_LENGTH  
리턴됨: Always

### ***ObjectCount***

설명: 회계 데이터가 기록된 간격에 액세스한 큐의 수. 이 값은 메시지에 포함된 *QAccountingData* PCF 그룹 수로 설정됩니다.  
ID: MQIAMO\_OBJECT\_COUNT  
데이터 유형: MQCFIN  
리턴됨: Always

### ***QAccountingData***

설명: 큐에 대한 회계 세부사항을 지정하는 그룹화된 매개변수.  
ID: MQGACF\_Q\_ACCOUNTING\_DATA  
데이터 유형: MQCFGR

그룹의 매개변수:

- QName*
- CreateDate*
- CreateTime*
- QType*
- QDefinitionType*
- OpenCount*
- OpenDate*
- OpenTime*
- CloseDate*
- CloseTime*
- PutCount*
- PutFailCount*
- Put1Count*
- Put1FailCount*
- PutBytes*
- PutMinBytes*
- PutMaxBytes*
- GetCount*
- GetFailCount*
- GetBytes*
- GetMinBytes*
- GetMaxBytes*
- BrowseCount*
- BrowseFailCount*
- BrowseBytes*
- BrowseMinBytes*
- BrowseMaxBytes*
- TimeOnQMin*
- TimeOnQAvg*
- TimeOnQMax*

리턴됨: Always

**QName**

설명: 큐의 이름.  
 ID: MQCA\_Q\_NAME  
 데이터 유형: MQCFST  
 PCF 그룹에 포함: *QAccountingData*  
 최대 길이: MQ\_Q\_NAME\_LENGTH  
 리턴됨: 사용 가능할 때

**CreateDate**

설명: 큐가 작성된 날짜  
 ID: MQCA\_CREATION\_DATE  
 데이터 유형: MQCFST  
 PCF 그룹에 포함: *QAccountingData*  
 최대 길이: MQ\_DATE\_LENGTH

리턴됨: 사용 가능할 때

### **CreateTime**

설명: 큐가 작성된 시간  
ID: MQCA\_CREATION\_TIME  
데이터 유형: MQCFST  
PCF 그룹에 포함: *QAccountingData*  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: 사용 가능할 때

### **QType**

설명: 큐의 유형  
ID: MQIA\_Q\_TYPE  
데이터 유형: MQCFIN  
PCF 그룹에 포함: *QAccountingData*  
가치: MQQT\_LOCAL  
리턴됨: 사용 가능할 때

### **QDefinitionType**

설명: 큐 정의 유형  
ID: MQIA\_DEFINITION\_TYPE  
데이터 유형: MQCFIN  
PCF 그룹에 포함: *QAccountingData*  
값: 가능한 값은 다음과 같습니다.

**MQQDT\_PREDEFINED**  
**MQQDT\_PERMANENT\_DYNAMIC**  
**MQQDT\_TEMPORARY\_DYNAMIC**

리턴됨: 사용 가능할 때

### **OpenCount**

설명: 애플리케이션이 이 간격에 이 큐를 연 횟수  
ID: MQIAMO\_OPENS  
데이터 유형: MQCFIL  
PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **OpenDate**

설명: 이 레코딩 간격에 처음으로 큐를 연 날짜. 이 간격이 시작할 때 큐가 이미 열려 있으면 이 값은 큐가 원래 열린 날짜를 반영합니다.  
ID: MQCAMO\_OPEN\_DATE  
데이터 유형: MQCFST

PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **OpenTime**

설명: 이 레코딩 간격에 처음으로 큐를 연 시간. 이 간격이 시작할 때 큐가 이미 열려 있으면 이 값은 큐가 원래 열린 시간을 반영합니다.

ID: MQCAMO\_OPEN\_TIME  
데이터 유형: MQCFST  
PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **CloseDate**

설명: 이 레코딩 간격에 마지막으로 큐를 닫은 날짜. 큐가 여전히 열려 있으면 값이 리턴되지 않습니다.

ID: MQCAMO\_CLOSE\_DATE  
데이터 유형: MQCFST  
PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **CloseTime**

설명: 이 레코딩 간격에 마지막으로 큐를 닫은 시간. 큐가 여전히 열려 있으면 값이 리턴되지 않습니다.

ID: MQCAMO\_CLOSE\_TIME  
데이터 유형: MQCFST  
PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **PutCount**

설명: MQPUT1 호출을 제외하고, 큐에 넣은 지속 및 비지속 메시지의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUTS  
데이터 유형: MQCFIL  
PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **PutFailCount**

설명: MQPUT1 호출을 제외한, 실패한 메시지 넣기 시도 수

ID: MQIAMO\_PUTS\_FAILED  
데이터 유형: MQCFIN  
PCF 그룹에 포함: *QAccountingData*  
리턴됨: 사용 가능할 때

### **Put1Count**

설명: MQPUT1 호출을 사용하여 큐에 성공적으로 넣은 지속 및 비지속 메시지 수. 이 매개 변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUT1S

데이터 유형: MQCFIL

PCF 그룹에 포함: *QAccountingData*

리턴됨: 사용 가능할 때

### **Put1FailCount**

설명: MQPUT1 호출을 사용하여 메시지를 넣으려는 실패한 시도 수

ID: MQIAMO\_PUT1S\_FAILED

데이터 유형: MQCFIN

PCF 그룹에 포함: *QAccountingData*

리턴됨: 사용 가능할 때

### **PutBytes**

설명: 지속 및 비지속 메시지의 넣은 총 바이트 수. 이 매개 변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO64\_PUT\_BYTES

데이터 유형: MQCFIL64

PCF 그룹에 포함: *QAccountingData*

리턴됨: 사용 가능할 때

### **PutMinBytes**

설명: 큐에 있는 지속 및 비지속 메시지의 최소 크기. 이 매개 변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUT\_MIN\_BYTES

데이터 유형: MQCFIL

PCF 그룹에 포함: *QAccountingData*

리턴됨: 사용 가능할 때

### **PutMaxBytes**

설명: 큐에 있는 지속 및 비지속 메시지의 최대 크기. 이 매개 변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUT\_MAX\_BYTES

데이터 유형: MQCFIL

PCF 그룹에 포함: *QAccountingData*

리턴됨: 사용 가능할 때

### **GeneratedMsgCount**

설명:	생성된 메시지의 수. 생성되는 메시지는 다음과 같습니다. <ul style="list-style-type: none"><li>• 큐 용량 높음 이벤트</li><li>• 큐 용량 낮음 이벤트</li></ul>
ID:	MQIAMO_GENERATED_MSGS
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **GetCount**

설명:	지속 및 비지속 메시지에 대한 성공한 파괴적 MQGET 호출의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_GETS
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **GetFailCount**

설명:	실패한 파괴적 MQGET 호출 수
ID:	MQIAMO_GETS_FAILED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **GetBytes**

설명:	지속 및 비지속 메시지에 대한 파괴적 MQGET 호출의 읽은 바이트 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_GET_BYTES
데이터 유형:	MQCFIL64
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **GetMinBytes**

설명:	큐에서 검색한 지속 및 비지속 메시지의 최소 크기. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_GET_MIN_BYTES
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **GetMaxBytes**

설명:	큐에서 검색한 지속 및 비지속 메시지의 최대 크기. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_GET_MAX_BYTES
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **BrowseCount**

설명:	지속 및 비지속 메시지에 대한 성공한 비파괴적 MQGET 호출의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_BROWSES
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **BrowseFailCount**

설명:	실패한 비파괴 MQGET 호출 수
ID:	MQIAMO_BROWSES_FAILED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **BrowseBytes**

설명:	지속 메시지를 리턴한 비파괴 MQGET 호출의 읽은 바이트 수.
ID:	MQIAMO64_BROWSE_BYTES
데이터 유형:	MQCFIL64
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **BrowseMinBytes**

설명:	큐에서 찾아본 지속 및 비지속 메시지의 최소 크기. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_BROWSE_MIN_BYTES
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QAccountingData</i>
리턴됨:	사용 가능할 때

### **BrowseMaxBytes**

설명:	큐에서 찾아본 지속 및 비지속 메시지의 최대 크기. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_BROWSE_MAX_BYTES

데이터 유형: MQCFIL  
PCF 그룹에 포함: QAccountingData  
리턴됨: 사용 가능할 때

### **CBCount**

설명: 성공한 MQCB 요청 수. 조작 유형별로 색인화된 값 배열입니다.  
0 - 콜백이 작성 또는 변경됨  
1 - 콜백이 제거됨  
2 - 콜백이 재개됨  
3 - 콜백이 일시중단됨

ID: MQIAMO\_CBS  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **CBFailCount**

설명: 실패한 MQCB 요청 수.  
ID: MQIAMO\_CBS\_FAILED  
데이터 유형: MQCFIN  
리턴됨: 사용 가능할 때.

### **TimeOnQMin**

설명: 파괴적 검색 이전에 지속 및 비지속 메시지가 큐에 머문 최단 시간(마이크로초). 동기 점 아래에서 메시지가 검색되도록 이 값은 가져오기 조작이 커밋되기 전의 시간을 포함하지 않습니다. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO64\_Q\_TIME\_MIN  
데이터 유형: MQCFIL64  
PCF 그룹에 포함: QAccountingData  
리턴됨: 사용 가능할 때

### **TimeOnQAvg**

설명: 파괴적 검색 이전에 지속 및 비지속 메시지가 큐에 머문 평균 시간(마이크로초). 동기 점 아래에서 메시지가 검색되도록 이 값은 가져오기 조작이 커밋되기 전의 시간을 포함하지 않습니다. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO64\_Q\_TIME\_AVG  
데이터 유형: MQCFIL64  
PCF 그룹에 포함: QAccountingData  
리턴됨: 사용 가능할 때

### ***TimeOnQMax***

설명:	파괴적 검색 이전에 지속 및 비지속 메시지가 큐에 머문 최장 시간(마이크로초). 동기 점 아래에서 메시지가 검색되도록 이 값은 가져오기 조작이 커밋되기 전의 시간을 포함하지 않습니다. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_Q_TIME_MAX
데이터 유형:	MQCFIL64
PCF 그룹에 포함:	QAccountingData
리턴됨:	사용 가능할 때

### **MQI 통계 메시지 데이터**

이 페이지를 사용하여 MQI 통계 메시지의 구조를 봅니다.

메시지 이름:	MQI 통계 메시지.
플랫폼:	All, except WebSphere MQ for z/OS.
시스템 큐:	SYSTEM.ADMIN.STATISTICS.QUEUE.

### ***QueueManager***

설명:	큐 관리자의 이름.
ID:	MQCA_Q_MGR_NAME.
데이터 유형:	MQCFST.
최대 길이:	MQ_Q_MGR_NAME_LENGTH.
리턴됨:	항상.

### ***IntervalStartDate***

설명:	모니터링 기간의 시작 날짜.
ID:	MQCAMO_START_DATE.
데이터 유형:	MQCFST.
최대 길이:	MQ_DATE_LENGTH
리턴됨:	항상.

### ***IntervalStartTime***

설명:	모니터링 기간의 시작 시간.
ID:	MQCAMO_START_TIME.
데이터 유형:	MQCFST.
최대 길이:	MQ_TIME_LENGTH
리턴됨:	항상.

### ***IntervalEndDate***

설명:	모니터링 기간의 종료 날짜.
ID:	MQCAMO_END_DATE.
데이터 유형:	MQCFST.

최대 길이: MQ\_DATE\_LENGTH  
리턴됨: 항상.

### ***IntervalEndTime***

설명: 모니터링 기간의 종료 시간.  
ID: MQCAMO\_END\_TIME.  
데이터 유형: MQCFST.  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: 항상.

### ***CommandLevel***

설명: 큐 관리자 명령 레벨.  
ID: MQIA\_COMMAND\_LEVEL.  
데이터 유형: MQCFIN.  
리턴됨: 항상.

### ***ConnCount***

설명: 큐 관리자에 대한 연결 수.  
ID: MQIAMO\_CONNS.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### ***ConnFailCount***

설명: 실패한 연결 시도 수.  
ID: MQIAMO\_CONNS\_FAILED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### ***ConnsMax***

설명: 레코딩 간격의 최대 동시 연결 수.  
ID: MQIAMO\_CONNS\_MAX.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### ***DiscCount***

설명: 큐 관리자에 대한 연결 해제 수. 다음 상수에 따라 색인화된 정수 배열입니다.  
• MQDISCONNECT\_NORMAL  
• MQDISCONNECT\_IMPLICIT  
• MQDISCONNECT\_Q\_MGR  
ID: MQIAMO\_DISCS.  
데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **OpenCount**

설명: 열린 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_OPENS.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **OpenFailCount**

설명: 실패한 오브젝트 열기 시도 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_OPENS\_FAILED.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **CloseCount**

설명: 닫힌 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_CLOSES.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **CloseFailCount**

설명: 성공한 오브젝트 닫기 시도 수입니다. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_CLOSES\_FAILED.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **InqCount**

설명: 조회한 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_INQS.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **InqFailCount**

설명: 실패한 오브젝트 조회 시도 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_INQS\_FAILED.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **SetCount**

설명: 업데이트(SET)된 오브젝트 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_SETS.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **SetFailCount**

설명: 실패한 SET 시도 수. 이 매개변수는 오브젝트 유형에 의해 색인화되는 정수 목록입니다. [참조 노트 1](#)을 보십시오.

ID: MQIAMO\_SETS\_FAILED.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **PutCount**

설명: MQPUT1 호출을 제외하고, 큐에 넣은 지속 및 비지속 메시지의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUTS.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **PutFailCount**

설명: 실패한 메시지 넣기 시도 수.

ID: MQIAMO\_PUTS\_FAILED.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **Put1Count**

설명: MQPUT1 요청을 사용하여 큐에 성공적으로 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUT1S.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **Put1FailCount**

설명: MQPUT1 요청을 사용하여 지속 및 비지속 메시지를 큐에 넣으려는 실패한 시도 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

ID: MQIAMO\_PUT1S\_FAILED.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **PutBytes**

설명:	넣기 요청을 사용하여 기록된 지속 및 비지속 메시지의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_PUT_BYTES.
데이터 유형:	MQCFIL64.
리턴됨:	사용 가능할 때.

### **GetCount**

설명:	지속 및 비지속 메시지에 대한 성공한 파괴적 가져오기 요청의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_GETS.
데이터 유형:	MQCFIL.
리턴됨:	사용 가능할 때.

### **GetFailCount**

설명:	실패한 파괴적 가져오기 요청의 수.
ID:	MQIAMO_GETS_FAILED.
데이터 유형:	MQCFIN.
리턴됨:	사용 가능할 때.

### **GetBytes**

설명:	지속 및 비지속 메시지에 대한 파괴적 가져오기 요청의 읽은 바이트 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_GET_BYTES.
데이터 유형:	MQCFIL64.
리턴됨:	사용 가능할 때.

### **BrowseCount**

설명:	지속 및 비지속 메시지에 대한 성공한 비파괴 가져오기 요청의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO_BROWSES.
데이터 유형:	MQCFIL.
리턴됨:	사용 가능할 때.

### **BrowseFailCount**

설명:	실패한 비파괴 가져오기 요청의 수.
ID:	MQIAMO_BROWSES_FAILED.
데이터 유형:	MQCFIN.
리턴됨:	사용 가능할 때.

### **BrowseBytes**

설명:	지속 및 비지속 메시지에 대한 비파괴 가져오기 요청의 읽은 바이트 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
-----	---

ID: MQIAMO64\_BROWSE\_BYTES.  
데이터 유형: MQCFIL64.  
리턴됨: 사용 가능할 때.

#### **CommitCount**

설명: 완료한 트랜잭션 수. 이 수는 애플리케이션 연결 해제를 통해 내재적으로 커밋된 트랜잭션 및 미해결 작업이 없는 커밋 요청을 포함합니다.  
ID: MQIAMO\_COMMITS.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

#### **CommitFailCount**

설명: 실패한 트랜잭션 완료 시도 수.  
ID: MQIAMO\_COMMITS\_FAILED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

#### **BackCount**

설명: 비정상 연결 해제로 인한 암시적 백아웃을 포함하여, 처리된 백아웃 수.  
ID: MQIAMO\_BACKOUTS.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

#### **ExpiredMsgCount**

설명: 검색 가능하게 되기 전에 만기되어서 제거된 지속 및 비지속 메시지의 수.  
ID: MQIAMO\_MSGS\_EXPIRED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

#### **PurgeCount**

설명: 큐가 지워진 횟수.  
ID: MQIAMO\_MSGS\_PURGED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

#### **SubCountDur**

설명: 지속 가능한 구독을 작성, 변경 또는 재개한 구독 요청의 수. 조작 유형별로 색인화된 값 배열입니다.  
0 = 작성된 구독 수  
1 = 변경된 구독 수  
2 = 재개한 구독 수  
ID: MQIAMO\_SUBS\_DUR.

데이터 유형: MQCFIL  
리턴됨: 사용 가능할 때.

#### **SubCountNDur**

설명: 지속 불가능한 구독을 작성, 변경 또는 재개한 구독 요청의 수. 조작 유형별로 색인화된 값 배열입니다.

0 = 작성된 구독 수  
1 = 변경된 구독 수  
2 = 재개한 구독 수

ID: MQIAMO\_SUBS\_NDUR.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

#### **SubFailCount**

설명: 실패한 구독 요청 수.

ID: MQIAMO\_SUBS\_FAILED.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

#### **UnsubCountDur**

설명: 지속 가능한 구독에 대한 성공한 구독 해제 요청 수. 조작 유형별로 색인화된 값 배열입니다.

0 - 구독이 닫혔지만 제거되지 않음  
1 - 구독이 닫히고 제거됨

ID: MQIAMO\_UNSUBS\_DUR.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

#### **UnsubCountNDur**

설명: 지속 불가능한 구독에 대한 성공한 구독 해제 요청 수. 조작 유형별로 색인화된 값 배열입니다.

0 - 구독이 닫혔지만 제거되지 않음  
1 - 구독이 닫히고 제거됨

ID: MQIAMO\_UNSUBS\_NDUR.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

#### **UnsubFailCount**

설명: 실패한 구독 해제 요청 수.

ID: MQIAMO\_UNSUBS\_FAILED.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **SubRqCount**

설명: 성공한 MQSUBRQ 요청 수.

ID: MQIAMO\_SUBRQS

데이터 유형: MQCFIN

리턴됨: 사용 가능할 때.

### **SubRqFailCount**

설명: 실패한 MQSUBRQ 요청 수.

ID: MQIAMO\_SUBRQS\_FAILED.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **CBCount**

설명: 성공한 MQCB 요청 수. 조작 유형별로 색인화된 값 배열입니다.

0 - 콜백이 작성 또는 변경됨

1 - 콜백이 제거됨

2 - 콜백이 재개됨

3 - 콜백이 일시중단됨

ID: MQIAMO\_CBS.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **CBFailCount**

설명: 실패한 MQCB 요청 수.

ID: MQIAMO\_CBS\_FAILED.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **CtlCount**

설명: 성공한 MQCTL 요청 수. 조작 유형에 의해 색인화된 값의 배열입니다.

0 - 연결이 시작됨

1 - 연결이 중지됨

2 - 연결이 재개됨

3 - 연결이 일시중단됨

ID: MQIAMO\_CTL.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### ***CtlFailCount***

설명: 실패한 MQCTL 요청 수.  
ID: MQIAMO\_CTL\_S\_FAILED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### ***StatCount***

설명: 성공한 MQSTAT 요청 수.  
ID: MQIAMO\_STATS.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### ***StatFailCount***

설명: 실패한 MQSTAT 요청 수.  
ID: MQIAMO\_STATS\_FAILED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### ***SubCountDurHighWater***

설명: 시간 간격 중의 지속 가능한 구독 수에 대한 고수위 표시. SUBTYPE별로 색인화된 값 배열입니다.  
0 - 시스템의 모든 지속 가능 구독에 대한 고수위 표시  
1 - 지속 가능 애플리케이션 구독(MQSUBTYPE\_API)에 대한 고수위 표시  
2 - 지속 가능 관리 구독(MQSUBTYPE\_ADMIN)에 대한 고수위 표시  
3 - 지속 가능 프록시 구독(MQSUBTYPE\_PROXY)에 대한 고수위 표시  
ID: MQIAMO\_SUB\_DUR\_HIGHWATER  
데이터 유형: MQCFIL.  
리턴됨: 사용 가능할 때.

### ***SubCountDurLowWater***

설명: 시간 간격 중의 지속 가능한 구독 수에 대한 저수위 표시. SUBTYPE별로 색인화된 값 배열입니다.  
0 - 시스템의 모든 지속 가능 구독에 대한 저수위 표시  
1 - 지속 가능 애플리케이션 구독(MQSUBTYPE\_API)에 대한 저수위 표시  
2 - 지속 가능 관리 구독(MQSUBTYPE\_ADMIN)에 대한 저수위 표시  
3 - 지속 가능 프록시 구독(MQSUBTYPE\_PROXY)에 대한 저수위 표시  
ID: MQIAMO\_SUB\_DUR\_LOWWATER  
데이터 유형: MQCFIL.  
리턴됨: 사용 가능할 때.

### **SubCountNDurHighWater**

설명: 시간 간격 중의 지속 불가능한 구독 수에 대한 고수위 표시. SUBTYPE별로 색인화된 값 배열입니다.

- 0 - 시스템의 모든 지속 불가능 구독에 대한 고수위 표시
- 1 - 지속 불가능 애플리케이션 구독(MQSUBTYPE\_API)에 대한 고수위 표시
- 2 - 지속 불가능 관리 구독(MQSUBTYPE\_ADMIN)에 대한 고수위 표시
- 3 - 지속 불가능 프록시 구독(MQSUBTYPE\_PROXY)에 대한 고수위 표시

ID: MQIAMO\_SUB\_NDUR\_HIGHWATER

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **SubCountNDurLowWater**

설명: 시간 간격 중의 지속 불가능한 구독 수에 대한 저수위 표시. SUBTYPE별로 색인화된 값 배열입니다.

- 0 - 시스템의 모든 지속 불가능 구독에 대한 저수위 표시
- 1 - 지속 불가능 애플리케이션 구독(MQSUBTYPE\_API)에 대한 저수위 표시
- 2 - 지속 불가능 관리 구독(MQSUBTYPE\_ADMIN)에 대한 저수위 표시
- 3 - 지속 불가능 프록시 구독(MQSUBTYPE\_PROXY)에 대한 저수위 표시

ID: MQIAMO\_SUB\_NDUR\_LOWWATER

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **PutTopicCount**

설명: MQPUT1 호출을 사용하여 넣은 메시지를 제외하고, 토픽에 정상적으로 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

참고: 토픽으로 분석되며 큐 알리어스를 사용하여 넣은 메시지가 이 값에 포함됩니다.

ID: MQIAMO\_TOPIC\_PUTS.

데이터 유형: MQCFIL.

리턴됨: 사용 가능할 때.

### **PutTopicFailCount**

설명: 토픽에 메시지를 넣으려는 실패한 시도 수.

ID: MQIAMO\_TOPIC\_PUTS\_FAILED.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **Put1TopicCount**

설명: MQPUT1 호출을 사용하여 토픽에 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. [참조 노트 2](#)를 보십시오.

참고: 토픽으로 분석되며 큐 알리어스를 사용하여 넣은 메시지가 이 값에 포함됩니다.

ID: MQIAMO\_TOPIC\_PUT1S.  
데이터 유형: MQCFIL.  
리턴됨: 사용 가능할 때.

### **Put1TopicFailCount**

설명: MQPUT1 호출을 사용하여 토픽에 메시지를 넣으려는 실패한 시도 수.  
ID: MQIAMO\_TOPIC\_PUT1S\_FAILED.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### **PutTopicBytes**

설명: 발행 조작으로 분석된 지속 및 비지속 메시지에 대해 넣기 호출을 사용하여 쓴 바이트 수. 이 수는 구독자에게 전달된 결과적인 바이트 수가 아니라, 애플리케이션이 넣은 바이트 수입니다(이 값에 대해 PublishMsgBytes 참조). 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. 참조 노트 2를 보십시오.  
ID: MQIAMO64\_TOPIC\_PUT\_BYTES.  
데이터 유형: MQCFIL64.  
리턴됨: 사용 가능할 때.

### **PublishMsgCount**

설명: 시간 간격 중 구독에 전달된 메시지의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. 참조 노트 2를 보십시오.  
ID: MQIAMO64\_PUBLISH\_MSG\_COUNT  
데이터 유형: MQCFIL.  
리턴됨: 사용 가능할 때.

### **PublishMsgBytes**

설명: 시간 간격 중 구독에 전달된 바이트의 수. 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. 참조 노트 2를 보십시오.  
ID: MQIAMO64\_PUBLISH\_MSG\_BYTES  
데이터 유형: MQCFIL64.  
리턴됨: 사용 가능할 때.

## **큐 통계 메시지 데이터**

이 페이지를 사용하여 큐 통계 메시지의 구조를 봅니다.

메시지 이름:	큐 통계 메시지.
플랫폼:	All, except WebSphere MQ for z/OS.
시스템 큐:	SYSTEM.ADMIN.STATISTICS.QUEUE.

### **QueueManager**

설명: 큐 관리자의 이름  
ID: MQCA\_Q\_MGR\_NAME

데이터 유형: MQCFST  
최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH  
리턴됨: Always

#### ***IntervalStartDate***

설명: 모니터링 기간의 시작 날짜  
ID: MQCAMO\_START\_DATE  
데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH  
리턴됨: Always

#### ***IntervalStartTime***

설명: 모니터링 기간의 시작 시간  
ID: MQCAMO\_START\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: Always

#### ***IntervalEndDate***

설명: 모니터링 기간의 종료 날짜  
ID: MQCAMO\_END\_DATE  
데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH  
리턴됨: Always

#### ***IntervalEndTime***

설명: 모니터링 기간의 종료 시간  
ID: MQCAMO\_END\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH  
리턴됨: Always

#### ***CommandLevel***

설명: 큐 관리자 명령 레벨  
ID: MQIA\_COMMAND\_LEVEL  
데이터 유형: MQCFIN  
리턴됨: Always

#### ***ObjectCount***

설명: 통계 데이터가 기록된 간격에 액세스한 큐 오브젝트 수. 이 값은 메시지에 포함된 QStatisticsData PCF 그룹 수로 설정됩니다.  
ID: MQIAMO\_OBJECT\_COUNT

데이터 유형: MQCFIN  
리턴됨: Always

### ***QStatisticsData***

설명: 큐에 대한 통계 세부사항을 지정하는 그룹화된 매개변수.

ID: MQGACF\_Q\_STATISTICS\_DATA

데이터 유형: MQCFGR

그룹의 매개변수:

- QName*
- CreateDate*
- CreateTime*
- QType*
- QDefinitionType*
- QMinDepth*
- QMaxDepth*
- AvgTimeOnQ*
- PutCount*
- PutFailCount*
- Put1Count*
- Put1FailCount*
- PutBytes*
- GetCount*
- GetFailCount*
- GetBytes*
- BrowseCount*
- BrowseFailCount*
- BrowseBytes*
- NonQueuedMsgCount*
- ExpiredMsgCount*
- PurgeCount*

리턴됨: Always

### ***QName***

설명: 큐의 이름.

ID: MQCA\_Q\_NAME

데이터 유형: MQCFST

최대 길이: MQ\_Q\_NAME\_LENGTH

리턴됨: Always

### ***CreateDate***

설명: 큐가 작성된 날짜

ID: MQCA\_CREATION\_DATE

데이터 유형: MQCFST

최대 길이: MQ\_DATE\_LENGTH

리턴됨: Always

### **CreateTime**

설명:	큐가 작성된 시간
ID:	MQCA_CREATION_TIME
데이터 유형:	MQCFST
최대 길이:	MQ_TIME_LENGTH
리턴됨:	Always

### **QType**

설명:	큐의 유형
ID:	MQIA_Q_TYPE
데이터 유형:	MQCFIN
가치:	MQOT_LOCAL
리턴됨:	Always

### **QDefinitionType**

설명:	큐 정의 유형
ID:	MQIA_DEFINITION_TYPE
데이터 유형:	MQCFIN
값:	가능한 값은 다음과 같습니다. <ul style="list-style-type: none"><li>• MQQDT_PREDEFINED</li><li>• MQQDT_PERMANENT_DYNAMIC</li><li>• MQQDT_TEMPORARY_DYNAMIC</li></ul>
리턴됨:	사용 가능할 때

### **QMinDepth**

설명:	모니터링 기간의 최소 큐 용량
ID:	MQIAMO_Q_MIN_DEPTH
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **QMaxDepth**

설명:	모니터링 기간의 최대 큐 용량
ID:	MQIAMO_Q_MAX_DEPTH
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **AvgTimeOnQ**

설명:	모니터링 기간 중 큐에서 파괴적으로 검색한 메시지의 평균 대기 시간(마이크로초). 이 매개변수는 지속성 값에 의해 색인화되는 정수 목록입니다. <a href="#">참조 노트 2</a> 를 보십시오.
ID:	MQIAMO64_AVG_Q_TIME
데이터 유형:	MQCFIL64
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **PutCount**

설명:	MQPUT1 요청을 제외하고, 큐에 넣은 지속 및 비지속 메시지의 수. 이 매개변수는 지속 값별로 색인화된 정수 목록입니다. <a href="#">참조 참고 2</a> 를 참조하십시오.
ID:	MQIAMO_PUTS
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **PutFailCount**

설명:	큐에 메시지를 넣으려는 실패한 시도 수.
ID:	MQIAMO_PUTS_FAILED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **Put1Count**

설명:	MQPUT1 호출을 사용하여 큐에 성공적으로 넣은 지속 및 비지속 메시지 수. 이 매개변수는 지속 값별로 색인화된 정수 목록입니다. <a href="#">참조 참고 2</a> 를 참조하십시오.
ID:	MQIAMO_PUT1S
데이터 유형:	MQCFIL
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **Put1FailCount**

설명:	MQPUT1 호출을 사용하여 메시지를 넣으려는 실패한 시도 수
ID:	MQIAMO_PUT1S_FAILED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **PutBytes**

설명:	큐에 대한 넣기 요청의 기록된 바이트 수
ID:	MQIAMO64_PUT_BYTES

데이터 유형: MQCFIL64  
PCF 그룹에 포함: *QStatisticsData*  
리턴됨: 사용 가능할 때

### **GetCount**

설명: 지속 및 비지속 메시지에 대한 성공한 파괴적 가져오기 요청의 수. 이 매개변수는 지속 값별로 색인화된 정수 목록입니다. [참조 참고 2](#)를 참조하십시오.

ID: MQIAMO\_GETS  
데이터 유형: MQCFIL  
PCF 그룹에 포함: *QStatisticsData*  
리턴됨: 사용 가능할 때

### **GetFailCount**

설명: 실패한 파괴적 가져오기 요청의 수

ID: MQIAMO\_GETS\_FAILED  
데이터 유형: MQCFIN  
PCF 그룹에 포함: *QStatisticsData*  
리턴됨: 사용 가능할 때

### **GetBytes**

설명: 지속 및 비지속 메시지에 대한 파괴적 넣기 요청의 읽은 바이트 수. 이 매개변수는 지속 값별로 색인화된 정수 목록입니다. [참조 참고 2](#)를 참조하십시오.

ID: MQIAMO64\_GET\_BYTES  
데이터 유형: MQCFIL64  
PCF 그룹에 포함: *QStatisticsData*  
리턴됨: 사용 가능할 때

### **BrowseCount**

설명: 지속 및 비지속 메시지에 대한 성공한 비파괴 가져오기 요청의 수. 이 매개변수는 지속 값별로 색인화된 정수 목록입니다. [참조 참고 2](#)를 참조하십시오.

ID: MQIAMO\_BROWSES  
데이터 유형: MQCFIL  
PCF 그룹에 포함: *QStatisticsData*  
리턴됨: 사용 가능할 때

### **BrowseFailCount**

설명: 실패한 비파괴 가져오기 요청의 수

ID: MQIAMO\_BROWSES\_FAILED  
데이터 유형: MQCFIN  
PCF 그룹에 포함: *QStatisticsData*  
리턴됨: 사용 가능할 때

### **BrowseBytes**

설명:	지속 및 비지속 메시지에 대한 비파괴 가져오기 요청의 읽은 바이트 수. 이 매개변수는 지속 값별로 색인화된 정수 목록입니다. <a href="#">참조 참고 2</a> 를 참조하십시오.
ID:	MQIAMO64_BROWSE_BYTES
데이터 유형:	MQCFIL64
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **NonQueuedMsgCount**

설명:	큐를 무시하고 대기 중인 애플리케이션에 직접 전송된 메시지의 수. 큐 무시는 특정 상황에만 발생할 수 있습니다. 이 수는 애플리케이션이 대기한 횟수가 아니라 WebSphere MQ가 큐를 몇 번이나 생략할 수 있었는지 표시합니다.
ID:	MQIAMO_MSGS_NOT_QUEUED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **ExpiredMsgCount**

설명:	검색 가능하게 되기 전에 만기되어서 제거된 지속 및 비지속 메시지의 수.
ID:	MQIAMO_MSGS_EXPIRED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **PurgeCount**

설명:	제거된 메시지의 수.
ID:	MQIAMO_MSGS_PURGED
데이터 유형:	MQCFIN
PCF 그룹에 포함:	<i>QStatisticsData</i>
리턴됨:	사용 가능할 때

### **CBCCount**

설명:	성공한 MQCB 요청 수. 조작 유형별로 색인화된 값 배열입니다. 0 - 콜백이 작성 또는 변경됨 1 - 콜백이 제거됨 2 - 콜백이 재개됨 3 - 콜백이 일시중단됨
ID:	MQIAMO_CBS
데이터 유형:	MQCFIN
리턴됨:	사용 가능할 때.

### ***CBFailCount***

설명:	실패한 MQCB 요청 수.
ID:	MQIAMO_CBS_FAILED
데이터 유형:	MQCFIN
리턴됨:	사용 가능할 때.

### **채널 통계 메시지 데이터**

이 페이지를 사용하여 채널 통계 메시지의 구조를 봅니다.

메시지 이름:	채널 통계 메시지.
플랫폼:	All, except WebSphere MQ for z/OS.
시스템 큐:	SYSTEM.ADMIN.STATISTICS.QUEUE.

### ***QueueManager***

설명:	대기열 관리자의 이름입니다.
ID:	MQCA_Q_MGR_NAME.
데이터 유형:	MQCFST.
최대 길이:	MQ_Q_MGR_NAME_LENGTH.
리턴됨:	항상.

### ***IntervalStartDate***

설명:	모니터링 기간의 시작 날짜.
ID:	MQCAMO_START_DATE.
데이터 유형:	MQCFST.
최대 길이:	MQ_DATE_LENGTH.
리턴됨:	항상.

### ***IntervalStartTime***

설명:	모니터링 기간의 시작 시간.
ID:	MQCAMO_START_TIME.
데이터 유형:	MQCFST.
최대 길이:	MQ_TIME_LENGTH.
리턴됨:	항상.

### ***IntervalEndDate***

설명:	모니터링 기간의 종료 날짜
ID:	MQCAMO_END_DATE.
데이터 유형:	MQCFST.
최대 길이:	MQ_DATE_LENGTH.
리턴됨:	항상.

### ***IntervalEndTime***

설명:                   모니터링 기간의 종료 시간  
ID:                     MQCAMO\_END\_TIME.  
데이터 유형:           MQCFST.  
최대 길이:             MQ\_TIME\_LENGTH  
리턴됨:                항상.

### ***CommandLevel***

설명:                   큐 관리자 명령 레벨.  
ID:                     MQIA\_COMMAND\_LEVEL.  
데이터 유형:           MQCFIN.  
리턴됨:                항상.

### ***ObjectCount***

설명:                   통계 데이터가 기록된 간격에 액세스한 채널 오브젝트의 수. 이 값은 메시지에 포함된 ChlStatisticsData PCF 그룹 수로 설정됩니다.  
ID:                     MQIAMO\_OBJECT\_COUNT  
데이터 유형:           MQCFIN.  
리턴됨:                항상.

### ***ChlStatisticsData***

설명:                   채널에 대한 통계 세부사항을 지정하는 그룹화된 매개변수.  
ID:                     MQGACF\_CHL\_STATISTICS\_DATA.  
데이터 유형:           MQCFGR.  
그룹의 매개변수:        *ChannelName*  
                          *ChannelType*  
                          *RemoteQmgr*  
                          *ConnectionName*  
                          *MsgCount*  
                          *TotalBytes*  
                          *NetTimeMin*  
                          *NetTimeAvg*  
                          *NetTimeMax*  
                          *ExitTimeMin*  
                          *ExitTimeAvg*  
                          *ExitTimeMax*  
                          *FullBatchCount*  
                          *IncplBatchCount*  
                          *AverageBatchSize*  
                          *PutRetryCount*  
  
리턴됨:                항상.

### ***ChannelName***

설명:                   채널의 이름.

ID: MQCACH\_CHANNEL\_NAME.  
 데이터 유형: MQCFST.  
 최대 길이: MQ\_CHANNEL\_NAME\_LENGTH.  
 리턴됨: 항상.

**ChannelType**

설명: 채널 유형.  
 ID: MQIACH\_CHANNEL\_TYPE.  
 데이터 유형: MQCFIN.  
 값: 가능한 값은 다음과 같습니다.

- MQCHT\_SENDER**  
송신자 채널.
- MQCHT\_SERVER**  
서버 채널.
- MQCHT\_RECEIVER**  
수신자 채널.
- MQCHT\_REQUESTER**  
요청자 채널.
- MQCHT\_CLUSRCVR**  
클러스터 수신자 채널.
- MQCHT\_CLUSSDR**  
클러스터 송신자 채널.

리턴됨: 항상.

**RemoteQmgr**

설명: 리모트 큐 관리자의 이름.  
 ID: MQCA\_REMOTE\_Q\_MGR\_NAME.  
 데이터 유형: MQCFST.  
 최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH  
 리턴됨: 사용 가능할 때.

**ConnectionName**

설명: 리모트 큐 관리자의 연결 이름.  
 ID: MQCACH\_CONNECTION\_NAME.  
 데이터 유형: MQCFST  
 최대 길이: MQ\_CONN\_NAME\_LENGTH  
 리턴됨: 사용 가능할 때.

**MsgCount**

설명: 송신 또는 수신된 지속 및 비지속 메시지의 수.  
 ID: MQIAMO\_MSGS.  
 데이터 유형: MQCFIN  
 리턴됨: 사용 가능할 때.

### **TotalBytes**

설명: 지속 및 비지속 메시지의 송신 또는 수신된 바이트 수.  
ID: MQIAMO64\_BYTES.  
데이터 유형: MQCFIN64.  
리턴됨: 사용 가능할 때.

### **NetTimeMin**

설명: 레코딩 간격에 측정된 최단 기록 채널 라운드트립(마이크로초).  
ID: MQIAMO\_NET\_TIME\_MIN.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### **NetTimeAvg**

설명: 레코딩 간격에 측정된 평균 기록 채널 라운드트립(마이크로초).  
ID: MQIAMO\_NET\_TIME\_AVG.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### **NetTimeMax**

설명: 레코딩 간격에 측정된 최장 기록 채널 라운드트립(마이크로초).  
ID: MQIAMO\_NET\_TIME\_MAX.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### **ExitTimeMin**

설명: 레코딩 간격에 사용자 엑시트를 실행하는 데 소모한 최단 기록 시간(마이크로초).  
ID: MQIAMO\_EXIT\_TIME\_MIN.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### **ExitTimeAvg**

설명: 레코딩 간격에 사용자 엑시트를 실행하는 데 소모한 평균 기록 시간(마이크로초). 마이크로초로 측정됩니다.  
ID: MQIAMO\_EXIT\_TIME\_AVG.  
데이터 유형: MQCFIN.  
리턴됨: 사용 가능할 때.

### **ExitTimeMax**

설명: 레코딩 간격에 사용자 엑시트를 실행하는 데 소모한 최장 기록 시간(마이크로초). 마이크로초로 측정됩니다.  
ID: MQIAMO\_EXIT\_TIME\_MAX.  
데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **FullBatchCount**

설명: 채널 속성 BATCHSZ 또는 BATCHLIM의 값에 도달해서 송신된, 채널에서 처리된 배치의 수.

ID: MQIAMO\_FULL\_BATCHES.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **IncmlBatchCount**

설명: 채널 속성 BATCHSZ의 값에 도달하지 않아서 송신된, 채널에서 처리된 배치의 수.

ID: MQIAMO\_INCOMPLETE\_BATCHES.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **AverageBatchSize**

설명: 채널에서 처리된 배치의 평균 배치 크기.

ID: MQIAMO\_AVG\_BATCH\_SIZE.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

### **PutRetryCount**

설명: 시간 간격 중 메시지 넣기에 실패하여 재시도 루프에 들어간 횟수.

ID: MQIAMO\_PUT\_RETRIES.

데이터 유형: MQCFIN.

리턴됨: 사용 가능할 때.

## **참조 참고**

이 페이지를 사용하여 회계 및 통계 메시지 구조에 대한 설명이 참조하는 참고를 봅니다.

다음 메시지 데이터 설명이 참고를 참조합니다.

- [138 페이지의 『MQI 회계 메시지 데이터』](#)
- [148 페이지의 『큐 회계 메시지 데이터』](#)
- [158 페이지의 『MQI 통계 메시지 데이터』](#)
- [168 페이지의 『큐 통계 메시지 데이터』](#)
- [175 페이지의 『채널 통계 메시지 데이터』](#)

1. 이 매개변수는 WebSphere MQ 오브젝트와 연관됩니다. 이 매개변수는 다음 상수로 색인화된 값(MQCFIL 또는 MQCFIL64) 배열입니다.

표 24. 오브젝트 유형별로 색인화된 배열	
오브젝트 유형	값 컨텍스트
MQOT_Q (1)	큐 오브젝트에 관련된 값을 포함합니다.
MQOT_NAMELIST (2)	이름 목록 오브젝트에 관련된 값을 포함합니다.

표 24. 오브젝트 유형별로 색인화된 배열 (계속)	
오브젝트 유형	값 컨텍스트
MQOT_PROCESS (3)	프로세스 오브젝트에 관련된 값을 포함합니다.
MQOT_Q_MGR (5)	큐 관리자 오브젝트에 관련된 값을 포함합니다.
MQOT_CHANNEL (6)	채널 오브젝트에 관련된 값을 포함합니다.
MQOT_AUTH_INFO (7)	인증 정보 오브젝트에 관련된 값을 포함합니다.
MQOT_TOPIC (8)	토픽 오브젝트에 관련된 값을 포함합니다.

참고: 13 MQCFIL 또는 MQCFIL64 값의 배열이 리턴되지만 나열된 값만 의미가 있습니다.

- 이 매개변수는 WebSphere MQ 메시지와 연관됩니다. 이 매개변수는 다음 상수로 색인화된 값(MQCFIL 또는 MQCFIL64) 배열입니다.

표 25. 지속 값별로 색인화된 배열	
상수	가치
1	비지속 메시지에 대한 값을 포함합니다.
2	지속 메시지에 대한 값을 포함합니다.

참고: 각 배열에 대한 색인은 0에서 시작하므로 색인 1은 배열의 두 번째 행을 나타냅니다. 이 테이블에 나열되지 않은 배열의 요소는 회계 또는 통계 정보를 포함하지 않습니다.

## 애플리케이션 활동 추적

애플리케이션 활동 추적은 큐 관리자에 연결된 애플리케이션의 작동에 대한 자세한 정보를 생성합니다. 애플리케이션의 작동을 추적하고 IBM WebSphere MQ 자원과 상호작용할 때 애플리케이션에 사용된 매개변수의 상세 보기를 제공합니다. 애플리케이션이 발행한 MQI 호출의 시퀀스도 표시합니다.

이벤트 모니터링, 메시지 모니터링, 회계 및 통계 메시지, 실시간 모니터링이 제공하는 정보 이상의 자세한 정보가 필요할 때 애플리케이션 활동 추적을 사용하십시오.

## 애플리케이션 활동 추적 정보 수집

애플리케이션 활동 추적 메시지는 PCF 메시지입니다. 구성 파일을 사용하여 활동 추적을 구성합니다. 애플리케이션 활동 추적 정보를 수집하려면 ACTVTRC 큐 관리자 속성을 설정합니다. MQCONNX 옵션을 사용하여 연결 레벨에서 또는 활동 추적 구성 파일을 사용하여 애플리케이션 스탠자 레벨에서 이 설정을 대체할 수 있습니다.

### 이 태스크 정보

활동 추적 메시지는 MQMD 구조 즉, PCF(MQCFH) 헤더 구조와 뒤이은 여러 PCF 매개변수로 이루어집니다. PCF 매개변수 다음에는 일련의 ApplicationTraceData PCF 그룹이 있습니다. 이 PCF 그룹은 애플리케이션에 큐 관리자에 연결된 동안에 수행하는 MQI 조작에 대한 정보를 수집합니다. mqat.ini라는 구성 파일을 사용하여 활동 추적을 구성합니다.

애플리케이션 활동 추적 정보의 수집 여부를 제어하려면 하나 이상의 다음 설정을 구성합니다.

- ACTVTRC 큐 관리자 속성.
- ACTVCONO 설정(MQCONN에 전달된 MQCNO 구조의).
- 활동 추적 구성 파일 mqat.ini의 애플리케이션에 대한 일치하는 스탠자.

이전 순서가 중요합니다. ACTVTRC 속성은 ACTVCONO 설정으로 대체되고 이 설정은 mqat.ini 파일의 설정으로 대체됩니다.

다르게 명시되지 않는 한, 각 조작이 완료된 후에 추적 항목이 기록됩니다. 이 항목은 먼저 시스템 큐 SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE에 기록된 후 애플리케이션과 큐 관리자의 연결이 끊어질 때 애플리

케이션 활동 추적 메시지에 기록됩니다. 장기 실행 애플리케이션의 경우 다음 이벤트가 발생하면 중간 이벤트가 기록됩니다.

- 연결 수명이 정의된 제한시간 값에 도달합니다.
- 조작 수가 지정된 수에 도달합니다.
- 메모리의 수집된 데이터 양이 큐의 허용된 최대 메시지 길이에 도달합니다.

ActivityInterval 매개변수를 사용하여 제한시간 값을 설정합니다. 조작 수는 ActivityCount 매개변수를 사용하여 설정합니다. 두 매개변수 모두 활동 추적 구성 파일 mqat.ini에 지정됩니다.

애플리케이션 활동 추적의 사용은 성능에 영향을 미칠 수 있습니다. ActivityCount 및 ActivityInterval 설정을 조정해서 오버헤드를 줄일 수 있습니다. [187 페이지의 『애플리케이션 활동 추적의 성능 영향 조정』](#)를 참조하십시오.

애플리케이션 활동 추적 메시지의 콘텐츠를 보는 가장 단순한 방법은 [188 페이지의 『amqsact 샘플 프로그램』](#)을 사용하는 것입니다.

## 프로시저

1. [181 페이지의 『ACTVTRC를 설정하여 활동 추적 정보 콜렉션 제어』](#).
2. [182 페이지의 『MQCONNX를 설정하여 활동 추적 정보 콜렉션 제어』](#).
3. [182 페이지의 『mqat.ini를 사용하여 활동 추적 작동 구성』](#).
4. [187 페이지의 『애플리케이션 활동 추적의 성능 영향 조정』](#).

## ACTVTRC를 설정하여 활동 추적 정보 콜렉션 제어

큐 관리자 속성 ACTVTRC를 사용하여 MQI 애플리케이션 활동 추적 정보 콜렉션을 제어합니다.

### 이 태스크 정보

애플리케이션 활동 추적 메시지는 애플리케이션 활동 추적이 사용된 후에 시작한 연결에 대해서만 생성됩니다. ACTVTRC 매개변수의 가능한 값은 다음과 같습니다.

#### 켜짐(ON)

API 활동 추적 콜렉션을 켭니다.

#### 꺼짐(OFF)

API 활동 추적 콜렉션을 끕니다.

**참고:** ACTVTRC 설정은 큐 관리자 ACTVCONO 매개변수로 대체될 수 있습니다. ACTVCONO 매개변수를 ENABLED로 설정하면 MQCNO 구조에서 Options 필드를 사용하여 주어진 연결에 대한 ACTVTRC 설정을 대체할 수 있습니다. [182 페이지의 『MQCONNX를 설정하여 활동 추적 정보 콜렉션 제어』](#)를 참조하십시오.

### 예

ACTVTRC 매개변수의 값을 변경하려면 MQSC 명령 ALTER QMGR을 사용합니다. 예를 들어, MQI 애플리케이션 활동 추적 정보 콜렉션을 사용하려면 다음 MQSC 명령을 사용하십시오.

```
ALTER QMGR ACTVTRC(ON)
```

## 다음에 수행할 작업

애플리케이션 활동 추적 메시지의 콘텐츠를 보는 가장 단순한 방법은 [188 페이지의 『amqsact 샘플 프로그램』](#)을 사용하는 것입니다.

애플리케이션 활동 추적의 사용은 성능에 영향을 미칠 수 있습니다. ActivityCount 및 ActivityInterval 설정을 조정해서 오버헤드를 줄일 수 있습니다. [187 페이지의 『애플리케이션 활동 추적의 성능 영향 조정』](#)를 참조하십시오.

## MQCONNX를 설정하여 활동 추적 정보 콜렉션 제어

큐 관리자 속성 **ACTVCONO**가 ENABLED로 설정되면 MQCONNX 호출에 **ConnectOpts** 매개변수를 사용하여 각 연결별로 애플리케이션 활동 보고서를 사용 가능 또는 불가능하게 할 수 있습니다. 이 옵션은 큐 관리자 속성 **ACTVTRC**에 정의된 활동 추적 작동을 대체하며 활동 추적 구성 파일 `mqat.ini`의 설정으로 대체될 수 있습니다.

### 프로시저

1. 큐 관리자 속성 **ACTVCONO**를 ENABLED로 설정하십시오.

**참고:** 애플리케이션이 **ConnectOpts** 매개변수를 사용하여 애플리케이션의 회계 작동을 수정하려 시도하며 QMGR 속성 **ACTVCONO**가 DISABLED로 설정된 경우에는, 애플리케이션에 오류가 리턴되지 않고 활동 추적 콜렉션이 큐 관리자 속성 또는 활동 추적 구성 파일 `mqat.ini`를 통해 정의됩니다.

2. MQCONNX 호출의 **ConnectOpts** 매개변수를 `MQCNO_ ACTIVITY_ TRACE_ENABLED`로 설정하십시오.

MQCONNX 호출의 **ConnectOpts** 매개변수의 가능한 값은 다음과 같습니다.

#### **MQCNO\_ACTIVITY\_TRACE\_DISABLED**

연결에 대한 활동 추적을 끕니다.

#### **MQCNO\_ACTIVITY\_TRACE\_ENABLED**

연결에 대한 활동 추적을 켭니다.

**참고:** 애플리케이션이 MQCONNX에 대해 `MQCNO_ ACTIVITY_ TRACE_ENABLED` 및 `MQCNO_ACTIVITY_ TRACE_DISABLED`를 모두 선택하면 이유 코드 `MQRC_OPTIONS_ERROR`로 호출에 실패합니다.

3. 활동 추적 구성 파일 `mqat.ini`의 설정으로 활동 추적 설정이 대체되지 않는지 확인하십시오.

[182 페이지의 『mqat.ini를 사용하여 활동 추적 작동 구성』](#)을 참조하십시오.

### 다음에 수행할 작업

애플리케이션 활동 추적 메시지의 콘텐츠를 보는 가장 단순한 방법은 [188 페이지의 『amqsact 샘플 프로그램』](#)을 사용하는 것입니다.

애플리케이션 활동 추적의 사용은 성능에 영향을 미칠 수 있습니다. **ActivityCount** 및 **ActivityInterval** 설정을 조정해서 오버헤드를 줄일 수 있습니다. [187 페이지의 『애플리케이션 활동 추적의 성능 영향 조정』](#)를 참조하십시오.

### mqat.ini를 사용하여 활동 추적 작동 구성

활동 추적 작동은 `mqat.ini` 구성 파일을 사용하여 구성됩니다. 이 파일은 `mqs.ini` 및 `qm.ini` 파일과 동일한 스탠자 키 및 매개변수-값 쌍 형식을 따릅니다.

### 이 태스크 정보

**Linux** **UNIX** UNIX and Linux 시스템에서 `mqat.ini`는 큐 관리자 데이터 디렉토리에 있으며 이는 `qm.ini` 파일과 동일한 위치입니다.

**Windows** Windows 시스템에서 `mqat.ini`는 큐 관리자 데이터 디렉토리 `C:\Program Files\IBM\WebSphere MQ\qmgrs\queue_manager_name`에 있습니다. 추적할 애플리케이션을 실행하는 사용자에게는 이 파일을 읽을 권한이 필요합니다.

**참고:** IBM WebSphere MQ Version 7.1 또는 이전에서 마이그레이션한 큐 관리자에서 `mqat.ini` 파일이 누락됩니다. 그런 경우, `mqat.ini` 파일을 수동으로 작성해야 하며 660 권한이 파일에 설정되어야 합니다.

파일 형식의 구문 규칙은 다음과 같습니다.

- 해시 또는 세미콜론으로 시작하는 텍스트는 행의 맨 끝까지 계속되는 주석으로 간주됩니다.
- 첫 번째 유효(무주석) 행은 스탠자 키여야 합니다.

- 스탠자 키는 스탠자의 이름과 뒤이은 콜론으로 이루어집니다.
- 매개변수-값 쌍은 매개변수의 이름과 뒤이은 등호 부호 및 값으로 이루어집니다.
- 한 행에는 단일 매개변수-값 쌍만 표시할 수 있습니다. (매개변수-값이 다른 행으로 넘어가면 안됩니다.)
- 선두 및 후미 공백은 무시됩니다. 스탠자 이름, 매개변수 이름과 값 또는 매개변수/값 쌍 사이의 공백 수에는 제한이 없습니다. 행 바꾸기는 유효하며 무시되지 않습니다.
- 행의 최대 길이는 2048자입니다.
- 스탠자 키, 매개변수 이름, 상수 매개변수 값은 대소문자를 구분하지 않지만 변수 매개변수 값(**AppName** 및 **DebugPath**)은 대소문자를 구분합니다.

### 스탠자 키

구성 파일에서 두 유형의 스탠자 키(AllActivityTrace 스탠자 및 ApplicationTrace 스탠자)를 사용할 수 있습니다.

#### AllActivityTrace 스탠자

AllActivityTrace 스탠자는 대체하지 않는 한 모든 IBM WebSphere MQ 연결에 적용될 활동 추적의 설정을 정의합니다.

AllActivityTrace 스탠자의 개별 값은 ApplicationTrace 스탠자의 보다 특정한 정보로 대체될 수 있습니다.

두 개 이상의 AllActivityTrace 스탠자가 지정된 경우 마지막 스탠자의 값이 사용됩니다. 선택된 AllActivityTrace 에서 누락된 매개변수는 기본값을 사용합니다. 이전 AllActivityTrace 스탠자의 매개변수와 값은 무시됩니다.

#### ApplicationTrace 스탠자

ApplicationTrace 스탠자는 IBM WebSphere MQ 연결의 특정 이름, 유형 또는 둘 다에 적용될 수 있는 설정을 정의합니다.

이 스탠자에는 연결 일치 규칙에 정의된 일치 규칙에 따라 사용되는 ApplName 및 ApplClass 값이 포함되어 있어 스탠자가 특정 연결에 적용되는지 여부를 판별합니다.

#### 매개변수/값 쌍

다음 표에는 활동 추적 구성 파일에서 사용될 수 있는 매개변수/값 쌍이 나열되어 있습니다.

표 26. 활동 추적 구성 파일에 사용 가능한 매개변수/값 쌍			
이름	스탠자 유형	값(기본값은 굵게)	설명
추적	ApplicationTrace	<b>ON/OFF</b>	활동 추적 스위치. 이 스위치는 현재 애플리케이션 스탠자 범위에 대한 활동 추적이 활성화되는지 여부를 판별하는 용도로 애플리케이션별 스탠자에 사용할 수 있습니다. 이 값은 큐 관리자의 ACTVTRC 및 ACTVCONO 설정을 대체함에 유의하십시오.

표 26. 활동 추적 구성 파일에 사용 가능한 매개변수/값 쌍 (계속)

이름	스탠자 유형	값(기본값은 굵게)	설명
ActivityInterval	AllActivityTrace ApplicationTrace	<b>0</b> - 99999999( <b>0=off</b> )	메시지 추적 간 시간 간격 (초 단위)입니다. 활동 추적에서는 타이머 스레드를 사용하지 않으므로, 추적 메시지는 시간이 경과되는 정확한 순간에 기록되지 않습니다. 대신 시간 간격이 경과한 후 첫 번째 MQI 조치가 실행될 때 기록됩니다. 이 값이 0이면 연결이 끊겼을 경우 또는 활동 수에 도달했을 경우에 추적 메시지가 기록됩니다.
ActivityCount	AllActivityTrace ApplicationTrace	<b>0</b> - 99999999( <b>0=off</b> )	추적 메시지 간의 MQI 또는 XA 조작 수. 이 값이 0이면 연결이 끊겼을 경우 또는 활동 간격이 경과했을 경우에 추적 메시지가 기록됩니다.
TraceLevel	AllActivityTrace ApplicationTrace	LOW / <b>MEDIUM</b> / HIGH	각 조작에 대해 추적한 매개변수 세부사항의 양. 각 추적 레벨의 매개변수가 포함된 개별 조작 세부사항의 설명.
TraceMessageData	AllActivityTrace ApplicationTrace	<b>0</b> - 104 857 600(100MB)	MQGET, MQPUT, MQPUT1, 콜백 조작에 대해 추적한 메시지 데이터의 양(바이트)

표 26. 활동 추적 구성 파일에 사용 가능한 매개변수/값 쌍 (계속)

이름	스탠자 유형	값(기본값은 굵게)	설명
ApplName	ApplicationTrace	문자열(필수 매개변수 - 기본값 없음)	이 값은 ApplicationTrace 스탠자가 적용되는 애플리케이션을 판별하는 데 사용됩니다. API 엑시트 컨텍스트 구조의 ApplName 값 (MQMD.PutApplName에 해당함)에 일치합니다. ApplName 값의 콘텐츠는 애플리케이션 환경에 따라 다릅니다. 분산 플랫폼에서는 MQXC.ApplName의 파일 이름 부분만이 스탠자의 값과 일치됩니다. 비교를 수행할 때 오른쪽 끝에 있는 경로 구분 기호 왼쪽의 문자는 무시됩니다. z/OS 애플리케이션의 경우 전체 MQXC.ApplName이 스탠자의 값에 일치합니다. 단일 와일드카드 문자(*)를 ApplName 값의 끝에 사용하여 해당 지점 다음에 오는 임의의 수의 문자를 일치시킬 수 있습니다. ApplName 값이 단일 와일드카드 문자(*)로 설정되면 ApplName 값은 모든 애플리케이션에 일치합니다.
ApplClass	ApplicationTrace	<b>USER/MCA/INTERNAL/ALL</b>	애플리케이션의 클래스. AppType 값이 IBM WebSphere MQ 연결에 해당하는 방식에 대한 설명은 다음 표를 참조하십시오.

다음 표에서는 AppClass 값이 연결 API 엑시트 컨텍스트 구조의 APICallerType 및 APIEnvironment 필드에 대응되는 방법을 표시합니다.

표 27. Appclass 값과 APICallerType 및 APIEnvironment 필드에 이 값을 일치시키는 방식

APPLCLASS	API 호출자 유형:	API 환경:	설명
user	MQXACT_EXTERNAL	MQXE_OTHER	사용자 애플리케이션만 추적함
MCA	(임의의 값)	MQXE_MCA MQXE_MCA_CLNTCONN MQXE_MCA_SVRCONN	클라이언트 및 채널(amqrmppa)
내부	MQXACT_EXTERNAL	MQXE_COMMAND_SERVER MQXE_MQSC	'runmqsc' 및 명령 서버

표 27. <i>Appclass</i> 값과 <i>APICallerType</i> 및 <i>APIEnvironment</i> 필드에 이 값을 일치시키는 방식 (계속)			
APPLCLASS	API 호출자 유형:	API 환경:	설명
내부	MQXACT_INTERNAL	(임의의 값)	"신뢰" 및 내부 애플리케이션과 프로세스. 예: amqzdmaa
모두	(임의의 값)	(임의의 값)	추적되는 모든 사용자 및 내부 연결



**주의:** 클라이언트 사용자 애플리케이션에 대해 **APPLCLASS/MCA**를 사용해야 합니다. *USER*의 한 클래스로서 이들과 일치되지 않습니다.

예를 들어 **amqspu<sub>t</sub>c** 샘플 애플리케이션을 추적하려면 다음 코드를 사용합니다.

```

ApplicationTrace:
AppClass=MCA                                # Application type
                                           # Values: (USER | MCA | INTERNAL | ALL)
                                           # Default: USER
AppName=amqsputc      # Application name (may be wildcarded)
                                           # (matched to app name without path)
                                           # Default: *
Trace=ON                                     # Activity trace switch for application
                                           # Values: ( ON | OFF )
                                           # Default: OFF
ActivityInterval=30                         # Time interval between trace messages
                                           # Values: 0-99999999 (0=off)
                                           # Default: 0
ActivityCount=1                             # Number of operations between trace msgs
                                           # Values: 0-99999999 (0=off)
                                           # Default: 0
TraceLevel=MEDIUM                           # Amount of data traced for each operation
                                           # Values: LOW | MEDIUM | HIGH
                                           # Default: MEDIUM
TraceMessageData=1000                       # Amount of message data traced
                                           # Values: 0-100000000
                                           # Default: 0

```

### 연결 일치 규칙

큐 관리자는 연결에 사용할 스탠자 설정을 판별하기 위해 다음 규칙을 적용합니다.

1. AllActivityTrace 스탠자에서 지정되는 값은 값이 ApplicationTrace 스탠자에도 발생하고 스탠자가 2, 3 및 4 지점에서 설명한 연결 기준에 일치하는 경우가 아니면 연결에 사용되지 않습니다.
2. AppClass 는 IBM WebSphere MQ 연결 유형과 일치합니다. AppClass가 연결 유형과 일치하지 않는 경우 이 연결에서 스탠자가 무시됩니다.
3. 스탠자의 AppName 값이 연결에 대한 API 엑시트 컨텍스트 구조(MQAXC)의 AppName 필드에 있는 파일 이름 부분에 관해 일치합니다. 파일 이름 부분은 마지막 경로 구분 기호(/ 또는 \) 문자의 오른쪽에 있는 문자에서 파생됩니다. 스탠자 AppName에 (\*) 와일드카드가 포함된 경우 와일드카드 왼쪽에 있는 문자만 연결 AppName의 해당 문자 수와 비교합니다. 예를 들어, 스탠자 값 "FRE\*"가 지정되면 처음 세 개의 문자만 비교에 사용되므로 "path/FREEDOM" 및 "path\FREDDY"는 일치하지만 "path/FRIEND"는 일치하지 않습니다. 스탠자의 AppName 값이 연결 AppName과 일치하지 않는 경우 이 연결에서 스탠자가 무시됩니다.
4. 두 개 이상의 스탠자가 연결의 AppName 및 AppClass와 일치하는 경우 가장 특정한 AppName의 스탠자가 사용됩니다. 가장 특정한 AppName은 연결의 AppName과 일치하는 문자가 가장 많은 스탠자로 정의됩니다. 예를 들어 ini 파일에 AppName="FRE\*"인 스탠자와 AppName="FREE\*"인 다른 스탠자가 있으면, AppName="FREE\*"인 스탠자에서 4글자가 일치하므로 AppName="path/FREEDOM"과의 연결에 가장 일치하는 것으로 선택됩니다(여기서 AppName="FRE\*"는 3자만 일치함).
5. 2, 3 및 4의 규칙을 적용한 후 연결 AppName 및 AppClass와 일치하는 스탠자가 둘 이상이면, 마지막 일치하는 값이 사용되고 다른 모든 스탠자는 무시합니다.

### 애플리케이션 활동 추적 파일 예

다음 예는 활동 추적 ini 파일에 구성 데이터가 지정되는 방식을 보여줍니다. 이 예는 C 샘플 디렉토리 (amqsact.c file과 같은 디렉토리)의 mqat.ini라는 샘플로 제공됩니다.

```

AllActivityTrace:
  ActivityInterval=0          # Time interval between trace messages
                              # Values: 0-99999999 (0=off)
                              # Default: 0
  ActivityCount=0            # Number of operations between trace msgs
                              # Values: 0-99999999 (0=off)
                              # Default: 0
  TraceLevel=MEDIUM         # Amount of data traced for each operation
                              # Values: LOW | MEDIUM | HIGH
                              # Default: MEDIUM
  TraceMessageData=0        # Amount of message data traced
                              # Values: 0-100000000
                              # Default: 0

ApplicationTrace:
  ApplClass=USER             # Application type
                              # Values: (USER | MCA | INTERNAL | ALL)
                              # Default: USER
  ApplName=AppName*         # Application name (may be wildcard)
                              # (matched to app name without path)
                              # Default: *
  Trace=OFF                 # Activity trace switch for application
                              # Values: ( ON | OFF )
                              # Default: OFF
ActivityInterval=0          # Time interval between trace messages
                              # Values: 0-99999999 (0=off)
                              # Default: 0
  ActivityCount=0            # Number of operations between trace msgs
                              # Values: 0-99999999 (0=off)
                              # Default: 0
  TraceLevel=MEDIUM         # Amount of data traced for each operation
                              # Values: LOW | MEDIUM | HIGH
                              # Default: MEDIUM
  TraceMessageData=0        # Amount of message data traced
                              # Values: 0-100000000
                              # Default: 0

```

## 다음에 수행할 작업

애플리케이션 활동 추적의 사용은 성능에 영향을 미칠 수 있습니다. **ActivityCount** 및 **ActivityInterval** 설정을 조정해서 오버헤드를 줄일 수 있습니다. [187 페이지의 『애플리케이션 활동 추적의 성능 영향 조정』](#)를 참조하십시오.

## 애플리케이션 활동 추적의 성능 영향 조정

애플리케이션 활동 추적을 사용하면 성능이 저하될 수 있습니다. 큐를 드레인하는 애플리케이션의 수를 늘리거나 `mqat.ini`에서 **ActivityInterval**, **ActivityCount**, **TraceLevel**을 조정해서 필요한 애플리케이션만 추적함으로써 이 영향을 감소시킬 수 있습니다.

## 이 태스크 정보

선택적으로 한 애플리케이션에 또는 모든 큐 관리자 애플리케이션에 애플리케이션 활동 추적을 사용하면 추가 메시징 활동이 유발되어 큐 관리자에 추가 스토리지 공간이 필요하게 됩니다. 워크로드가 높은 애플리케이션이나 SLA(Service Level Agreement)에 메시징 제공자로부터의 최소 응답이 시간이 요구되는 환경처럼 메시징 성능이 중요한 환경에서는, 애플리케이션 활동 추적을 수집하는 것이 적합하지 않거나 생성되는 추적 활동 메시지의 세부사항 또는 빈도를 조정해야 할 수 있습니다. `mqat.ini` 파일에서 **ActivityInterval**, **ActivityCount**, **TraceLevel**의 사전 설정된 값은 세부사항 및 성능의 기본 밸런스를 제공합니다. 하지만 시스템의 세밀한 기능 및 성능 요구사항에 맞게 이 값을 조정할 수 있습니다.

## 프로시저

- 필요한 애플리케이션만 추적하십시오.

`mqat.ini`에 ApplicationTrace 애플리케이션별 스탠자를 작성하거나 MQCONNX 호출에서 **MQCNO** 구조의 옵션 필드에 **MQCNO\_ACTIVITY\_TRACE\_ENABLED**를 지정하도록 애플리케이션을 변경하여 이를 수행하십시오. [182 페이지의 『mqat.ini를 사용하여 활동 추적 작동 구성』](#) 및 [182 페이지의 『MQCONNX를 설정하여 활동 추적 정보 콜렉션 제어』](#)를 참조하십시오.

- 추적을 시작하기 전에 최소 하나의 애플리케이션이 실행 중이며 SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE에서 활동 추적 메시지 데이터를 검색할 준비가 되어 있는지 확인하십시오.
- 큐를 드레인하는 애플리케이션의 수를 늘려서 큐 용량을 가능한 낮게 유지하십시오.
- mqat.ini 파일에 **TraceLevel** 값을 설정하여 필요한 최소 데이터를 수집하십시오.  
TraceLevel=LOW가 메시징 성능에 가장 적은 영향을 미칩니다. 182 페이지의 『mqat.ini를 사용하여 활동 추적 작동 구성』의 내용을 참조하십시오.
- mqat.ini에서 **ActivityCount** 및 **ActivityInterval** 값을 조정하여 활동 추적 메시지가 생성되는 빈도를 조정하십시오.  
여러 애플리케이션을 추적하는 경우에는 활동 추적 메시지가 SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE에서 제거 가능한 것보다 더 빠르게 생성되고 있는 것일 수 있습니다. 하지만 활동 추적 메시지가 생성되는 빈도를 줄이면 그와 동시에 큐 관리자에 필요한 스토리지 공간 및 큐에 기록될 때의 메시지 크기를 늘리는 것이기도 합니다.

## 다음에 수행할 작업

### amqsact 샘플 프로그램

**amqsact**는 애플리케이션 활동 추적 메시지를 형식화하며 WebSphere MQ와 함께 제공됩니다.

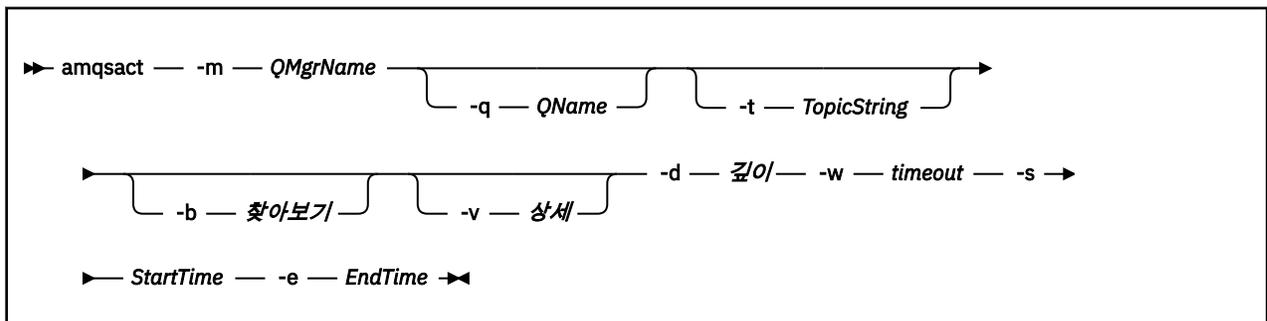
컴파일된 프로그램은 다음과 같은 샘플 디렉토리에 있습니다.

- UNIX and Linux MQ\_INSTALLATION\_PATH/samp/bin
- Windows MQ\_INSTALLATION\_PATH\tools\c\Samples\Bin

### 표시 모드

기본적으로 표시 모드에서 **amqsact**는 SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE의 메시지를 처리합니다. 큐 이름이나 토픽 문자열을 지정하여 이 작동을 대체할 수 있습니다.

표시된 추적 기간을 제어하고 활동 추적 메시지를 표시 후 제거하거나 보유할지 여부를 지정할 수도 있습니다.



### 표시 모드의 필수 매개변수

#### **-m QMgrName**

큐 관리자의 이름.

#### **-d Depth**

표시할 레코드 수.

#### **-w Timeout**

대기 시간(초). 지정된 기간에 추적 메시지가 표시되지 않을 경우 **amqsact**가 종료됩니다.

#### **-s StartTime**

처리할 레코드의 시작 시간.

### -e EndTime

처리할 레코드의 종료 시간.

## 표시 모드의 선택적 매개변수

### -q QName

기본 큐 이름을 대체할 특정 큐를 지정하십시오.

### -t TopicString

이벤트 토픽을 구독함

### -b

레코드만 찾아봄

### -v

상세 출력

## 표시 모드의 출력 예

amqsact를 큐 관리자 TESTQM에 사용하며 MQCONN API 호출에 대해 상세 출력합니다.

```
amqsact -m TESTQM -v
```

이전 명령의 출력 예는 다음과 같습니다.

```
MonitoringType: MQI Activity Trace
Correl_id:
00000000: 414D 5143 5445 5354 514D 2020 2020 2020 'AMQCTESTQM '
00000010: B5F6 4251 2000 E601
QueueManager: 'TESTQM'
Host Name: 'ADMINIB-1VTJ6N1'
IntervalStartDate: '2014-03-15'
IntervalStartTime: '12:08:10'
IntervalEndDate: '2014-03-15'
IntervalEndTime: '12:08:10'
CommandLevel: 750
SeqNumber: 0
ApplicationName: 'MQ_1\bin\amqsput.exe'
Application Type: MQAT_WINDOWS_7
ApplicationPid: 14076
UserId: 'Emma_Bushby'
API Caller Type: MQXACT_EXTERNAL
API Environment: MQXE_OTHER
Application Function: ''
Appl Function Type: MQFUN_TYPE_UNKNOWN
Trace Detail Level: 2
Trace Data Length: 0
Pointer size: 4
Platform: MQPL_WINDOWS_7
MQI Operation: 0
Operation Id: MQXF_CONN
ApplicationTid: 1
OperationDate: '2014-03-15'
OperationTime: '12:08:10'
ConnectionId:
00000000: 414D 5143 5445 5354 514D 2020 2020 2020 'AMQCTESTQM '
00000010: FFFFFFFB5FFFFFFF6 4251 2000 FFFFFFFE601 '
QueueManager: 'TESTQM'
Completion Code: MQCC_OK
Reason Code: 0
```

## 애플리케이션 활동 추적 메시지 참조

이 페이지를 사용하여 애플리케이션 활동 추적 메시지의 형식과 이 메시지에 리턴되는 정보를 개괄합니다.

애플리케이션 활동 추적 메시지는 메시지 디스크립터 및 메시지 데이터를 포함한 표준 IBM WebSphere MQ 메시징입니다. 메시지 데이터는 IBM WebSphere MQ 애플리케이션이 수행한 MQI 조작에 대한 정보나 IBM WebSphere MQ 시스템에서 발생하는 활동에 대한 정보를 포함합니다.

## 메시지 디스크립터

- MQMD 구조

## 메시지 데이터

- PCF 헤더(MQCFH)
- 항상 리턴되는 애플리케이션 활동 추적 메시지 데이터
- 조작별 애플리케이션 활동 추적 메시지 데이터

## 애플리케이션 활동 추적 메시지 MQMD(메시지 디스크립터)

이 페이지를 사용하여 애플리케이션 활동 추적 메시지의 메시지 디스크립터와 이벤트 메시지의 메시지 디스크립터 간 차이를 이해합니다.

애플리케이션 활동 추적 메시지의 메시지 디스크립터의 매개변수와 값은 다음을 제외하면, 이벤트 메시지의 메시지 디스크립터에서와 동일합니다.

### **Format**

설명:	메시지 데이터의 형식 이름입니다.
가치:	<b>MQFMT_ADMIN</b> 관리 메시지.

### **CorrelId**

설명:	상관 ID.
가치:	애플리케이션의 ConnectionId로 초기화됨

## MQCFH(PCF 헤더)

이 페이지를 사용하여 활동 추적 메시지의 MQCFH 구조에 따라 포함된 PCF 값을 봅니다.

활동 추적 메시지의 경우 MQCFH 구조가 다음 값을 포함합니다.

### **Type**

설명:	메시지의 콘텐츠를 식별하는 구조 유형입니다.
데이터 유형:	MQLONG.
가치:	MQCFT_APP_ACTIVITY

### **StrucLength**

설명:	MQCFH 구조의 길이(바이트 단위).
데이터 유형:	MQLONG.
가치:	MQCFH_STRUC_LENGTH

### **Version**

설명:	구조 버전 번호입니다.
데이터 유형:	MQLONG.
값:	MQCFH_VERSION_3

### **Command**

설명:	명령 ID입니다. 이 필드는 메시지의 범주를 식별합니다.
데이터 유형:	MQLONG.

값: MQCMD\_ACTIVITY\_TRACE

### **MsgSeqNumber**

설명: 메시지 순서 번호입니다. 이 필드는 관련된 메시지 그룹 내에 있는 메시지의 순서 번호입니다.

데이터 유형: MQLONG.

값: 1

### **Control**

설명: 제어 옵션입니다.

데이터 유형: MQLONG.

값: MQCFC\_LAST.

### **CompCode**

설명: 완료 코드.

데이터 유형: MQLONG.

값: MQCC\_OK.

### **Reason**

설명: 완료 코드를 규정하는 이유 코드입니다.

데이터 유형: MQLONG.

값: MQRC\_NONE.

### **ParameterCount**

설명: 매개변수 구조 수입니다. 이 필드는 MQCFH 구조를 따르는 매개변수 구조의 수입니다. 그룹 구조(MQCFGR) 및 포함된 해당 매개변수 구조는 하나의 구조로만 계산됩니다.

데이터 유형: MQLONG.

값: 1 이상

## **애플리케이션 활동 추적 메시지 데이터**

PCF 헤더 바로 다음에는 활동 추적의 시간 간격을 설명하는 매개변수 세트가 있습니다. 이 매개변수는 메시지가 기록될 때 메시지의 순서도 표시합니다. 나중에 추가 정보가 추가될 수 있도록 헤더 다음의 필드 수와 순서와 보장되지 않습니다.

메시지 이름: 활동 추적 메시지.

시스템 큐: SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE.

### **QueueManager**

설명: 큐 매니저의 이름

ID: MQCA\_Q\_MGR\_NAME

데이터 유형: MQCFST

최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH

### **QSGName**

**HostName**

설명: 큐 관리자가 실행 중인 시스템의 호스트 이름  
ID: MQCACF\_HOST\_NAME  
데이터 유형: MQCFST

**IntervalStartDate**

설명: 모니터링 기간의 시작 날짜  
ID: MQCAMO\_START\_DATE  
데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH

**IntervalStartTime**

설명: 모니터링 기간의 시작 시간  
ID: MQCAMO\_START\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH

**IntervalEndDate**

설명: 모니터링 기간의 종료 날짜  
ID: MQCAMO\_END\_DATE  
데이터 유형: MQCFST  
최대 길이: MQ\_DATE\_LENGTH

**IntervalEndTime**

설명: 모니터링 기간의 종료 시간  
ID: MQCAMO\_END\_TIME  
데이터 유형: MQCFST  
최대 길이: MQ\_TIME\_LENGTH

**CommandLevel**

설명: IBM WebSphere MQ 명령 레벨  
ID: MQIA\_COMMAND\_LEVEL  
데이터 유형: MQCFIN

**SeqNumber**

설명: 일반적으로 순서 번호 0. 이 값은 장기 실행 연결에 대한 각 후속 레코드마다 증분됩니다.  
ID: MQIACF\_SEQUENCE\_NUMBER  
데이터 유형: MQCFIN

**ApplicationName**

설명: 애플리케이션의 이름. (프로그램 이름)

ID: MQCACF\_APPL\_NAME  
데이터 유형: MQCFST  
최대 길이: MQ\_APPL\_NAME\_LENGTH

### **ApplClass**

설명: 활동을 수행한 애플리케이션의 유형. 가능한 값: MQAT\_\*  
ID: MQIA\_APPL\_TYPE  
데이터 유형: MQCFIN

### **ApplicationPid**

설명: 애플리케이션의 운영 체제 프로세스 ID.  
ID: MQIACF\_PROCESS\_ID  
데이터 유형: MQCFIN

### **UserId**

설명: 애플리케이션의 사용자 ID 컨텍스트  
ID: MQCACF\_USER\_IDENTIFIER  
데이터 유형: MQCFST  
최대 길이: MQ\_USER\_ID\_LENGTH

### **APICallerType**

설명: 애플리케이션의 유형. 가능한 값: MQXACT\_EXTERNAL 또는 MQXACT\_INTERNAL  
ID: MQIACF\_API\_CALLER\_TYPE  
데이터 유형: MQCFIN

### **Environment**

설명: 애플리케이션의 런타임 환경. 가능한 값: MQXE\_OTHER MQXE\_MCA  
MQXE\_MCA\_SVRCONN MQXE\_COMMAND\_SERVER MQXE\_MQSC  
ID: MQIACF\_API\_ENVIRONMENT  
데이터 유형: MQCFIN

### **Detail**

설명: 연결에 대해 기록할 세부사항 레벨. 가능한 값: 1=LOW 2=MEDIUM 3=HIGH  
ID: MQIACF\_TRACE\_DETAIL  
데이터 유형: MQCFIN

### **TraceDataLength**

설명: 이 연결에 대해 추적하는 메시지 데이터의 길이(바이트).  
ID: MQIACF\_TRACE\_DATA\_LENGTH  
데이터 유형: MQCFIN

### **Pointer Size**

설명: 애플리케이션이 실행 중인 플랫폼의 포인터(2진 구조의 해석을 지원) 길이(바이트)

ID: MQIACF\_POINTER\_SIZE  
 데이터 유형: MQCFIN

**Platform**

설명: 큐 관리자가 실행 중인 플랫폼. 값은 MQPL\_\* 값 중 하나입니다.  
 ID: MQIA\_PLATFORM  
 데이터 유형: MQCFIN

**애플리케이션 활동 MQI 조작에 대한 변수 매개변수**

애플리케이션 활동 데이터 MQCFGR 구조 다음에는 수행 중인 조작에 해당하는 PCF 매개변수 세트가 이어집니다. 각 조작에 대한 매개변수는 다음 절에 정의되어 있습니다.

추적 레벨은 추적이 포함시킬 매개변수에 필요한 추적 세분화 레벨을 표시합니다. 가능한 추적 레벨 값은 다음과 같습니다.

1. 낮음

애플리케이션에 "낮음", "중간" 또는 "높음" 활동 추적이 구성될 때 매개변수가 포함됩니다. 이 설정은 매개변수가 항상 조작에 대한 AppActivityData 그룹에 포함됨을 의미합니다. 애플리케이션이 수행하는 MQI 호출을 추적하고 호출이 성공했는지 여부를 확인하는 데에는 이 매개변수 세트면 충분합니다.

2. 보통

애플리케이션에 대해 "중간" 또는 "높음" 활동 추적이 구성되어 있을 때에만 조작의 AppActivityData 그룹에 매개변수가 포함됩니다. 이 매개변수 세트는 애플리케이션에 사용되는 큐와 토픽 이름 같은 자원에 대한 정보를 추가합니다.

3. 높음

애플리케이션에 대해 "높음" 활동 추적이 구성되어 있을 때에만 조작의 AppActivityData 그룹에 매개변수가 포함됩니다. 이 매개변수 세트는 MQI 및 XA 함수에 전달되는 구조의 메모리 덤프를 포함합니다. 이러한 이유로, MQI 및 XA 호출에 사용되는 매개변수에 대한 자세한 정보를 포함합니다. 구조 메모리 덤프는 구조의 셸로우(shallow) 사본입니다. 참조 해제 포인터에 대한 잘못된 시도를 피하기 위해 구조의 포인터 값은 NULL로 설정됩니다.

**참고:** 덤프된 구조 버전이 애플리케이션에 사용된 버전과 반드시 동일하지는 않습니다. API 교차 엑시트, 활동 추적 코드 또는 큐 관리자를 통해 구조를 수정할 수 있습니다. 큐 관리자는 이후 버전에 구조를 수정할 수 있지만 구조의 이전 버전에는 결코 구조를 변경하지 않습니다. 이전 버전에 대해 변경할 경우 데이터 손실 위험이 있습니다.

**MQBACK**

애플리케이션이 MQBACK MQI 함수를 시작함

**CompCode**

설명: 조작 결과를 나타내는 완료 코드  
 PCF 매개변수: MQIACF\_COMP\_CODE  
 추적 레벨: 1  
 유형: MQCFIN

**Reason**

설명: 조작의 이유 코드 결과  
 PCF 매개변수: MQIACF\_REASON\_CODE  
 추적 레벨: 1  
 유형: MQCFIN

## **MQBEGIN**

애플리케이션이 MQBEGIN MQI 함수를 시작함

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형	MQCFIN

## **MQBO**

설명:	MQBEGIN 옵션 구조. NULL 포인터가 MQBEGIN 호출에 사용되면 이 매개변수가 포함되지 않습니다.
PCF 매개변수:	MQBACF_MQBO_STRUCT
추적 레벨:	3
유형	MQCFBS
길이:	MQBO 구조의 길이(바이트).

## **MQCALLBACK**

애플리케이션이 MQCALLBACK MQI 함수를 시작함

### **ObjectHandle**

설명:	오브젝트 핸들
PCF 매개변수:	MQIACF_HOBJ
추적 레벨:	1
유형	MQCFIN

### **CallType**

설명:	함수가 호출된 이유. MQCBCT_* 값 중 하나
PCF 매개변수:	MQIACF_CALL_TYPE
추적 레벨:	1
유형	MQCFIN

### **MsgBuffer**

설명:	메시지 데이터.
PCF 매개변수:	MQBACF_MESSAGE_DATA
추적 레벨:	1
유형	MQCFBS

길이: 길이는 APPTRACE 구성에 설정된 TRACEDATA() 매개변수로 제어됩니다. TRACEDATA=NONE이면 이 매개변수가 생략됩니다.

### **MsgLength**

설명: 메시지의 길이 (MQCBC 구조의 DataLength 필드에서 가져옴).  
PCF 매개변수: MQIACF\_MSG\_LENGTH  
추적 레벨: 1  
유형: MQCFIN

### **HighResTime**

설명: 1970년 1월 1일(UTC), 자정 이후의 조작 시간(마이크로초)  
참고: 이 타이머의 정확도는 고해상도 타이머에 대한 플랫폼 지원에 따라 다릅니다.  
PCF 매개변수: MQIAMO64\_HIGHRES\_TIME  
추적 레벨: 2  
유형: MQCFIN64

### **ReportOptions**

설명: 보고 메시지의 옵션  
PCF 매개변수: MQIACF\_REPORT  
추적 레벨: 2  
유형: MQCFIN

### **MsgType**

설명: 메시지의 유형  
PCF 매개변수: MQIACF\_MSG\_TYPE  
추적 레벨: 2  
유형: MQCFIN

### **Expiry**

설명: 메시지 수명  
PCF 매개변수: MQIACF\_EXPIRY  
추적 레벨: 2  
유형: MQCFIN

### **Format**

설명: 메시지 데이터의 형식 이름  
PCF 매개변수: MQCACH\_FORMAT\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_FORMAT\_LENGTH

### **Priority**

설명:	메시지 우선순위
PCF 매개변수:	MQIACF_PRIORITY
추적 레벨:	2
유형	MQCFIN

### **Persistence**

설명:	메시지 지속성
PCF 매개변수:	MQIACF_PERSISTENCE
추적 레벨:	2
유형	MQCFIN

### **MsgId**

설명:	메시지 ID
PCF 매개변수:	MQBACF_MSG_ID
추적 레벨:	2
유형	MQCFBS
길이:	MQ_MSG_ID_LENGTH

### **CorrelId**

설명:	상관 ID
PCF 매개변수:	MQBACF_CORREL_ID
추적 레벨:	2
유형	MQCFBS
길이:	MQ_CORREL_ID_LENGTH

### **ObjectName**

설명:	열린 오브젝트의 이름.
PCF 매개변수:	MQCACF_OBJECT_NAME
추적 레벨:	2
유형	MQCFST
길이:	MQ_Q_NAME_LENGTH

### **ResolvedQName**

설명:	메시지를 검색한 큐의 로컬 이름.
PCF 매개변수:	MQCACF_RESOLVED_Q_NAME
추적 레벨:	2
유형	MQCFST
길이:	MQ_Q_NAME_LENGTH

### **ReplyToQueue**

설명:	MQ_Q_NAME_LENGTH
PCF 매개변수:	MQCACF_REPLY_TO_Q
추적 레벨:	2
유형	MQCFST

### **ReplyToQMgr**

설명:	MQ_Q_MGR_NAME_LENGTH
PCF 매개변수:	MQCACF_REPLY_TO_Q_MGR
추적 레벨:	2
유형	MQCFST

### **CodedCharSetId**

설명:	메시지 데이터의 문자 세트 ID
PCF 매개변수:	MQIA_CODED_CHAR_SET_ID
추적 레벨:	2
유형	MQCFIN

### **Encoding**

설명:	메시지 데이터의 숫자 인코딩입니다.
PCF 매개변수:	MQIACF_ENCODING
추적 레벨:	2
유형	MQCFIN

### **PutDate**

설명:	MQ_PUT_DATE_LENGTH
PCF 매개변수:	MQCACF_PUT_DATE
추적 레벨:	2
유형	MQCFST

### **PutTime**

설명:	MQ_PUT_TIME_LENGTH
PCF 매개변수:	MQCACF_PUT_TIME
추적 레벨:	2
유형	MQCFST

### **ResolvedQName**

설명:	ResolvedType이 MQOT_Q일 때 ObjectHandle에 참조된 큐 이름.
PCF 매개변수:	MQCACF_RESOLVED_LOCAL_Q_NAME
추적 레벨:	2
유형	MQCFST

길이: MQ\_Q\_NAME\_LENGTH.

### **ResObjectString**

설명: ResolvedType이 MQOT\_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_OBJECT\_STRING  
추적 레벨: 2  
유형: MQCFST  
길이: 길이가 다양합니다.

### **ResolvedType**

설명: ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT\_Q, MQOT\_TOPIC 또는 MQOT\_NONE입니다.  
PCF 매개변수: MQIACF\_RESOLVED\_TYPE  
추적 레벨: 2  
유형: MQCFIN

### **PolicyName**

설명: 이 메시지에 적용된 정책 이름.  
**참고:** AMS 보호 메시지만 해당함  
PCF 매개변수: MQCA\_POLICY\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_OBJECT\_NAME\_LENGTH

### **XmitqMsgId**

설명: 전송 큐 헤더에 있는 메시지의 메시지 ID.  
**참고:** 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함  
PCF 매개변수: MQBACF\_XQH\_MSG\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_MSG\_ID\_LENGTH

### **XmitqCorrelId**

설명: 전송 큐 헤더에 있는 메시지의 상관 ID.  
**참고:** 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함  
PCF 매개변수: MQBACF\_XQH\_CORREL\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_CORREL\_ID\_LENGTH

### ***XmitqPutTime***

설명:	전송 큐 헤더에 있는 메시지의 넣기 시간. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_PUT_TIME
추적 레벨:	2
유형	MQCFST
길이:	MQ_PUT_TIME_LENGTH

### ***XmitqPutDate***

설명:	전송 큐 헤더에 있는 메시지의 넣기 날짜. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_PUT_DATE
추적 레벨:	2
유형	MQCFST
길이:	MQ_PUT_DATE_LENGTH

### ***XmitqRemoteQName***

설명:	전송 큐 헤더에 있는 메시지의 리모트 큐 목적지. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_REMOTE_Q_Name
추적 레벨:	2
유형	MQCFST
길이:	MQ_Q_NAME_LENGTH

### ***XmitqRemoteQMGr***

설명:	전송 큐 헤더에 있는 메시지의 메시지 ID. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_REMOTE_Q_MGR
추적 레벨:	2
유형	MQCFST
길이:	MQ_MSG_ID_LENGTH

### ***MsgDescStructure***

설명:	MQMD 구조. 버전 4 MQGMO를 사용하여 MQMD 대신 메시지 핸들을 리턴하도록 요청한 경우에는 이 매개변수가 생략됩니다.
PCF 매개변수:	MQBACF_MQMD_STRUCT
추적 레벨:	3
유형	MQCFBS
길이:	MQMD 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **GetMsgOptsStructure**

설명:	MQGMO 구조.
PCF 매개변수:	MQBACF_MQGMO_STRUCTURE
추적 레벨:	3
유형	MQCFBS
길이:	MQGMO 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **MQCBCContextStructure**

설명:	MQCBC 구조.
PCF 매개변수:	MQBACF_MQCBC_STRUCTURE
추적 레벨:	3
유형	MQCFBS
길이:	MQCBC 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **MQCB**

애플리케이션이 콜백 MQI 함수 관리를 시작함

#### **CallbackOperation**

설명:	콜백 함수 관리 조작. MQOP_* 값 중 하나로 설정됩니다.
PCF 매개변수:	MQIACF_MQCB_OPERATION
추적 레벨:	1
유형	MQCFIN

#### **CallbackType**

설명:	콜백 함수의 유형(MQCBD 구조의 CallbackType 필드). MQCBT_* 값 중 하나로 설정됩니다.
PCF 매개변수:	MQIACF_MQCB_TYPE
추적 레벨:	1
유형	MQCFIN

#### **CallbackOptions**

설명:	콜백 옵션. MQCBDO_* 값 중 하나로 설정됩니다.
PCF 매개변수:	MQIACF_MQCB_OPTIONS
추적 레벨:	1
유형	MQCFIN

#### **CallbackFunction**

설명:	함수 호출로 시작된 경우 콜백 함수에 대한 포인터.
PCF 매개변수:	MQBACF_MQCB_FUNCTION
추적 레벨:	1
유형	MQCFBS
길이:	MQPTR의 크기

### **CallbackName**

설명:	동적 링크된 프로그램으로 시작된 경우 콜백 함수의 이름.
PCF 매개변수:	MQCACF_MQCB_NAME
추적 레벨:	1
유형	MQCFST
길이:	MQCHAR128의 크기

### **ObjectHandle**

설명:	오브젝트 핸들
PCF 매개변수:	MQIACF_HOBJ
추적 레벨:	1
유형	MQCFIN

### **MaxMsgLength**

설명:	최대 메시지 길이. 정수 또는 특수 값 MQCBD_FULL_MSG_LENGTH로 설정됩니다.
PCF 매개변수:	MQIACH_MAX_MSG_LENGTH
추적 레벨:	2
유형	MQCFIN

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형	MQCFIN

### **ResolvedQName**

설명:	ResolvedType이 MQOT_Q일 때 ObjectHandle에 참조된 큐 이름.
PCF 매개변수:	MQCACF_RESOLVED_LOCAL_Q_NAME
추적 레벨:	2
유형	MQCFST
길이:	MQ_Q_NAME_LENGTH.

### **ResObjectString**

설명:	ResolvedType이 MQOT_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.
PCF 매개변수:	MQCACF_RESOLVED_OBJECT_STRING

추적 레벨:	2
유형	MQCFST
길이:	길이가 다양합니다.

### **ResolvedType**

설명:	ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT_Q, MQOT_TOPIC 또는 MQOT_NONE입니다.
PCF 매개변수:	MQIACF_RESOLVED_TYPE
추적 레벨:	2
유형	MQCFIN

### **Callback DescriptorStructure**

설명:	MQCBD 구조. NULL MQCBC 값이 MQCB 호출에 전달된 경우 이 매개변수는 생략됩니다.
PCF 매개변수:	MQBACF_MQCBD_STRUCT
추적 레벨:	3
유형	MQCFBS
길이:	MQCBC 구조의 길이(바이트)

### **MsgDescStructure**

설명:	MQMD 구조. NULL MQMD 값이 MQCB 호출에 전달된 경우 MsgDescStructure 매개변수는 생략됩니다.
PCF 매개변수:	MQBACF_MQMD_STRUCT
추적 레벨:	3
유형	MQCFBS
길이:	MQMD 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **GetMsgOptsStructure**

설명:	MQGMO 구조. NULL MQGMO 값이 MQCB 호출에 전달된 경우 이 매개변수는 생략됩니다.
PCF 매개변수:	MQBACF_MQGMO_STRUCT
추적 레벨:	3
유형	MQCFBS
길이:	MQGMO 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

## **MQCLOSE**

애플리케이션이 MQCLOSE MQI 함수를 시작함

### **ObjectHandle**

설명:	오브젝트 핸들
PCF 매개변수:	MQIACF_HOBJ
추적 레벨:	1
유형	MQCFIN

### **CloseOptions**

설명:	닫기 옵션
PCF 매개변수:	MQIACF_CLOSE_OPTIONS
추적 레벨:	1
유형	MQCFIN

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형	MQCFIN

### **ResolvedQName**

설명:	ResolvedType이 MQOT_Q일 때 ObjectHandle에 참조된 큐 이름.
PCF 매개변수:	MQCACF_RESOLVED_LOCAL_Q_NAME
추적 레벨:	2
유형	MQCFST
길이:	MQ_Q_NAME_LENGTH.

### **ResObjectString**

설명:	ResolvedType이 MQOT_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.
PCF 매개변수:	MQCACF_RESOLVED_OBJECT_STRING
추적 레벨:	2
유형	MQCFST
길이:	길이가 다양합니다.

### **ResolvedType**

설명:	ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT_Q, MQOT_TOPIC 또는 MQOT_NONE입니다.
PCF 매개변수:	MQIACF_RESOLVED_TYPE
추적 레벨:	2
유형	MQCFIN

### **MQCMIT**

애플리케이션이 MQCMIT MQI 함수를 시작함

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형	MQCFIN

### **MQCONN 및 MQCONNX**

애플리케이션이 MQCONN 또는 MQCONNX MQI 함수를 시작함

### **ConnectionId**

설명:	사용 가능한 경우 연결 ID 또는 그렇지 않으면 MQCONNID_NONE
PCF 매개변수:	MQBACF_CONNECTION_ID
추적 레벨:	1
유형:	MQCFBS
최대 길이:	MQ_CONNECTION_ID_LENGTH

### **QueueManagerName**

설명:	MQCONN(X) 호출에 사용된 큐 관리자의 (분석되지 않은) 이름
PCF 매개변수:	MQCA_Q_MGR_NAME
추적 레벨:	1
유형:	MQCFST
최대 길이:	MQ_Q_MGR_NAME_LENGTH

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

### **ConnectOptions**

설명:	MQCNO_* 값에서 파생된 연결 옵션 참고: MQCONNX만 해당함
PCF 매개변수:	MQIACF_CONNECT_OPTIONS
추적 레벨:	2
유형:	MQCFIN

### **ConnectionOptionsStructure**

설명:	MQCNO 구조. 참고: (MQCONNX만 해당함)
PCF 매개변수:	MQBACF_MQCNO_STRUCT
추적 레벨:	3
유형:	MQCFBS
최대 길이:	MQCNO 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **ChannelDefinitionStructure**

설명:	MQCD 구조. 참고: 클라이언트 연결만 해당함
PCF 매개변수:	MQBACF_MQCD_STRUCT
추적 레벨:	3
유형:	MQCFBS
최대 길이:	MQCD 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **MQCTL**

애플리케이션이 MQCTL MQI 함수를 시작함

#### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

#### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

#### **CtlOperation**

설명:	MQOP_* 값 중 하나
PCF 매개변수:	MQIACF_CTL_OPERATION

추적 레벨: 1  
유형: MQCFIN

### **MQDISC**

애플리케이션이 MQDISC MQI 함수를 시작함

#### **CompCode**

설명: 조작 결과를 나타내는 완료 코드  
PCF 매개변수: MQIACF\_COMP\_CODE  
추적 레벨: 1  
유형: MQCFIN

#### **Reason**

설명: 조작의 이유 코드 결과  
PCF 매개변수: MQIACF\_REASON\_CODE  
추적 레벨: 1  
유형: MQCFIN

### **MQGET**

애플리케이션이 MQGET MQI 함수를 시작함

#### **ObjectHandle**

설명: 오브젝트 핸들  
PCF 매개변수: MQIACF\_HOBJ  
추적 레벨: 1  
유형: MQCFIN

#### **GetOptions**

설명: MQGMO.Options의 가져오기 옵션  
PCF 매개변수: MQIACF\_GET\_OPTIONS  
추적 레벨: 1  
유형: MQCFIN

#### **CompCode**

설명: 조작 결과를 나타내는 완료 코드  
PCF 매개변수: MQIACF\_COMP\_CODE  
추적 레벨: 1  
유형: MQCFIN

#### **Reason**

설명: 조작의 이유 코드 결과  
PCF 매개변수: MQIACF\_REASON\_CODE  
추적 레벨: 1  
유형: MQCFIN

### **MsgBuffer**

설명:	메시지 데이터. TRACEDATA=NONE이면 이 매개변수가 생략됩니다.
PCF 매개변수:	MQBACF_MESSAGE_DATA
추적 레벨:	1
유형:	MQCFBS
최대 길이:	길이는 APPTRACE 구성에 설정된 TRACEDATA() 매개변수로 제어됩니다. (추적 메시지에 MQIACF_TRACE_DATA_LENGTH로 포함됨).

### **MsgLength**

설명:	메시지의 길이
PCF 매개변수:	MQIACF_MSG_LENGTH
추적 레벨:	1
유형:	MQCFIN

### **HighResTime**

설명:	1970년 1월 1일(UTC), 자정 이후의 조작 시간(마이크로초) <b>참고:</b> 이 타이머의 정확도는 고해상도 타이머에 대한 플랫폼 지원에 따라 다릅니다.
PCF 매개변수:	MQIAMO64_HIGHRES_TIME
추적 레벨:	2
유형:	MQCFIN64

### **BufferLength**

설명:	애플리케이션이 제공하는 버퍼의 길이
PCF 매개변수:	MQIACF_BUFFER_LENGTH
추적 레벨:	2
유형:	MQCFIN

### **ObjectName**

설명:	열린 오브젝트의 이름
PCF 매개변수:	MQCACF_OBJECT_NAME
추적 레벨:	2
유형:	MQCFST
길이:	MQ_Q_NAME_LENGTH

### **ResolvedQName**

설명:	메시지를 검색한 큐의 로컬 이름.
PCF 매개변수:	MQCACF_RESOLVED_Q_NAME
추적 레벨:	2
유형:	MQCFST
최대 길이:	MQ_Q_NAME_LENGTH

### **ReportOptions**

설명:	메시지 보고 옵션
PCF 매개변수:	MQIACF_REPORT
추적 레벨:	2
유형:	MQCFIN

### **MsgType**

설명:	메시지의 유형
PCF 매개변수:	MQIACF_MSG_TYPE
추적 레벨:	2
유형:	MQCFIN

### **Expiry**

설명:	메시지 수명
PCF 매개변수:	MQIACF_EXPIRY
추적 레벨:	2
유형:	MQCFIN

### **Format**

설명:	메시지 데이터의 형식 이름
PCF 매개변수:	MQCACH_FORMAT_NAME
추적 레벨:	2
유형:	MQCFST
최대 길이:	MQ_FORMAT_LENGTH

### **Priority**

설명:	메시지 우선순위
PCF 매개변수:	MQIACF_PRIORITY
추적 레벨:	2
유형:	MQCFIN

### **Persistence**

설명:	메시지 지속성
PCF 매개변수:	MQIACF_PERSISTENCE
추적 레벨:	2
유형:	MQCFIN

### **MsgId**

설명:	메시지 ID
PCF 매개변수:	MQBACF_MSG_ID
추적 레벨:	2

유형: MQCFBS  
최대 길이: MQ\_MSG\_ID\_LENGTH

### ***CorrelId***

설명: 상관 ID  
PCF 매개변수: MQBACF\_CORREL\_ID  
추적 레벨: 2  
유형: MQCFBS  
최대 길이: MQ\_CORREL\_ID\_LENGTH

### ***ReplyToQueue***

설명:  
PCF 매개변수: MQCACF\_REPLY\_TO\_Q  
추적 레벨: 2  
유형: MQCFST  
최대 길이: MQ\_Q\_NAME\_LENGTH

### ***ReplyToQMgr***

설명:  
PCF 매개변수: MQCACF\_REPLY\_TO\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH

### ***CodedCharSetId***

설명: 메시지 데이터의 문자 세트 ID  
PCF 매개변수: MQIA\_CODED\_CHAR\_SET\_ID  
추적 레벨: 2  
유형: MQCFIN

### ***Encoding***

설명: 메시지 데이터의 숫자 인코딩입니다.  
PCF 매개변수: MQIACF\_ENCODING  
추적 레벨: 2  
유형: MQCFIN

### ***PutDate***

설명:  
PCF 매개변수: MQCACF\_PUT\_DATE  
추적 레벨: 2  
유형: MQCFST

최대 길이: MQ\_PUT\_DATE\_LENGTH

### **PutTime**

설명:

PCF 매개변수: MQCACF\_PUT\_TIME

추적 레벨: 2

유형: MQCFST

최대 길이: MQ\_PUT\_TIME\_LENGTH

### **ResolvedQName**

설명: ResolvedType이 MQOT\_Q일 때 ObjectHandle에 참조된 큐 이름.

PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_NAME

추적 레벨: 2

유형: MQCFST

길이: MQ\_Q\_NAME\_LENGTH.

### **ResObjectString**

설명: ResolvedType이 MQOT\_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.

PCF 매개변수: MQCACF\_RESOLVED\_OBJECT\_STRING

추적 레벨: 2

유형: MQCFST

길이: 길이가 다양합니다.

### **ResolvedType**

설명: ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT\_Q, MQOT\_TOPIC 또는 MQOT\_NONE입니다.

PCF 매개변수: MQIACF\_RESOLVED\_TYPE

추적 레벨: 2

유형: MQCFIN

### **PolicyName**

설명: 이 메시지에 적용된 정책 이름.

**참고:** AMS 보호 메시지만 해당함

PCF 매개변수: MQCA\_POLICY\_NAME

추적 레벨: 2

유형: MQCFST

길이: MQ\_OBJECT\_NAME\_LENGTH

### **XmitqMsgId**

설명: 전송 큐 헤더에 있는 메시지의 메시지 ID.

**참고:** 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQBACF\_XQH\_MSG\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_MSG\_ID\_LENGTH

### ***XmitqCorrelId***

설명: 전송 큐 헤더에 있는 메시지의 상관 ID.  
참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQBACF\_XQH\_CORREL\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_CORREL\_ID\_LENGTH

### ***XmitqPutTime***

설명: 전송 큐 헤더에 있는 메시지의 넣기 시간.  
참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_PUT\_TIME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_PUT\_TIME\_LENGTH

### ***XmitqPutDate***

설명: 전송 큐 헤더에 있는 메시지의 넣기 날짜.  
참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_PUT\_DATE  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_PUT\_DATE\_LENGTH

### ***XmitqRemoteQName***

설명: 전송 큐 헤더에 있는 메시지의 리모트 큐 목적지.  
참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_REMOTE\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

### ***XmitqRemoteQMgr***

설명: 전송 큐 헤더에 있는 메시지의 리모트 큐 관리자 목적지.  
참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_REMOTE\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

**MsgDescStructure**

설명: MQMD 구조.  
PCF 매개변수: MQBACF\_MQMD\_STRUCT  
추적 레벨: 3  
유형: MQCFBS  
최대 길이: MQMD 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

**GetMsgOptsStructure**

설명: MQGMO 구조.  
PCF 매개변수: MQBACF\_MQGMO\_STRUCT  
추적 레벨: 3  
유형: MQCFBS  
최대 길이: MQGMO 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

**MQINQ**

애플리케이션이 MQINQ MQI 함수를 시작함

**ObjectHandle**

설명: 오브젝트 핸들  
PCF 매개변수: MQIACF\_HOBJ  
추적 레벨: 1  
유형: MQCFIN

**CompCode**

설명: 조작 결과를 나타내는 완료 코드  
PCF 매개변수: MQIACF\_COMP\_CODE  
추적 레벨: 1  
유형: MQCFIN

**Reason**

설명: 조작의 이유 코드 결과  
PCF 매개변수: MQIACF\_REASON\_CODE  
추적 레벨: 1  
유형: MQCFIN

**SelectorCount**

설명: Selectors 배열에 제공되는 선택자 수.

PCF 매개변수: MQIACF\_SELECTOR\_COUNT  
추적 레벨: 2  
유형: MQCFIN

### **Selectors**

설명: MQINQ가 값을 리턴해야 하는 속성의 목록(정수 또는 문자).  
PCF 매개변수: MQIACF\_SELECTORS  
추적 레벨: 2  
유형: MQCFIL

### **ResolvedQName**

설명: ResolvedType이 MQOT\_Q일 때 ObjectHandle에 참조된 큐 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
최대 길이: MQ\_Q\_NAME\_LENGTH

### **ResObjectString**

설명: ResolvedType이 MQOT\_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_OBJECT\_STRING  
추적 레벨: 2  
유형: MQCFST  
최대 길이: 길이가 다양합니다.

### **ResolvedType**

설명: ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT\_Q, MQOT\_TOPIC 또는 MQOT\_NONE입니다.  
PCF 매개변수: MQIACF\_RESOLVED\_TYPE  
추적 레벨: 2  
유형: MQCFIN

### **IntAttrCount**

설명: 조회 조작이 리턴하는 정수 속성의 수  
PCF 매개변수: MQIACF\_INTATTR\_COUNT  
추적 레벨: 3  
유형: MQCFIN

### **IntAttrrs**

설명: 조회 조작이 리턴하는 정수 속성 값. 이 매개변수는 MQINQ가 리턴할 때 IntAttrCount > 0인 경우에만 존재합니다.  
PCF 매개변수: MQIACF\_INT\_ATTRS  
추적 레벨: 3

유형: MQCFIL

### **CharAttrs**

설명: 조회 조작이 리턴하는 문자 속성. 값이 서로 연결됩니다. 이 매개변수는 MQINQ가 리턴할 때 CharAttrLength > 0인 경우에만 포함됩니다.

PCF 매개변수: MQCACF\_CHAR\_ATTRS

추적 레벨: 3

유형: MQCFST

### **MQOPEN**

애플리케이션이 MQOPEN MQI 함수를 시작함

### **ObjectType**

설명: MQOT.ObjectType에 전달된 오브젝트 유형

PCF 매개변수: MQIACF\_OBJECT\_TYPE

추적 레벨: 1

유형: MQCFIN

### **ObjectName**

설명: 큐 이름 분석을 시도하기 전에 MQI 호출에 전달된 오브젝트의 이름.

PCF 매개변수: MQCACF\_OBJECT\_NAME

추적 레벨: 1

유형: MQCFST

최대 길이: MQ\_Q\_NAME\_LENGTH

### **ObjectQMgrName**

설명: 큐 이름 분석을 시도하기 전에 MQI 호출에 전달된 오브젝트 큐 관리자의 이름.

PCF 매개변수: MQCACF\_OBJECT\_Q\_MGR\_NAME

추적 레벨: 1

유형: MQCFST

최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH

### **ObjectHandle**

설명: 오브젝트 핸들

PCF 매개변수: MQIACF\_HOBJ

추적 레벨: 1

유형: MQCFIN

### **CompCode**

설명: 조작 결과를 나타내는 완료 코드

PCF 매개변수: MQIACF\_COMP\_CODE

추적 레벨: 1

유형: MQCFIN

### **Reason**

설명: 조작의 이유 코드 결과  
PCF 매개변수: MQIACF\_REASON\_CODE  
추적 레벨: 1  
유형: MQCFIN

### **OpenOptions**

설명: 오브젝트를 여는 데 사용하는 옵션  
PCF 매개변수: MQIACF\_OPEN\_OPTIONS  
추적 레벨: 1  
유형: MQCFIN

### **AlternateUserId**

설명: MQOO\_ALTERNATE\_USER\_AUTHORITY가 지정된 경우에만 포함됩니다.  
PCF 매개변수: MQCACF\_ALTERNATE\_USERID  
추적 레벨: 2  
유형: MQCFST  
최대 길이: MQ\_USER\_ID\_LENGTH

### **RecsPresent**

설명: 오브젝트 이름 레코드의 수가 표시됩니다. MQOD 버전  $\geq$  MQOD\_VERSION\_2인 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_RECS\_PRESENT  
추적 레벨: 1  
유형: MQCFIN

### **KnownDestCount**

설명: 열기에 성공한 로컬 큐 수. MQOD 버전  $\geq$  MQOD\_VERSION\_2인 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_KNOWN\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **UnknownDestCount**

설명: 열기에 성공한 리모트 큐 수. MQOD 버전  $\geq$  MQOD\_VERSION\_2인 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_UNKNOWN\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **InvalidDestCount**

설명: 열기에 실패한 큐 수. MQOD 버전  $\geq$  MQOD\_VERSION\_2인 경우에만 포함됩니다.

PCF 매개변수: MQIACF\_INVALID\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### ***DynamicQName***

설명: MQOPEN 호출에 입력으로 전달된 동적 큐 이름.  
PCF 매개변수: MQCACF\_DYNAMIC\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
최대 길이: MQ\_Q\_NAME\_LENGTH

### ***ResolvedLocalQName<sup>12</sup>***

설명: 이름 분석이 수행된 후의 로컬 큐 이름을 포함합니다. (예를 들어, 리모트 큐의 경우 전송 큐의 이름입니다.)  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
범위: MQOD.Version이 MQOD\_VERSION\_3 미만이면 MQOPEN 호출이 완료된 후의 MQOD.ObjectName 필드 값을 포함합니다. MQOD.Version이 MQOD\_VERSION\_3 이상이면 MQOD의 값이 포함됩니다. ResolvedQName 필드,  
최대 길이: MQ\_Q\_NAME\_LENGTH

### ***ResolvedLocalQMgrName<sup>12</sup>***

설명: 이름 분석이 수행된 후의 로컬 큐 관리자 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
범위: MQOD 버전 >= MQOD\_VERSION\_3인 경우에만 해당됩니다.  
최대 길이: MQ\_Q\_MGR\_NAME\_LENGTH

### ***ResolvedQName<sup>12</sup>***

설명: 이름 분석이 수행된 후의 큐 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
범위: MQOD.Version이 MQOD\_VERSION\_3 미만이면 MQOPEN 호출이 완료된 후의 MQOD.ObjectName 필드 값을 포함합니다. MQOD.Version이 MQOD\_VERSION\_3 이상이면 MQOD의 값이 포함됩니다. ResolvedQName 필드,  
최대 길이: MQ\_Q\_NAME\_LENGTH

### **ResolvedQMgrName<sup>12</sup>**

설명:	이름 분석이 수행된 후의 큐 관리자 이름을 포함합니다. MQOD.Version이 MQOD_VERSION_3 미만이면 MQOD의 값이 포함됩니다. MQOPEN 호출 다음의 ObjectQMgrName 필드가 완료되었습니다. MQOD.Version이 MQOD_VERSION_3 이상이면 MQOD의 값이 포함됩니다. ResolvedQMgrName 필드
PCF 매개변수:	MQCACF_RESOLVED_Q_MGR
추적 레벨:	2
유형:	MQCFST
최대 길이:	MQ_Q_MGR_NAME_LENGTH

### **AlternateSecurityId**

설명:	대체 보안 ID. MQOD.Version이 MQOD_VERSION_3 이상이고, MQOO_ALTERNATE_USER_AUTHORITY가 지정되어 있으며, MQOD.AlternateSecurityId가 MQSID_NONE과 동일하지 않을 때에만 표시됩니다.
PCF 매개변수:	MQBACF_ALTERNATE_SECURITYID
추적 레벨:	2
유형:	MQCFBS
최대 길이:	MQ_SECURITY_ID_LENGTH

### **ObjectString**

설명:	긴 오브젝트 이름. MQOD.Version이 MQOD_VERSION_4 이상이고 MQOD.ObjectString의 VSLength 필드가 MQVS_NULL_TERMINATED 또는 0보다 큰 경우에만 포함됩니다.
PCF 매개변수:	MQCACF_OBJECT_STRING
추적 레벨:	2
유형:	MQCFST
최대 길이:	길이가 다양합니다.

### **SelectionString**

설명:	선택 문자열. MQOD.Version이 MQOD_VERSION_4 이상이고 VSLength 필드가 MQOD와 같은 경우에만 포함됩니다. SelectionString은 MQVS_NULL_TERMINATED이거나 0보다 큼니다.
PCF 매개변수:	MQCACF_SELECTION_STRING
추적 레벨:	2
유형:	MQCFST
최대 길이:	길이가 다양합니다.

### **ResObjectString**

설명:	큐 관리자가 ObjectName 필드에 제공된 이름을 분석한 후의 긴 오브젝트 이름. MQOD.Version이 MQOD_VERSION_4 이상이고 VSLength가 MQVS_NULL_TERMINATED 또는 0보다 큰 경우에만 토픽 오브젝트를 참조하는 토픽 및 큐 알리어스에 대해 포함됩니다.
PCF 매개변수:	MQCACF_RESOLVED_OBJECT_STRING
추적 레벨:	2

유형: MQCFST  
 최대 길이: 길이가 다양합니다.

**ResolvedType**

설명: 여는 해석된(기본) 오브젝트의 유형입니다. MQOD.Version이 MQOD\_VERSION\_4 이상인 경우에만 포함됩니다. 가능한 값은 MQOT\_Q, MQOT\_TOPIC 또는 MQOT\_NONE 입니다.

PCF 매개변수: MQIACF\_RESOLVED\_TYPE

추적 레벨: 2

유형: MQCFIN

애플리케이션 활동 분배 목록 PCF 그룹 헤더 구조

MQOPEN 함수가 분배 목록을 여는 경우 MQOPEN 매개변수는 RecsPresent에 번호 지정된 구조 수까지, 분배 목록의 각 큐에 대한 AppActivityDistList PCF 그룹을 하나씩 포함합니다. Ap-pActivityDistList PCF 그룹은 MQOR 및 MQRR 구조의 정보를 결합하여 큐 이름을 식별하고 큐에 대한 열기 조작의 결과를 표시합니다. AppActivityDistList 그룹은 항상 다음 MQCFGR 구조로 시작합니다.

표 28. AppActivityDistList 그룹 MQCFGR 구조		
MQCFGR 필드	가치	설명
유형	MQCFT_GROUP	
StrucLength	MQCFGR 구조의 길이(바이트)	
매개변수	MQGACF_APP_DIST_LIST	분배 목록 그룹 매개변수
ParameterCount	4	이 그룹 내에 포함된 MQCFGR 구조를 따르는 매개변수 구조의 수.

**ObjectName**

설명: 분배 목록 MQ\_Q\_NAME\_LENGTH의 큐 이름. MQOR 구조가 제공된 경우에만 포함됩니다.

PCF 매개변수: MQCACF\_OBJECT\_NAME

추적 레벨: 2

유형: MQCFST

길이: MQ\_Q\_NAME\_LENGTH. MQOR 구조가 제공된 경우에만 포함됩니다.

**ObjectQMgrName**

설명: ObjectName에 이름 지정된 큐가 정의되어 있는 큐 관리자의 이름.

PCF 매개변수: MQCACF\_OBJECT\_Q\_MGR\_NAME

추적 레벨: 2

유형: MQCFST

길이: MQ\_Q\_MGR\_NAME\_LENGTH. MQOR 구조가 제공된 경우에만 포함됩니다.

<sup>1</sup> 이 매개변수는 열린 오브젝트가 큐로 해석되고 MQOO\_INPUT\_\*, MQOO\_OUTPUT 또는 MQOO\_BROWSE에 대해 큐가 열린 경우에만 포함됩니다.

<sup>2</sup> ResolvedLocalQName 매개변수는 ResolvedQName 매개변수와 다른 경우에만 포함됩니다.

### **CompCode**

설명:	이 오브젝트에 대한 열기 결과를 나타내는 완료 코드. MQRR 구조가 제공되고 MQOPEN의 이유 코드가 MQRC_MULTIPLE_REASONS인 경우에만 포함됩니다.
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	2
유형:	MQCFIN

### **Reason**

설명:	이 오브젝트에 대한 열기 결과를 나타내는 이유 코드. MQRR 구조가 제공되고 MQOPEN의 이유 코드가 MQRC_MULTIPLE_REASONS인 경우에만 포함됩니다.
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	2
유형:	MQCFIN

### **MQPUT**

애플리케이션이 MQPUT MQI 함수를 시작했습니다.

### **ObjectHandle**

설명:	오브젝트 핸들
PCF 매개변수:	MQIACF_HOBJ
추적 레벨:	1
유형:	MQCFIN

### **PutOptions**

설명:	MQPMO.Options의 넣기 옵션
PCF 매개변수:	MQIACF_PUT_OPTIONS
추적 레벨:	1
유형:	MQCFIN

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

### **MsgBuffer**

설명:	메시지 데이터.
-----	----------

PCF 매개변수: MQBACF\_MESSAGE\_DATA  
추적 레벨: 1  
유형: MQCFBS  
길이: 길이는 APPTRACE 구성에 설정된 TRACEDATA() 매개변수로 제어됩니다.  
TRACEDATA=NONE이면 이 매개변수가 생략됩니다.

### **MsgLength**

설명: 메시지의 길이  
PCF 매개변수: MQIACF\_MSG\_LENGTH  
추적 레벨: 1  
유형: MQCFIN

### **RecsPresent**

설명: 레코드 또는 응답 레코드가 있는 메시지 넣기 수. MQPMO 버전 >= MQPMO\_VERSION\_2인 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_RECS\_PRESENT  
추적 레벨: 1  
유형: MQCFIN

### **KnownDestCount**

설명: 로컬 큐에 성공적으로 전송된 메시지 수  
PCF 매개변수: MQIACF\_KNOWN\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **UnknownDestCount**

설명: 리모트 큐에 성공적으로 전송된 메시지 수  
PCF 매개변수: MQIACF\_UNKNOWN\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **InvalidDestCount**

설명: 전송할 수 없는 메시지 수  
PCF 매개변수: MQIACF\_INVALID\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **HighResTime**

설명: 1970년 1월 1일(UTC), 자정 이후의 조작 시간(마이크로초)  
참고: 이 타이머의 정확도는 고해상도 타이머에 대한 플랫폼 지원에 따라 다릅니다.  
PCF 매개변수: MQIAMO64\_HIGHRES\_TIME  
추적 레벨: 2

유형: MQCFIN64

### **ObjectName**

설명: 열린 오브젝트의 이름.  
PCF 매개변수: MQCACF\_OBJECT\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

### **ResolvedQName**

설명: 큐 이름 분석이 수행된 후의 큐 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

### **ResolvedQMgrName**

설명: 이름 분석이 수행된 후의 큐 관리자 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_MGR\_NAME\_LENGTH

### **ResolvedLocalQName<sup>3</sup>**

설명: 이름 분석이 수행된 후의 로컬 큐 이름을 포함합니다.  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST

### **ResolvedLocalQMgrName<sup>3</sup>**

설명: 이름 분석이 수행된 후의 로컬 큐 관리자 이름을 포함합니다.  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_MGR\_NAME\_LENGTH

### **ReportOptions**

설명: 메시지 보고 옵션  
PCF 매개변수: MQIACF\_REPORT  
추적 레벨: 2  
유형: MQCFIN

### **MsgType**

설명:	메시지의 유형
PCF 매개변수:	MQIACF_MSG_TYPE
추적 레벨:	2
유형:	MQCFIN

### **Expiry**

설명:	메시지 수명
PCF 매개변수:	MQIACF_EXPIRY
추적 레벨:	2
유형:	MQCFIN

### **Format**

설명:	메시지 데이터의 형식 이름
PCF 매개변수:	MQCACH_FORMAT_NAME
추적 레벨:	2
유형:	MQCFST
길이:	MQ_FORMAT_LENGTH

### **Priority**

설명:	메시지 우선순위
PCF 매개변수:	MQIACF_PRIORITY
추적 레벨:	2
유형:	MQCFIN

### **Persistence**

설명:	메시지 지속성
PCF 매개변수:	MQIACF_PERSISTENCE
추적 레벨:	2
유형:	MQCFIN

### **MsgId**

설명:	메시지 ID
PCF 매개변수:	MQBACF_MSG_ID
추적 레벨:	2
유형:	MQCFBS
길이:	MQ_MSG_ID_LENGTH

### **CorrelId**

설명:	상관 ID
PCF 매개변수:	MQBACF_CORREL_ID

추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_CORREL\_ID\_LENGTH

### ***ReplyToQueue***

설명:  
PCF 매개변수: MQCACF\_REPLY\_TO\_Q  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

### ***ReplyToQMgr***

설명:  
PCF 매개변수: MQCACF\_REPLY\_TO\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_MGR\_NAME\_LENGTH

### ***CodedCharSetId***

설명: 메시지 데이터의 문자 세트 ID  
PCF 매개변수: MQIA\_CODED\_CHAR\_SET\_ID  
추적 레벨: 2  
유형: MQCFIN

### ***Encoding***

설명: 메시지 데이터의 숫자 인코딩입니다.  
PCF 매개변수: MQIACF\_ENCODING  
추적 레벨: 2  
유형: MQCFIN

### ***PutDate***

설명:  
PCF 매개변수: MQCACF\_PUT\_DATE  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_PUT\_DATE\_LENGTH

### ***PutTime***

설명:  
PCF 매개변수: MQCACF\_PUT\_TIME  
추적 레벨: 2

유형: MQCFST  
길이: MQ\_PUT\_TIME\_LENGTH

### **ResolvedQName**

설명: ResolvedType이 MQOT\_Q일 때 ObjectHandle에 참조된 큐 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH.

### **ResObjectString**

설명: ResolvedType이 MQOT\_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_OBJECT\_STRING  
추적 레벨: 2  
유형: MQCFST  
길이: 길이가 다양합니다.

### **ResolvedType**

설명: ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT\_Q, MQOT\_TOPIC 또는 MQOT\_NONE입니다.  
PCF 매개변수: MQIACF\_RESOLVED\_TYPE  
추적 레벨: 2  
유형: MQCFIN

### **PolicyName**

설명: 이 메시지에 적용된 정책 이름.  
**참고:** AMS 보호 메시지만 해당함  
PCF 매개변수: MQCA\_POLICY\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_OBJECT\_NAME\_LENGTH

### **XmitqMsgId**

설명: 전송 큐 헤더에 있는 메시지의 메시지 ID.  
**참고:** 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함  
PCF 매개변수: MQBACF\_XQH\_MSG\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_MSG\_ID\_LENGTH

### ***XmitqCorrelId***

설명:	전송 큐 헤더에 있는 메시지의 상관 ID. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQBACF_XQH_CORREL_ID
추적 레벨:	2
유형:	MQCFBS
길이:	MQ_CORREL_ID_LENGTH

### ***XmitqPutTime***

설명:	전송 큐 헤더에 있는 메시지의 넣기 시간. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_PUT_TIME
추적 레벨:	2
유형:	MQCFST
길이:	MQ_PUT_TIME_LENGTH

### ***XmitqPutDate***

설명:	전송 큐 헤더에 있는 메시지의 넣기 날짜. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_PUT_DATE
추적 레벨:	2
유형:	MQCFST
길이:	MQ_PUT_DATE_LENGTH

### ***XmitqRemoteQName***

설명:	전송 큐 헤더에 있는 메시지의 리모트 큐 목적지. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_REMOTE_Q_NAME
추적 레벨:	2
유형:	MQCFST
길이:	MQ_Q_NAME_LENGTH

### ***XmitqRemoteQMgr***

설명:	전송 큐 헤더에 있는 메시지의 리모트 큐 관리자 목적지. <b>참고:</b> 형식이 MQFMT_XMIT_Q_HEADER일 때에만 해당함
PCF 매개변수:	MQCACF_XQH_REMOTE_Q_MGR
추적 레벨:	2
유형:	MQCFST
길이:	MQ_Q_NAME_LENGTH

## **PutMsgOptsStructure**

설명:	MQPMO 구조.
PCF 매개변수:	MQBACF_MQPMO_STRUCT
추적 레벨:	3
유형:	MQCFBS
길이:	MQPMO 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **MQPUT 애플리케이션 활동 분배 목록 PCF 그룹 헤더 구조**

MQPUT 함수가 분배 목록에 넣기를 수행 중이면 MQPUT 매개변수가 AppCompatActivityDistList PCF 그룹을 하나 포함합니다. 분배 목록의 각 큐에 대해서는 219 페이지의 『애플리케이션 활동 분배 목록 PCF 그룹 헤더 구조』의 내용을 참조하십시오. AppCompatActivityDistList PCF 그룹은 MQPMR 및 MQRR 구조의 정보를 결합하여 PUT 매개변수를 식별하고 각 큐에 대한 PUT 조작의 결과를 표시합니다. MQPUT 조작의 경우 AppCompatActivityDistList 그룹은 다음의 모든 또는 일부 매개변수를 포함합니다(이유 코드가 MQRC\_MULTIPLE\_REASONS이고 MQPMO.PutMsgRecFields 필드를 통해 다른 매개변수가 판별되면 CompCode 및 Reason 필드가 존재함).

#### **CompCode**

설명:	조작 결과를 나타내는 완료 코드. MQRR 구조가 제공되고 MQPUT의 이유 코드가 MQRC_MULTIPLE_REASONS인 경우에만 포함됩니다.
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	2
유형:	MQCFIN

#### **Reason**

설명:	이 오브젝트에 대한 넣기 결과를 나타내는 이유 코드. MQRR 구조가 제공되고 MQPUT의 이유 코드가 MQRC_MULTIPLE_REASONS인 경우에만 포함됩니다.
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	2
유형:	MQCFIN

#### **MsgId**

설명:	메시지 ID. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF_MSG_ID가 있는 경우에만 포함됩니다.
PCF 매개변수:	MQBACF_MSG_ID
추적 레벨:	2
유형:	MQCFBS
길이:	MQ_MSG_ID_LENGTH

#### **CorrelId**

설명:	상관 ID. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF_CORREL_ID가 있는 경우에만 포함됩니다.
PCF 매개변수:	MQBACF_CORREL_ID
추적 레벨:	2

<sup>3</sup> ResolvedLocalQName 매개변수는 ResolvedQName 매개변수와 다른 경우에만 포함됩니다.

유형: MQCFBS  
길이: MQ\_CORREL\_ID\_LENGTH

### **GroupId**

설명: 그룹 ID. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_GROUP\_ID가 있는 경우에만 포함됩니다.  
PCF 매개변수: MQBACF\_GROUP\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_GROUP\_ID\_LENGTH

### **Feedback**

설명: 피드백. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_FEEDBACK이 있는 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_FEEDBACK  
추적 레벨: 2  
유형: MQCFIN

### **AccountingToken**

설명: AccountingToken. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_ACCOUNTING\_TOKEN이 있는 경우에만 포함됩니다.  
PCF 매개변수: MQBACF\_ACCOUNTING\_TOKEN  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_ACCOUNTING\_TOKEN\_LENGTH.

### **MQPUT1**

애플리케이션이 MQPUT1 MQI 함수를 시작함

### **ObjectType**

설명: MQOT.ObjectType에 전달된 오브젝트 유형  
PCF 매개변수: MQIACF\_OBJECT\_TYPE  
추적 레벨: 1  
유형: MQCFIN

### **ObjectName**

설명: 큐 이름 분석을 시도하기 전에 MQI 호출에 전달된 오브젝트의 이름.  
PCF 매개변수: MQCACF\_OBJECT\_NAME  
추적 레벨: 1  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

### **ObjectQMgrName**

설명:	큐 이름 분석을 시도하기 전에 MQI 호출에 전달된 오브젝트 큐 관리자의 이름.
PCF 매개변수:	MQCACF_OBJECT_Q_MGR_NAME
추적 레벨:	1
유형:	MQCFST
길이:	MQ_Q_MGR_NAME_LENGTH

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

### **PutOptions**

설명:	MQPMO.Options의 넣기 옵션
PCF 매개변수:	MQIACF_PUT_OPTIONS
추적 레벨:	1
유형:	MQCFIN

### **AlternateUserId**

설명:	MQPMO_ALTERNATE_USER_AUTHORITY가 지정된 경우에만 포함됩니다.
PCF 매개변수:	MQCACF_ALTERNATE_USERID
추적 레벨:	2
유형:	MQCFST
길이:	MQ_USER_ID_LENGTH

### **RecsPresent**

설명:	오브젝트 이름 레코드의 수가 표시됩니다.
PCF 매개변수:	MQIACF_RECS_PRESENT
추적 레벨:	1
유형:	MQCFIN

### **KnownDestCount**

설명:	열린 로컬 큐의 수
PCF 매개변수:	MQIACF_KNOWN_DEST_COUNT

추적 레벨: 1  
유형: MQCFIN

### **UnknownDestCount**

설명: 열린 리모트 큐의 수  
PCF 매개변수: MQIACF\_UNKNOWN\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **InvalidDestCount**

설명: 여는 데 실패한 큐의 수  
PCF 매개변수: MQIACF\_INVALID\_DEST\_COUNT  
추적 레벨: 1  
유형: MQCFIN

### **MsgBuffer**

설명: 메시지 데이터.  
PCF 매개변수: MQBACF\_MESSAGE\_DATA  
추적 레벨: 1  
유형: MQCFBS  
길이: 길이는 APPTRACE 구성에 설정된 TRACEDATA() 매개변수로 제어됩니다. TRACEDATA=NONE이면 이 매개변수가 생략됩니다.

### **MsgLength**

설명: 메시지의 길이  
PCF 매개변수: MQIACF\_MSG\_LENGTH  
추적 레벨: 1  
유형: MQCFIN

### **HighResTime**

설명: 1970년 1월 1일(UTC), 자정 이후의 조작 시간(마이크로초)  
**참고:** 이 타이머의 정확도는 고해상도 타이머에 대한 플랫폼 지원에 따라 다릅니다.  
PCF 매개변수: MQIAMO64\_HIGHRES\_TIME  
추적 레벨: 2  
유형: MQCFIN64

### **ResolvedQName**

설명: 큐 이름 분석이 수행된 후의 큐 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST

길이: MQ\_Q\_NAME\_LENGTH

#### **ResolvedQMgrName**

설명: 이름 분석이 수행된 후의 큐 관리자 이름.  
PCF 매개변수: MQCACF\_RESOLVED\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_MGR\_NAME\_LENGTH

#### **ResolvedLocalQName<sup>4</sup>**

설명: 이름 분석이 수행된 후의 로컬 큐 이름을 포함합니다.  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_NAME  
추적 레벨: 2  
유형: MQCFST

#### **ResolvedLocalQMgrName<sup>4</sup>**

설명: 이름 분석이 수행된 후의 로컬 큐 관리자 이름을 포함합니다.  
PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_MGR  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_MGR\_NAME\_LENGTH

#### **AlternateSecurityId**

설명: 대체 보안 ID. MQOD.Version이 MQOD\_VERSION\_3 이상이고 MQOD.AlternateSecurityId가 MQSID\_NONE과 동일하지 않을 때에만 표시됩니다.  
PCF 매개변수: MQBACF\_ALTERNATE\_SECURITYID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_SECURITY\_ID\_LENGTH

#### **ObjectString**

설명: 긴 오브젝트 이름. MQOD.Version이 MQOD\_VERSION\_4 이상이고 MQOD.ObjectString의 VSLength 필드가 MQVS\_NULL\_TERMINATED 또는 0보다 큰 경우에만 포함됩니다.  
PCF 매개변수: MQCACF\_OBJECT\_STRING  
추적 레벨: 2  
유형: MQCFST  
길이: 길이가 다양합니다.

### **ResObjectString**

설명:	큐 관리자가 ObjectName 필드에 제공된 이름을 분석한 후의 긴 오브젝트 이름. MQOD.Version이 MQOD_VERSION_4 이상이고 VSLength가 MQVS_NULL_TERMINATED 또는 0보다 큰 경우에만 토픽 오브젝트를 참조하는 토픽 및 큐 알리어스에 대해 포함됩니다.
PCF 매개변수:	MQCACF_RESOLVED_OBJECT_STRING
추적 레벨:	2
유형:	MQCFST
길이:	길이가 다양합니다.

### **ResolvedType**

설명:	여는 해석된(기본) 오브젝트의 유형입니다. MQOD.Version이 MQOD_VERSION_4 이상인 경우에만 포함됩니다. 가능한 값은 MQOT_Q, MQOT_TOPIC 또는 MQOT_NONE 입니다.
PCF 매개변수:	MQIACF_RESOLVED_TYPE
추적 레벨:	2
유형:	MQCFIN

### **ReportOptions**

설명:	메시지 보고 옵션
PCF 매개변수:	MQIACF_REPORT
추적 레벨:	2
유형:	MQCFIN

### **MsgType**

설명:	메시지의 유형
PCF 매개변수:	MQIACF_MSG_TYPE
추적 레벨:	2
유형:	MQCFIN

### **Expiry**

설명:	메시지 수명
PCF 매개변수:	MQIACF_EXPIRY
추적 레벨:	2
유형:	MQCFIN

### **Format**

설명:	메시지 데이터의 형식 이름
PCF 매개변수:	MQCACH_FORMAT_NAME
추적 레벨:	2
유형:	MQCFST
길이:	MQ_FORMAT_LENGTH

### **Priority**

설명:	메시지 우선순위
PCF 매개변수:	MQIACF_PRIORITY
추적 레벨:	2
유형:	MQCFIN

### **Persistence**

설명:	메시지 지속성
PCF 매개변수:	MQIACF_PERSISTENCE
추적 레벨:	2
유형:	MQCFIN

### **MsgId**

설명:	메시지 ID
PCF 매개변수:	MQBACF_MSG_ID
추적 레벨:	2
유형:	MQCFBS
길이:	MQ_MSG_ID_LENGTH

### **CorrelId**

PCF 매개변수:	상관 ID
설명:	MQBACF_CORREL_ID
추적 레벨:	2
유형:	MQCFBS
길이:	MQ_CORREL_ID_LENGTH

### **ReplyToQueue**

설명:	
PCF 매개변수:	MQCACF_REPLY_TO_Q
추적 레벨:	2
유형:	MQCFST
길이:	MQ_Q_NAME_LENGTH

### **ReplyToQMgr**

설명:	
PCF 매개변수:	MQCACF_REPLY_TO_Q_MGR
추적 레벨:	2
유형:	MQCFST
길이:	MQCFST

### ***CodedCharSetId***

설명: 메시지 데이터의 문자 세트 ID  
PCF 매개변수: MQIA\_CODED\_CHAR\_SET\_ID  
추적 레벨: 2  
유형: MQCFIN

### ***Encoding***

설명: 메시지 데이터의 숫자 인코딩입니다.  
PCF 매개변수: MQIACF\_ENCODING  
추적 레벨: 2  
유형: MQCFIN

### ***PutDate***

설명:  
PCF 매개변수: MQCACF\_PUT\_DATE  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_PUT\_DATE\_LENGTH

### ***PutTime***

설명:  
PCF 매개변수: MQCACF\_PUT\_TIME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_PUT\_TIME\_LENGTH

### ***PolicyName***

설명: 이 메시지에 적용된 정책 이름.  
**참고:** AMS 보호 메시지만 해당함  
PCF 매개변수: MQCA\_POLICY\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_OBJECT\_NAME\_LENGTH

### ***XmitqMsgId***

설명: 전송 큐 헤더에 있는 메시지의 메시지 ID.  
**참고:** 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함  
PCF 매개변수: MQBACF\_XQH\_MSG\_ID  
추적 레벨: 2  
유형: MQCFBS

길이: MQ\_MSG\_ID\_LENGTH

### ***XmitqCorrelId***

설명: 전송 큐 헤더에 있는 메시지의 상관 ID.

참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQBACF\_XQH\_CORREL\_ID

추적 레벨: 2

유형: MQCFBS

길이: MQ\_CORREL\_ID\_LENGTH

### ***XmitqPutTime***

설명: 전송 큐 헤더에 있는 메시지의 넣기 시간.

참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_PUT\_TIME

추적 레벨: 2

유형: MQCFST

길이: MQ\_PUT\_TIME\_LENGTH

### ***XmitqPutDate***

설명: 전송 큐 헤더에 있는 메시지의 넣기 날짜.

참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_PUT\_DATE

추적 레벨: 2

유형: MQCFST

길이: MQ\_PUT\_DATE\_LENGTH

### ***XmitqRemoteQName***

설명: 전송 큐 헤더에 있는 메시지의 리모트 큐 목적지.

참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_REMOTE\_Q\_NAME

추적 레벨: 2

유형: MQCFST

길이: MQ\_Q\_NAME\_LENGTH

### ***XmitqRemoteQMgr***

설명: 전송 큐 헤더에 있는 메시지의 리모트 큐 관리자 목적지.

참고: 형식이 MQFMT\_XMIT\_Q\_HEADER일 때에만 해당함

PCF 매개변수: MQCACF\_XQH\_REMOTE\_Q\_MGR

추적 레벨: 2

유형: MQCFST

길이: MQ\_Q\_NAME\_LENGTH

### **PutMsgOptsStructure**

설명: MQPMO 구조.  
PCF 매개변수: MQBACF\_MQPMO\_STRUCT  
추적 레벨: 3  
유형: MQCFBS  
길이: MQPMO 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

### **MQPUT1 AppActivityDistList PCF 그룹 헤더 구조**

MQPUT1 함수가 분배 목록에 넣기를 수행 중이면 변수 매개변수가 AppActivityDistList PCF 그룹을 하나 포함합니다. 분배 목록의 각 큐에 대해서는 219 페이지의 『애플리케이션 활동 분배 목록 PCF 그룹 헤더 구조』의 내용을 참조하십시오. AppActivityDistList PCF 그룹은 MQOR, MQPMR, MQRR 구조의 정보를 결합하여 PUT 매개변수를 식별하고 각 큐에 대한 PUT 조작의 결과를 표시합니다. MQPUT1 조작의 경우 AppActivityDistList 그룹은 다음의 모든 또는 일부 매개변수를 포함합니다(이유 코드가 MQRC\_MULTIPLE\_REASONS이고 MQPMO.PutMsgRecFields 필드를 통해 다른 매개변수가 판별되면 CompCode, Reason, ObjectName, ObjectQMgrName 필드가 존재함).

#### **CompCode**

설명: 이 오브젝트에 대한 넣기 결과를 나타내는 완료 코드. MQRR 구조가 제공되고 MQPUT1의 이유 코드가 MQRC\_MULTIPLE\_REASONS인 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_COMP\_CODE  
추적 레벨: 2  
유형: MQCFIN

#### **Reason**

설명: 이 오브젝트에 대한 넣기 결과를 나타내는 이유 코드. MQRR 구조가 제공되고 MQPUT1의 이유 코드가 MQRC\_MULTIPLE\_REASONS인 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_REASON\_CODE  
추적 레벨: 2  
유형: MQCFIN

#### **ObjectName**

설명: 분배 목록의 큐 이름. MQOR 구조가 제공된 경우에만 포함됩니다.  
PCF 매개변수: MQCACF\_OBJECT\_NAME  
추적 레벨: 2  
유형: MQCFST  
길이: MQ\_Q\_NAME\_LENGTH

#### **MsgId**

설명: 메시지 ID. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_MSG\_ID가 있는 경우에만 포함됩니다.

---

<sup>4</sup> ResolvedLocalQName 매개변수는 ResolvedQName 매개변수와 다른 경우에만 포함됩니다.

PCF 매개변수: MQBACF\_MSG\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_MSG\_ID\_LENGTH

### **CorrelId**

설명: 상관 ID. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_CORREL\_ID가 있는 경우에만 포함됩니다.  
PCF 매개변수: MQBACF\_CORREL\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_CORREL\_ID\_LENGTH

### **GroupId**

설명: 그룹 ID. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_GROUP\_ID가 있는 경우에만 포함됩니다.  
PCF 매개변수: MQBACF\_GROUP\_ID  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_GROUP\_ID\_LENGTH

### **Feedback**

설명: 피드백. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_FEEDBACK이 있는 경우에만 포함됩니다.  
PCF 매개변수: MQIACF\_FEEDBACK  
추적 레벨: 2  
유형: MQCFIN

### **AccountingToken**

설명: AccountingToken. MQPMR 구조가 제공되고 PutMsgRecFields에 MQPMRF\_ACCOUNTING\_TOKEN이 있는 경우에만 포함됩니다.  
PCF 매개변수: MQBACF\_ACCOUNTING\_TOKEN  
추적 레벨: 2  
유형: MQCFBS  
길이: MQ\_ACCOUNTING\_TOKEN\_LENGTH.

### **MQSET**

애플리케이션이 MQSET MQI 함수를 시작함

### **ObjectHandle**

설명: 오브젝트 핸들  
PCF 매개변수: MQIACF\_HOBJ  
추적 레벨: 1

유형: MQCFIN

### **CompCode**

설명: 조작 결과를 나타내는 완료 코드

PCF 매개변수: MQIACF\_COMP\_CODE

추적 레벨: 1

유형: MQCFIN

### **Reason**

설명: 조작의 이유 코드 결과

PCF 매개변수: MQIACF\_REASON\_CODE

추적 레벨: 1

유형: MQCFIN

### **SelectorCount**

설명: Selectors 배열에 제공되는 선택자 수.

PCF 매개변수: MQIACF\_SELECTOR\_COUNT

추적 레벨: 2

유형: MQCFIN

### **Selectors**

설명: MQSET을 통해 값을 업데이트하고 있는 속성의 목록(정수 또는 문자).

PCF 매개변수: MQIACF\_SELECTORS

추적 레벨: 2

유형: MQCFIL

### **ResolvedQName**

설명: ResolvedType이 MQOT\_Q일 때 ObjectHandle에 참조된 큐 이름.

PCF 매개변수: MQCACF\_RESOLVED\_LOCAL\_Q\_NAME

추적 레벨: 2

유형: MQCFST

길이: MQ\_Q\_NAME\_LENGTH.

### **ResObjectString**

설명: ResolvedType이 MQOT\_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.

PCF 매개변수: MQCACF\_RESOLVED\_OBJECT\_STRING

추적 레벨: 2

유형: MQCFST

길이: 길이가 다양합니다.

### **ResolvedType**

설명:	ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT_Q, MQOT_TOPIC 또는 MQOT_NONE입니다.
PCF 매개변수:	MQIACF_RESOLVED_TYPE
추적 레벨:	2
유형:	MQCFIN

### **IntAttrCount**

설명:	설정 조작으로 업데이트할 정수 속성의 수.
PCF 매개변수:	MQIACF_INTATTR_COUNT
추적 레벨:	3
유형:	MQCFIN

### **IntAttrs**

설명:	정수 속성 값
PCF 매개변수:	MQIACF_INT_ATTRS
추적 레벨:	3
유형:	MQCFIL
범위:	이 매개변수는 IntAttrCount가 0보다 큰 경우에만 제공됩니다.

### **CharAttrs**

설명:	설정 조작으로 업데이트할 문자 속성. 값이 서로 연결됩니다.
PCF 매개변수:	MQCACF_CHAR_ATTRS
추적 레벨:	3
유형:	MQCFST
범위:	이 매개변수는 CharAttrLength가 0보다 큰 경우에만 포함됩니다.

### **MQSUB**

애플리케이션이 MQSUB MQI 함수를 시작함

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

### **SubHandle**

설명:	구독 핸들
PCF 매개변수:	MQIACF_HSUB
추적 레벨:	1
유형:	MQCFIN

### **ObjectHandle**

설명:	오브젝트 핸들
PCF 매개변수:	MQIACF_HOBJ
추적 레벨:	1
유형:	MQCFIN

### **Options**

설명:	구독 옵션
PCF 매개변수:	MQIACF_SUB_OPTIONS
추적 레벨:	1
유형:	MQCFIN

### **ObjectName**

설명:	오브젝트의 이름.
PCF 매개변수:	MQCACF_OBJECT_NAME
추적 레벨:	1
유형:	MQCFST
길이:	MQ_Q_NAME_LENGTH

### **ObjectString**

설명:	긴 오브젝트 이름.
PCF 매개변수:	MQCACF_OBJECT_STRING
추적 레벨:	1
유형:	MQCFST
범위:	MQSD.ObjectString의 VSLength 필드가 MQVS_NULL_TERMINATED 또는 0보다 큰 경우에만 포함됩니다.
길이:	길이 다양합니다.

### **AlternateUserId**

설명:	
PCF 매개변수:	MQCACF_ALTERNATE_USERID
추적 레벨:	2
유형:	MQCFST
범위:	MQSO_ALTERNATE_USER_AUTHORITY가 지정된 경우에만 포함됩니다.
길이:	MQ_USER_ID_LENGTH

### **AlternateSecurityId**

설명:	대체 보안 ID.
PCF 매개변수:	MQBACF_ALTERNATE_SECURITYID
추적 레벨:	2
유형:	MQCFBS
범위:	MQSO_ALTERNATE_USER_AUTHORITY가 지정되고 MQSD.AlternateSecurityId가 MQSID_NONE이 아닌 경우에만 존재합니다.
길이:	MQ_SECURITY_ID_LENGTH

### **SubName**

설명:	구독 이름
PCF 매개변수:	MQCACF_SUB_NAME
추적 레벨:	2
유형:	MQCFST
범위:	MQSD.SubName의 VSLength 필드가 MQVS_NULL_TERMINATED 또는 0보다 큰 경우에만 포함됩니다.
길이:	길이 다양합니다.

### **SubUserData**

설명:	Subscription 사용자 데이터
PCF 매개변수:	MQCACF_SUB_USER_DATA
추적 레벨:	2
유형:	MQCFST
범위:	MQSD.SubName의 VSLength 필드가 MQVS_NULL_TERMINATED 또는 0보다 큰 경우에만 포함됩니다.
길이:	길이 다양합니다.

### **SubCorrelId**

설명:	구독 상관 ID
PCF 매개변수:	MQBACF_SUB_CORREL_ID
추적 레벨:	2
유형:	MQCFBS
길이:	MQ_CORREL_ID_LENGTH

### **SelectionString**

설명:	선택 문자열.
PCF 매개변수:	MQCACF_SELECTION_STRING
추적 레벨:	2
유형:	MQCFST
범위:	MQSD의 VSLength 필드가 지정된 경우에만 포함됩니다. SelectionString은 MQVS_NULL_TERMINATED이거나 0보다 큼니다.
길이:	길이 다양합니다.

### **ResolvedQName**

설명:	ResolvedType이 MQOT_Q일 때 ObjectHandle에 참조된 큐 이름.
PCF 매개변수:	MQCACF_RESOLVED_LOCAL_Q_NAME
추적 레벨:	2
유형	MQCFST
길이:	MQ_Q_NAME_LENGTH.

### **ResObjectString**

설명:	ResolvedType이 MQOT_TOPIC일 때 ObjectHandle에 참조된 오브젝트 이름.
PCF 매개변수:	MQCACF_RESOLVED_OBJECT_STRING
추적 레벨:	2
유형	MQCFST
길이:	길이가 다양합니다.

### **ResolvedType**

설명:	ObjectHandle에 참조된 오브젝트의 유형. 가능한 값은 MQOT_Q, MQOT_TOPIC 또는 MQOT_NONE입니다.
PCF 매개변수:	MQIACF_RESOLVED_TYPE
추적 레벨:	2
유형	MQCFIN

### **SubDescriptorStructure**

설명:	MQSD 구조.
PCF 매개변수:	MQBACF_MQSD_STRUCT
추적 레벨:	3
유형:	MQCFBS
길이:	MQSD 구조의 길이(바이트).

### **MQSUBRQ**

애플리케이션이 MQSUBRQ MQI 함수를 시작함

#### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

#### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

### **SubHandle**

설명:	구독 핸들
PCF 매개변수:	MQIACF_HSUB
추적 레벨:	1
유형:	MQCFIN

### **SubOptions**

설명:	MQSB.Options의 하위 옵션
PCF 매개변수:	MQIACF_SUBRQ_OPTIONS
추적 레벨:	2
유형:	MQCFIN

### **Action**

설명:	구독 요청 조치(MQSR_*)
PCF 매개변수:	MQIACF_SUBRQ_ACTION
추적 레벨:	2
유형:	MQCFIN

### **NumPubs**

설명:	이 호출의 결과로 송신된(MQSB.NumPubs에서) 발행물 수
PCF 매개변수:	MQIACF_NUM_PUBS
추적 레벨:	2
유형:	MQCFIN

### **MQSTAT**

애플리케이션이 MQSTAT MQI 함수를 시작함

### **CompCode**

설명:	조작 결과를 나타내는 완료 코드
PCF 매개변수:	MQIACF_COMP_CODE
추적 레벨:	1
유형:	MQCFIN

### **Reason**

설명:	조작의 이유 코드 결과
PCF 매개변수:	MQIACF_REASON_CODE
추적 레벨:	1
유형:	MQCFIN

### **Type**

설명:	요청 중인 상태 정보의 유형
PCF 매개변수:	MQIACF_STATUS_TYPE

추적 레벨: 2  
유형: MQCFIN

### **StatusStructure**

설명: MQSTS 구조.  
PCF 매개변수: MQBACF\_MQSTS\_STRUCT  
추적 레벨: 3  
유형: MQCFBS  
길이: MQSTS 구조의 바이트 길이(실제 크기는 구조 버전에 따라 다름)

## **애플리케이션 활동 XA 조작에 대한 변수 매개변수**

XA 조작은 MQ가 트랜잭션에 참여하도록 하기 위해 애플리케이션이 수행할 수 있는 API 호출입니다. 각 조작에 대한 매개변수는 다음 절에 정의되어 있습니다.

추적 레벨은 추적에 포함시킬 매개변수에 필요한 추적 세분화 레벨을 표시합니다. 가능한 추적 레벨 값은 다음과 같습니다.

### 1. 낮음

애플리케이션에 "낮음", "중간" 또는 "높음" 활동 추적이 구성될 때 매개변수가 포함됩니다. 이 설정은 매개변수가 항상 조작에 대한 AppActivityData 그룹에 포함됨을 의미합니다. 애플리케이션이 수행하는 MQI 호출을 추적하고 호출이 성공했는지 여부를 확인하는 데에는 이 매개변수 세트면 충분합니다.

### 2. 보통

애플리케이션에 대해 "중간" 또는 "높음" 활동 추적이 구성되어 있을 때에만 조작의 AppActivityData 그룹에 매개변수가 포함됩니다. 이 매개변수 세트는 애플리케이션에 사용되는 큐와 토픽 이름 같은 자원에 대한 정보를 추가합니다.

### 3. 높음

애플리케이션에 대해 "높음" 활동 추적이 구성되어 있을 때에만 조작의 AppActivityData 그룹에 매개변수가 포함됩니다. 이 매개변수 세트는 MQI 및 XA 함수에 전달되는 구조의 메모리 덤프를 포함합니다. 이러한 이유로, MQI 및 XA 호출에 사용되는 매개변수에 대한 자세한 정보를 포함합니다. 구조 메모리 덤프는 구조의 셸로우(shallow) 사본입니다. 참조 해제 포인터에 대한 잘못된 시도를 피하기 위해 구조의 포인터 값은 NULL로 설정됩니다.

**참고:** 덤프된 구조 버전이 애플리케이션에 사용된 버전과 반드시 동일하지는 않습니다. API 교차 엑시트, 활동 추적 코드 또는 큐 관리자를 통해 구조를 수정할 수 있습니다. 큐 관리자는 이후 버전에 구조를 수정할 수 있지만 구조의 이전 버전에는 결코 구조를 변경하지 않습니다. 이전 버전에 대해 변경할 경우 데이터 손실 위험이 있습니다.

## **AXREG**

애플리케이션이 AXREG AX 함수를 시작함

### **XID**

설명: XID 구조  
PCF 매개변수: MQBACF\_XA\_XID  
추적 레벨: 1  
유형: MQCFBS  
길이: Sizeof(XID)

### **Rmid**

설명: 자원 관리자 ID

PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### **Flags**

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS  
추적 레벨: 1  
유형: MQCFIN

### **XARetCode**

설명: 리턴 코드  
PCF 매개변수: MQIACF\_XA\_RETCODE  
추적 레벨: 1  
유형: MQCFIN

### **AXUNREG**

애플리케이션이 AXUNREG AX 함수를 시작함

#### **Rmid**

설명: 자원 관리자 ID  
PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### **Flags**

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS  
추적 레벨: 1  
유형: MQCFIN

### **XARetCode**

설명: 리턴 코드  
PCF 매개변수: MQIACF\_XA\_RETCODE  
추적 레벨: 1  
유형: MQCFIN

### **XACLOSE**

애플리케이션이 XACLOSE AX 함수를 시작함

#### **Xa\_info**

설명: 자원 관리자를 초기화하는 데 사용된 정보.  
PCF 매개변수: MQCACF\_XA\_INFO  
추적 레벨: 1

유형: MQCFST

### **Rmid**

설명: 자원 관리자 ID  
PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### **Flags**

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS  
추적 레벨: 1  
유형: MQCFIN

### **XARetCode**

설명: 리턴 코드  
PCF 매개변수: MQIACF\_XA\_RETCODE  
추적 레벨: 1  
유형: MQCFIN

### **XACOMMIT**

애플리케이션이 XACOMMIT AX 함수를 시작함

### **XID**

설명: XID 구조  
PCF 매개변수: MQBACF\_XA\_XID  
추적 레벨: 1  
유형: MQCFBS  
길이: Sizeof(XID)

### **Rmid**

설명: 자원 관리자 ID  
PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### **Flags**

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS  
추적 레벨: 1  
유형: MQCFIN

### ***XARetCode***

설명:	리턴 코드
PCF 매개변수:	MQIACF_XA_RETCODE
추적 레벨:	1
유형:	MQCFIN

### ***XACOMPLETE***

애플리케이션이 XACOMPLETE AX 함수를 시작함

#### ***Handle***

설명:	비동기 조작에 대한 핸들
PCF 매개변수:	MQIACF_XA_HANDLE
추적 레벨:	1
유형:	MQCFIN

#### ***Retval***

설명:	비동기 함수의 리턴 값
PCF 매개변수:	MQIACF_XA_RETVAL
추적 레벨:	1
유형:	MQCFINMQCFBS

#### ***Rmid***

설명:	자원 관리자 ID
PCF 매개변수:	MQIACF_XA_RMID
추적 레벨:	1
유형:	MQCFIN

#### ***Flags***

설명:	플래그
PCF 매개변수:	MQIACF_XA_FLAGS
추적 레벨:	1
유형:	MQCFIN

### ***XARetCode***

설명:	리턴 코드
PCF 매개변수:	MQIACF_XA_RETCODE
추적 레벨:	1
유형:	MQCFIN

### ***XAEND***

애플리케이션이 XAEND AX 함수를 시작함

#### ***XID***

설명:	XID 구조
-----	--------

PCF 매개변수: MQBACF\_XA\_XID  
추적 레벨: 1  
유형: MQCFBS  
길이: Sizeof(XID)

### **Rmid**

설명: 자원 관리자 ID  
PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### **Flags**

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS  
추적 레벨: 1  
유형: MQCFIN

### **XARetCode**

설명: 리턴 코드  
PCF 매개변수: MQIACF\_XA\_RETCODE  
추적 레벨: 1  
유형: MQCFIN

### **XAFORGET**

애플리케이션이 AXREG AX 함수를 시작함

### **XID**

설명: XID 구조  
PCF 매개변수: MQBACF\_XA\_XID  
추적 레벨: 1  
유형: MQCFBS  
길이: Sizeof(XID)

### **Rmid**

설명: 자원 관리자 ID  
PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### **Flags**

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS

추적 레벨: 1  
유형: MQCFIN

### ***XARetCode***

설명: 리턴 코드  
PCF 매개변수: MQIACF\_XA\_RETCODE  
추적 레벨: 1  
유형: MQCFIN

### ***XAOPEN***

애플리케이션이 XAOPEN AX 함수를 시작함

### ***Xa\_info***

설명: 자원 관리자를 초기화하는 데 사용된 정보.  
PCF 매개변수: MQCACF\_XA\_INFO  
추적 레벨: 1  
유형: MQCFST

### ***Rmid***

설명: 자원 관리자 ID  
PCF 매개변수: MQIACF\_XA\_RMID  
추적 레벨: 1  
유형: MQCFIN

### ***Flags***

설명: 플래그  
PCF 매개변수: MQIACF\_XA\_FLAGS  
추적 레벨: 1  
유형: MQCFIN

### ***XARetCode***

설명: 리턴 코드  
PCF 매개변수: MQIACF\_XA\_RETCODE  
추적 레벨: 1  
유형: MQCFIN

### ***XAPREPARE***

애플리케이션이 XAPREPARE AX 함수를 시작함

### ***XID***

설명: XID 구조  
PCF 매개변수: MQBACF\_XA\_XID  
추적 레벨: 1  
유형: MQCFBS

길이:                    Sizeof(XID)

### **Rmid**

설명:                    자원 관리자 ID  
PCF 매개변수:           MQIACF\_XA\_RMID  
추적 레벨:               1  
유형:                    MQCFIN

### **Flags**

설명:                    플래그  
PCF 매개변수:           MQIACF\_XA\_FLAGS  
추적 레벨:               1  
유형:                    MQCFIN

### **XARetCode**

설명:                    리턴 코드  
PCF 매개변수:           MQIACF\_XA\_RETCODE  
추적 레벨:               1  
유형:                    MQCFIN

### **XARECOVER**

애플리케이션이 XARECOVER AX 함수를 시작함

### **Count**

설명:                    XID 수  
PCF 매개변수:           MQIACF\_XA\_COUNT  
추적 레벨:               1  
유형:                    MQCFIN

### **XIDs**

설명:                    XID 구조  
**참고:** 모든 XID 구조에 하나씩, 최대 Count XID 수까지 이 PCF 매개변수의 여러 인스턴스가 있습니다.  
PCF 매개변수:           MQBACF\_XA\_XID  
추적 레벨:               1  
유형:                    MQCFBS  
길이:                    Sizeof(XID)

### **Rmid**

설명:                    자원 관리자 ID  
PCF 매개변수:           MQIACF\_XA\_RMID  
추적 레벨:               1  
유형:                    MQCFIN

### **Flags**

설명:	플래그
PCF 매개변수:	MQIACF_XA_FLAGS
추적 레벨:	1
유형:	MQCFIN

### **XARetCode**

설명:	리턴 코드
PCF 매개변수:	MQIACF_XA_RETCODE
추적 레벨:	1
유형:	MQCFIN

### **XAROLLBACK**

애플리케이션이 XAROLLBACK AX 함수를 시작함

### **XID**

설명:	XID 구조
PCF 매개변수:	MQBACF_XA_XID
추적 레벨:	1
유형:	MQCFBS
길이:	Sizeof(XID)

### **Rmid**

설명:	자원 관리자 ID
PCF 매개변수:	MQIACF_XA_RMID
추적 레벨:	1
유형:	MQCFIN

### **Flags**

설명:	플래그
PCF 매개변수:	MQIACF_XA_FLAGS
추적 레벨:	1
유형:	MQCFIN

### **XARetCode**

설명:	리턴 코드
PCF 매개변수:	MQIACF_XA_RETCODE
추적 레벨:	1
유형:	MQCFIN

### **XASTART**

애플리케이션이 XASTART AX 함수를 시작함

## **XID**

설명:	XID 구조
PCF 매개변수:	MQBACF_XA_XID
추적 레벨:	1
유형:	MQCFBS
길이:	Sizeof(XID)

## **Rmid**

설명:	자원 관리자 ID
PCF 매개변수:	MQIACF_XA_RMID
추적 레벨:	1
유형:	MQCFIN

## **Flags**

설명:	플래그
PCF 매개변수:	MQIACF_XA_FLAGS
추적 레벨:	1
유형:	MQCFIN

## **XARetCode**

설명:	리턴 코드
PCF 매개변수:	MQIACF_XA_RETCODE
추적 레벨:	1
유형:	MQCFIN

## **실시간 모니터링**

---

실시간 모니터링은 큐 관리자 내 채널과 큐의 현재 상태를 판별할 수 있게 하는 기술입니다. 명령이 발행될 때 리턴되는 정보는 정확합니다.

큐와 채널에 대한 리턴 실시간 정보가 발행될 때 여러 명령이 사용 가능합니다. 하나 이상의 큐 또는 채널에 대한 정보가 리턴될 수 있으며 정보의 양은 차이가 있을 수 있습니다. 다음 태스크에서 실시간 모니터링을 사용할 수 있습니다.

- 시스템 관리자가 IBM WebSphere MQ 시스템의 정상 상태를 이해하도록 돕습니다. 시스템에서 문제가 발생할 경우 이는 문제점 진단에 유용합니다.
- 특정 이벤트나 문제점이 감지되지 않았어도 어느 시점에나 큐 관리자의 조건 판별.
- 시스템의 문제점 원인 판별 지원.

실시간 모니터링으로 큐나 채널에 대한 정보가 리턴될 수 있습니다. 리턴되는 실시간 정보의 양은 큐 관리자, 큐, 채널 속성으로 제어합니다.

- 명령을 발행하여 큐를 모니터해서 큐가 제대로 서비스하는지 확인합니다. 일부 큐 속성을 사용하려면 실시간 모니터링에 해당 속성을 사용 가능하게 해야 합니다.
- 명령을 발행하여 채널을 모니터해서 채널이 제대로 실행 중인지 확인합니다. 일부 채널 속성을 사용하려면 실시간 모니터링에 해당 속성을 사용 가능하게 해야 합니다.

큐 및 채널에 대한 실시간 모니터링은 성능 및 채널 이벤트 모니터링 외의 것으로 이와는 별도입니다.

## 실시간 모니터링을 제어하는 속성

실시간 모니터링이 사용되면 일부 큐 및 채널 상태 속성은 모니터링 정보를 보유합니다. 실시간 모니터링이 사용되지 않는 경우에는 이 모니터링 속성에 모니터링 정보가 보유되지 않습니다. 아래의 예는 이러한 큐 및 채널 상태 속성의 사용 방법을 보여줍니다.

개별 큐 또는 채널이나 여러 큐 또는 채널에 대해서 실시간 모니터링을 사용 가능 또는 불가능하게 할 수 있습니다. 개별 큐나 채널을 제어하려면 큐 속성 MONQ 또는 채널 속성 MONCHL을 설정하여 실시간 모니터링을 사용 가능 또는 불가능하게 하십시오. 여러 큐나 채널을 함께 제어하려면 큐 관리자 속성 MONQ 및 MONCHL을 사용하여 큐 관리자 레벨에서 실시간 모니터링을 사용 가능 또는 불가능하게 하십시오. 기본값이 지정된 모니터링 속성이 있는 모든 큐와 채널 오브젝트의 경우 큐 관리자 레벨에서 실시간 모니터링이 제어됩니다.

자동으로 정의된 클러스터 송신자 채널이 WebSphere MQ 오브젝트가 아니므로 채널 오브젝트와 동일한 방법으로 속성을 갖지 마십시오. 자동으로 정의된 클러스터 송신자 채널을 제어하려면 큐 관리자 속성, MONACLS를 사용하십시오. 이 속성은 큐 관리자 내의 자동으로 정의된 클러스터 송신자 채널이 채널 모니터링에 대해 사용 가능 또는 불가능한지 여부를 판별합니다.

채널 실시간 모니터링의 경우 MONCHL 속성을 세 개의 모니터링 레벨 낮음, 중간 또는 높음 중 하나로 설정할 수 있습니다. 오브젝트 레벨 또는 큐 관리자 레벨에서 모니터링 레벨을 설정할 수 있습니다. 레벨의 선택은 시스템에 따라 다릅니다. 모니터링 데이터를 수집하려면 시스템 시간 확보와 같이 계산 비용이 상대적으로 값비싼 몇 가지 명령어가 필요할 수 있습니다. 실시간 모니터링의 영향을 줄이기 위해 중간 및 낮음 모니터링 옵션은 데이터를 항상 수집하기 보다는 데이터 샘플을 일정한 간격으로 측정합니다. [253 페이지의 표 29](#)에는 채널의 실시간 모니터링에 사용 가능한 모니터링 레벨이 요약되어 있습니다.

표 29. 모니터링 레벨		
레벨	설명	사용법
낮음	일정한 간격으로 작은 데이터 샘플을 측정합니다.	높은 볼륨의 메시지를 처리하는 오브젝트의 경우.
보통	일정한 간격으로 데이터 샘플을 측정합니다.	대부분의 오브젝트의 경우.
높음	일정한 간격으로 모든 데이터를 측정합니다.	초당 몇 개의 메시지만을 처리하는 오브젝트의 경우(최신 정보가 중요함).

큐 실시간 모니터링의 경우에는 MONQ 속성을 세 개의 모니터링 레벨 낮음, 중간 또는 높음 중 하나로 설정할 수 있습니다. 하지만 이들 값의 차이가 뚜렷하지 않습니다. 모든 값이 데이터 콜렉션을 사용하지만 샘플 크기에 영향을 미치지 않습니다.

### 예:

다음 예는 모니터링 레벨을 제어하기 위해 필요한 큐, 채널, 큐 관리자 속성의 설정 방법을 보여줍니다. 모든 예에서, 모니터링이 사용될 때 큐와 채널 오브젝트의 모니터링 레벨은 중간입니다.

1. 큐 관리자 레벨에서 모든 큐와 채널에 대해 큐 및 채널 모니터링을 모두 사용하려면 다음 명령을 사용하십시오.

```
ALTER QMGR MONQ(MEDIUM) MONCHL(MEDIUM)
ALTER QL(Q1) MONQ(QMGR)
ALTER CHL(QM1.TO.QM2) CHLTYPE(SDR) MONCHL(QMGR)
```

2. 로컬 큐, Q1 및 송신자 채널, QM1.TO.QM2를 제외하고 모든 큐와 채널에 모니터링을 사용하려면 다음 명령을 사용하십시오.

```
ALTER QMGR MONQ(MEDIUM) MONCHL(MEDIUM)
ALTER QL(Q1) MONQ(OFF)
ALTER CHL(QM1.TO.QM2) CHLTYPE(SDR) MONCHL(OFF)
```

- 로컬 큐, Q1 및 송신자 채널, QM1.TO.QM2를 제외하고 모든 큐와 채널에 대해 큐 및 채널 모니터링을 모두 사용하지 않으려면 다음 명령을 사용하십시오.

```
ALTER QMGR MONQ(OFF) MONCHL(OFF)
ALTER QL(Q1) MONQ(MEDIUM)
ALTER CHL(QM1.TO.QM2) CHLTYPE(SDR) MONCHL(MEDIUM)
```

- 개별 오브젝트 속성과 무관하게, 모든 큐와 채널에 대해 큐 및 채널 모니터링을 모두 사용하지 않으려면 다음 명령을 사용하십시오.

```
ALTER QMGR MONQ(NONE) MONCHL(NONE)
```

- 자동으로 정의된 클러스터 송신자 채널의 모니터링 기능을 제어하려면 다음 명령을 사용하십시오.

```
ALTER QMGR MONACLS(MEDIUM)
```

- 자동으로 정의된 클러스터 송신자 채널에 채널 모니터링을 위한 큐 관리자 설정을 사용하도록 지정하려면 다음 명령을 사용하십시오.

```
ALTER QMGR MONACLS(QMGR)
```

## 관련 개념

### 252 페이지의 『실시간 모니터링』

실시간 모니터링은 큐 관리자 내 채널과 큐의 현재 상태를 판별할 수 있게 하는 기술입니다. 명령이 발행될 때 리턴되는 정보는 정확합니다.

### 큐 관리자에 대한 작업

#### 관련 태스크

### 254 페이지의 『큐와 채널 모니터링 데이터 표시』

큐나 채널에 대한 실시간 모니터링 정보를 표시하려면 IBM WebSphere MQ 탐색기 또는 해당 MQSC 명령을 사용하십시오. 일부 모니터링 필드는 큐 관리자의 조작을 모니터링하는 데 유용한 쉼표로 구분된 지표 값 쌍을 표시합니다. 아래의 예는 모니터링 데이터의 표시 방법을 보여줍니다.

### 모니터링(MONCHL)

## 큐와 채널 모니터링 데이터 표시

큐나 채널에 대한 실시간 모니터링 정보를 표시하려면 IBM WebSphere MQ 탐색기 또는 해당 MQSC 명령을 사용하십시오. 일부 모니터링 필드는 큐 관리자의 조작을 모니터링하는 데 유용한 쉼표로 구분된 지표 값 쌍을 표시합니다. 아래의 예는 모니터링 데이터의 표시 방법을 보여줍니다.

### 이 태스크 정보

쉼표로 구분된 값 쌍을 표시하는 모니터링 필드는 오브젝트에 대한 모니터링이 사용된 이후에 또는 큐 관리자가 시작되었을 때부터 측정된 시간의 단기 및 장기 지표를 제공합니다.

- 단기 지표는 쌍의 첫 번째 값이며 최신 측정에 더 높은 가중치를 주어서 이 값에 대한 영향을 더 크게 하는 방식으로 계산됩니다. 수행한 측정의 최근 추세를 표시합니다.
- 쌍에서 두 번째 값의 장기 지표는 최신 측정에 더 높은 가중치를 주지 않는 방식으로 계산됩니다. 자원의 성능에 대한 보다 장기적 활동을 표시합니다.

이 지표 값은 큐 관리자의 조작 변경사항을 감지하는 데 있어서 가장 유용합니다. 시간의 증가를 감지하려면 정상 사용일 때 지표가 표시하는 시간을 알아야 합니다. 이 값을 정기적으로 수집하고 확인해서 큐 관리자의 작동 변동을 감지할 수 있습니다. 이는 성능의 변화를 나타내는 것일 수 있습니다.

다음과 같이 실시간 모니터링 정보를 확보하십시오.

## 프로시저

1. 큐에 대한 실시간 모니터링 정보를 표시하려면 IBM WebSphere MQ 탐색기 또는 MQSC 명령 DISPLAY QSTATUS을 사용하여 선택적 매개변수 MONITOR를 지정하십시오.
2. 채널에 대한 실시간 모니터링 정보를 표시하려면 IBM WebSphere MQ 탐색기 또는 MQSC 명령 DISPLAY CHSTATUS을 사용하여 선택적 매개변수 MONITOR를 지정하십시오.

## 예

큐 Q1에는 기본값 QMGR로 설정된 속성 MONQ가 있고, 해당 큐를 소유한 큐 관리자에는 MEDIUM으로 설정된 속성 MONQ가 있습니다. 이 큐에 대해 수집된 모니터링 필드를 표시하려면 다음 명령을 사용하십시오.

```
DISPLAY QSTATUS(Q1) MONITOR
```

큐 Q1의 모니터링 필드 및 모니터링 레벨은 다음과 같이 표시됩니다.

```
QSTATUS(Q1)
TYPE(Queue)
MONQ(MEDIUM)
QTIME(11892157,24052785)
MSGAGE(37)
LPUTDATE(2005-03-02)
LPUTTIME(09.52.13)
LGETDATE(2005-03-02)
LGETTIME(09.51.02)
```

송신자 채널 QM1.TO.QM2에는 기본값 QMGR로 설정된 속성 MONCHL이 있고, 해당 큐를 소유한 큐 관리자에는 MEDIUM으로 설정된 속성 MONCHL이 있습니다. 이 채널에 대해 수집된 모니터링 필드를 표시하려면 다음 명령을 사용하십시오.

```
DISPLAY CHSTATUS(QM1.TO.QM2) MONITOR
```

송신자 채널 QM1.TO.QM2의 모니터링 필드 및 모니터링 레벨은 다음과 같이 표시됩니다.

```
CHSTATUS(QM1.TO.QM2)
XMITQ(Q1)
CONNNAME(127.0.0.1)
CURRENT
CHLTYPE(SDR)
STATUS(RUNNING)
SUBSTATE(MQGET)
MONCHL(MEDIUM)
XQTIME(755394737,755199260)
NETTIME(13372,13372)
EXITTIME(0,0)
XBATCHSZ(50,50)
COMPTIME(0,0)
STOPREQ(NO)
RQMNAME(QM2)
```

## 관련 개념

252 페이지의 『실시간 모니터링』

실시간 모니터링은 큐 관리자 내 채널과 큐의 현재 상태를 판별할 수 있게 하는 기술입니다. 명령이 발행될 때 리턴되는 정보는 정확합니다.

## 관련 참조

[DISPLAY QSTATUS](#)

## 큐 모니터링

이 페이지는 큐와 이 큐를 서비스하는 애플리케이션에 대한 문제점을 해결하는 데 도움을 줍니다. 다양한 모니터링 옵션을 사용하여 문제점을 판별할 수 있습니다.

서비스하는 큐에 대한 문제점의 첫 번째 신호는 종종 큐의 메시지 수(CURDEPTH) 증가입니다. 하루의 특정 시점이나 특정 워크로드 하에서 증가를 예상하는 경우 메시지 수 증가는 문제를 나타내는 것이 아닐 수 있습니다. 하지만 메시지 수 증가를 설명할 수 없으면 원인을 조사해야 할 수 있습니다.

애플리케이션에 대한 문제가 있는 애플리케이션 큐나 채널에 대한 문제가 있는 전송 큐가 있을 수 있습니다. 큐를 서비스하는 애플리케이션이 채널일 때 추가 모니터링 옵션이 사용 가능합니다.

다음 예는 Q1이라는 특정 큐에 대한 문제점을 조사하고 다양한 명령 출력이 표시되는 필드를 설명합니다.

## 애플리케이션이 큐를 여는지 판별

큐에 대한 문제가 있는 경우 애플리케이션이 큐를 여는지 여부를 확인하십시오.

### 이 태스크 정보

다음 단계를 수행하여 애플리케이션이 큐를 여는지 판별하십시오.

### 프로시저

1. 큐에 대해 실행 중인 애플리케이션이 예상한 애플리케이션인지 확인하십시오. 문제의 큐에 대해 다음 명령을 실행하십시오.

```
DISPLAY QSTATUS(Q1) TYPE(HANDLE) ALL
```

출력에서 APPLTAG 필드를 보고 애플리케이션의 이름이 표시되는지 확인하십시오. 애플리케이션의 이름이 표시되지 않은 경우나 출력이 전혀 없는 경우 애플리케이션을 시작하십시오.

2. 큐가 전송 큐이면 CHANNEL 필드에서 출력을 보십시오.  
채널 이름이 CHANNEL 필드에 표시되지 않은 경우에는 채널이 실행 중인지 판별하십시오.
3. 큐에 대해 실행 중인 애플리케이션이 입력을 위한 큐를 열었는지 확인하십시오. 다음 명령을 실행하십시오.

```
DISPLAY QSTATUS(Q1) TYPE(Queue) ALL
```

출력에서 IPPROCS 필드를 보고 애플리케이션이 입력을 위한 큐를 열었는지 확인하십시오. 값이 0이고 큐가 애플리케이션 큐이면 애플리케이션이 큐에서 메시지를 가져오기 위해 입력을 위한 큐를 여는지 확인하십시오.

## 큐의 메시지가 사용 가능한지 확인

큐에 수많은 메시지가 있고 애플리케이션이 이 메시지를 처리하지 않는 경우 큐의 메시지가 애플리케이션에 사용 가능한지 확인하십시오.

### 이 태스크 정보

다음 단계를 수행하여 애플리케이션이 큐의 메시지를 처리하지 않는 이유를 조사하십시오.

### 프로시저

1. 애플리케이션이 큐의 모든 메시지를 처리해야 할 때 특정 메시지 ID 또는 상관 ID를 요구하지 않는지 확인하십시오.
2. 큐의 현재 용량이 큐의 메시지 수가 증가하고 있음을 표시하지만, 큐의 일부 메시지는 커밋되지 않아서 애플리케이션이 가져오지 못할 수 있습니다(큐의 현재 용량은 큐에 대한 메시지의 커밋되지 않은 MQPUT 수를 포함함). 다음 명령을 실행하십시오.

```
DISPLAY QSTATUS(Q1) TYPE(Queue) ALL
```

출력에서 UNCOM 필드를 보고 큐에 커밋되지 않은 메시지가 있는지 확인하십시오.

3. 애플리케이션이 큐에서 메시지를 가져오려 시도 중인 경우에는 넣고 있는 애플리케이션이 메시지를 제대로 커밋하는지 확인하십시오. 다음 명령을 발행하여 이 큐에 메시지를 넣고 있는 애플리케이션의 이름을 찾으십시오.

```
DISPLAY QSTATUS(Q1) TYPE(HANDLE) OPENTYPE(OUTPUT)
```

4. 그리고 나서 이전 명령 출력의 APPLTAG 값을 <appltag>에 삽입하여 다음 명령을 실행하십시오.

```
DISPLAY CONN(*) WHERE(APPLTAG EQ <appltag>) UOWSTDA UOWSTTI
```

그러면 작업 단위가 언제 시작되었는지 표시되어 애플리케이션이 장기 실행 작업 단위를 작성 중인지 여부를 파악하는 데 도움이 됩니다. 넣고 있는 애플리케이션이 채널이면 배치를 완료하는 데 시간이 오래 걸리는 이유를 조사할 수 있습니다.

## 애플리케이션이 큐에서 메시지를 가져오는지 확인

큐와 이 큐를 서비스하는 애플리케이션에 문제가 있는 경우 애플리케이션이 큐에서 메시지를 가져오는지 여부를 확인하십시오.

### 이 태스크 정보

애플리케이션이 큐에서 메시지를 가져오는지 확인하려면 다음 검사를 수행하십시오.

### 프로시저

1. 큐에 대해 실행 중인 애플리케이션이 큐에서 실제로 메시지를 처리 중인지 확인하십시오. 다음 명령을 실행하십시오.

```
DISPLAY QSTATUS(Q1) TYPE(Queue) ALL
```

출력에서, 큐로부터 마지막 가져오기를 수행할 때 표시되는 LGETDATE 및 LGETTIME 필드를 보십시오.

2. 이 큐에서의 마지막 가져오기가 예상보다 오래 전이면 애플리케이션이 메시지를 제대로 처리하고 있는지 확인하십시오.

애플리케이션이 채널이면 메시지가 이 채널을 통해 이동하는지 확인하십시오.

## 애플리케이션이 메시지를 충분히 빨리 처리할 수 있는지 판별

메시지가 큐에 빌드되고 있지만 다른 검사에서 처리 문제점이 발견되지 않은 경우에는 애플리케이션이 메시지를 충분히 빨리 처리할 수 있는지 확인하십시오. 애플리케이션이 채널이면 채널이 메시지를 충분히 빠르게 처리할 수 있는지 확인하십시오.

### 이 태스크 정보

애플리케이션이 메시지를 빠르게 처리 중인지 판별하려면 다음 테스트를 수행하십시오.

### 프로시저

1. 다음 명령을 주기적으로 발행하여 큐에 대한 성능 데이터를 수집하십시오.

```
DISPLAY QSTATUS(Q1) TYPE(Queue) ALL
```

QTIME 표시기의 값이 높거나 일정 기간에 증가하고 있으며 큐의 메시지가 사용 가능한지 확인해서 작업 단위의 장기 실행 가능성을 이미 배제했으면, 가져오기 애플리케이션이 넣기 애플리케이션을 따라잡지 못하는 것일 수 있습니다.

2. 가져오기 애플리케이션이 넣기 애플리케이션을 따라잡지 못하는 경우 큐를 처리할 다른 가져오기 애플리케이션을 추가할 것을 고려하십시오.

다른 가져오기 애플리케이션을 추가할 수 있는지 여부는 애플리케이션의 설계와 둘 이상의 애플리케이션이 큐를 공유할 수 있는지 여부에 따라 다릅니다. 두 개의 애플리케이션이 한 큐를 동시에 처리할 수 있게 하려면 상관 ID별 메시지 가져오기나 그룹화와 같은 기능이 유용할 수 있습니다.

## 현재 용량이 증가하지 않을 때 큐 확인

큐의 현재 용량이 증가하지 않아도 큐를 모니터링하여 애플리케이션이 메시지를 제대로 처리 중인지 확인하는 데 여전히 유용할 수 있습니다.

### 이 태스크 정보

큐에 대한 성능 데이터를 수집하려면 다음 명령을 주기적으로 발행하십시오.

### 프로시저

다음 명령을 주기적으로 발행하십시오.

```
DISPLAY QSTATUS(Q1) TYPE(Queue) MSGAGE QTIME
```

출력에서, MSGAGE의 값이 일정 기간에 증가하며 애플리케이션이 모든 메시지를 처리하도록 설계된 경우 이는 일부 메시지가 전혀 처리되고 있지 않음을 나타내는 것일 수 있습니다.

## 채널 모니터링

이 페이지를 사용하여 전송 큐와 이 큐를 서비스하는 채널에 대한 문제점을 해결하는 데 도움이 되는 태스크를 봅니다. 다양한 채널 모니터링 옵션을 사용하여 문제점을 판별할 수 있습니다.

서비스하는 큐에 대한 문제점의 첫 번째 신호는 종종 큐의 메시지 수(CURDEPTH) 증가입니다. 하루의 특정 시점이나 특정 워크로드 하에서 증가를 예상하는 경우 메시지 수 증가는 문제를 나타내는 것이 아닐 수 있습니다. 하지만 메시지 수 증가를 설명할 수 없으면 원인을 조사해야 할 수 있습니다.

전송 큐를 서비스하는 채널에 문제가 있을 수 있습니다. 다양한 채널 모니터링 옵션을 사용하여 문제점을 판별할 수 있습니다.

다음 예는 전송 큐 QM2 및 채널 QM1.TO.QM2에 대한 문제점을 조사합니다. 이 채널은 큐 관리자 QM1에서 큐 관리자 QM2로 메시지를 송신하는 데 사용됩니다. 큐 관리자 QM1의 채널 정의는 송신자 또는 서버 채널이고 큐 관리자 QM2의 채널 정의는 수신자 또는 요청자 채널입니다.

### 채널이 실행 중인지 판별

전송 큐에 대한 문제가 있는 경우 채널이 실행 중인지 여부를 확인하십시오.

### 이 태스크 정보

다음 단계를 수행하여 전송 큐를 서비스하는 채널의 상태를 확인하십시오.

### 프로시저

1. 다음 명령을 발행하여 전송 큐 QM2를 처리할 것으로 예상하는 채널을 찾으십시오.

```
DIS CHANNEL(*) WHERE(XMITQ EQ QM2)
```

이 예에서, 이 명령의 출력은 전송 큐를 서비스하는 채널이 QM1.TO.QM2임을 보여줍니다.

2. 다음 명령을 발행하여 채널 QM1.TO.QM2의 상태를 판별하십시오.

```
DIS CHSTATUS(QM1.TO.QM2) ALL
```

3. **CHSTATUS** 명령의 출력에서 STATUS 필드를 조사하십시오.

- STATUS 필드의 값이 RUNNING이면 채널이 메시지를 이동시키고 있는지 확인하십시오.

- 명령의 출력에 상태가 표시되지 않거나 STATUS 필드의 값이 STOPPED, RETRY, BINDING 또는 REQUESTING인 경우 다음과 같이 적절한 단계를 수행하십시오.
4. 옵션: STATUS 필드의 값이 상태를 표시하지 않는 경우 채널이 비활성 상태이므로 다음 단계를 수행하십시오.
    - a) 트리거를 통해 채널을 자동으로 시작했으면 전송 큐의 메시지가 사용 가능한지 확인하십시오. 전송 큐에 사용 가능한 메시지가 있으면 전송 큐에 대한 트리거 설정이 올바른지 확인하십시오.
    - b) 다음 명령을 발행하여 채널을 다시 수동으로 시작하십시오.

```
START CHANNEL(QM1.TO.QM2)
```

5. 옵션: STATUS 필드의 값이 STOPPED인 경우 다음 단계를 수행하십시오.
  - a) 오류 로그를 확인하여 채널이 중지된 원인을 판별하십시오. 오류로 인해 채널이 중지된 경우 문제점을 정정하십시오.  
채널에 재시도 속성 *SHORTRTY* 및 *LONGRTY*의 값이 지정되어 있는지도 확인하십시오. 네트워크 오류와 같은 일시적 장애의 경우에는 채널이 자동으로 재시작을 시도합니다.
  - b) 다음 명령을 발행하여 채널을 다시 수동으로 시작하십시오.

```
START CHANNEL(QM1.TO.QM2)
```

6. 옵션: STATUS 필드의 값이 RETRY인 경우 다음 단계를 수행하십시오.
  - a) 오류 로그를 확인하여 오류를 식별한 후 문제점을 정정하십시오.
  - b) 다음 명령을 발행하여 채널을 다시 수동으로 시작하십시오.

```
START CHANNEL(QM1.TO.QM2)
```

또는 채널이 다음 번 재시도에 연결할 때까지 대기하십시오.

7. 옵션: STATUS 필드의 값이 BINDING 또는 REQUESTING이면 채널이 아직 파트너에 연결되지 않은 것입니다. 다음 단계를 수행하십시오.
  - a) 채널의 양쪽 끝에서 다음 명령을 발행하여 채널의 하위 상태를 판별하십시오.

```
DIS CHSTATUS(QM1.TO.QM2) ALL
```

**참고:**

- i) 일부 경우 채널의 한쪽 끝에만 하위 상태가 있을 수도 있습니다.
  - ii) 많은 하위 상태는 일시적이므로 명령을 몇 번 발행해서 채널이 특정 하위 상태로 유지되는지 판별하십시오.
- b) 259 페이지의 표 30를 확인하여 수행할 조치를 판별하십시오.

표 30. 바인딩 또는 요청 중 상태로 표시된 하위 상태		
시작 MCA 하위 상태 <sup>1</sup>	응답 MCA 하위 상태 <sup>2</sup>	참고사항
NAMESERVER		시작 MCA가 이름 서버 요청이 완료될 때까지 대기하고 있습니다. 채널 속성 CONNAME에 올바른 호스트 이름이 지정되었고 이름 서버가 제대로 설정되었는지 확인하십시오.
SCYEXIT	SCYEXIT	MCA가 현재 보안 엑시트를 통해 대화 중입니다. 추가 정보는 261 페이지의 『채널이 메시지를 충분히 빨리 처리할 수 있는지 판별』의 내용을 참조하십시오.
	CHADEXIT	채널 자동 정의 엑시트가 현재 실행 중입니다. 자세한 정보는 261 페이지의 『채널이 메시지를 충분히 빨리 처리할 수 있는지 판별』의 내용을 참조하십시오.

표 30. 바인딩 또는 요청 중 상태로 표시된 하위 상태 (계속)		
시작 MCA 하위 상태 <sup>1</sup>	응답 MCA 하위 상태 <sup>2</sup>	참고사항
RCVEXIT SENDEXIT MSGEXIT MREXIT	RCVEXIT SENDEXIT MSGEXIT MREXIT	MQXR_INIT에 대한 채널 시작 시 엑시트가 호출됩니다. 시간이 오래 걸리면 엑시트의 이 부분에서 처리를 검토하십시오. 자세한 정보는 261 페이지의 『채널이 메시지를 충분히 빨리 처리할 수 있는지 판별』의 내용을 참조하십시오.
SERIALIZE	SERIALIZE	이 하위 상태는 SHARED 속성 지정의 채널에만 적용됩니다.
NETCONNECT		올바르지 않은 네트워크 구성으로 인해 연결 지연이 있으면 이 하위 상태가 표시됩니다.
SSLHANDSHAKE	SSLHANDSHAKE	SSL 데이터 교환은 많은 송신 및 수신으로 이루어집니다. 네트워크 시간이 느리거나 검색 CRL 연결이 느린 경우 이는 데이터 교환을 수행하는 데 걸리는 시간에 영향을 미칩니다.

**참고사항:**

- i) 시작 MCA는 대화를 시작한 채널의 끝입니다. 송신자, 클러스터 송신자, 완전한 서버, 요청자일 수 있습니다. 서버-요청자 쌍에서는 채널을 시작한 끝입니다.
- ii) 응답 MCA는 대화 시작 요청에 응답한 채널의 끝입니다. 수신자, 클러스터 수신자, 요청자(서버 또는 송신자가 시작될 때), 서버(요청자가 시작될 때), 송신자(채널의 요청자-송신자 콜백 쌍에서)일 수 있습니다.

**채널이 메시지를 이동시키는지 확인**

전송 큐에 대한 문제가 있는 경우 채널이 메시지를 이동시키는지 확인하십시오.

**시작하기 전에**

명령 DIS CHSTATUS(QM1.TO.QM2) ALL을 발행하십시오. STATUS 필드의 값이 RUNNING이면 채널이 파트너 시스템에 연결된 것입니다.

256 페이지의 『큐의 메시지가 사용 가능한지 확인』에 설명된 대로, 전송 큐에 커밋되지 않은 메시지가 없는지 확인하십시오.

**이 태스크 정보**

채널의 가져오기 및 송신에 사용 가능한 메시지가 있으면 다음 검사를 수행하십시오.

**프로시저**

1. 채널 상태 표시 명령 DIS CHSTATUS(QM1.TO.QM2) ALL의 출력에서 다음 필드를 확인하십시오.

**MSGS**

이 세션 동안(채널이 시작된 이후) 송신했거나 수신한 메시지의 수(또는 서버 연결 채널의 경우, 처리된 MQI 호출 수).

**BUFSENT**

송신된 전송 버퍼의 수. 여기에는 제어 정보만을 송신하는 전송이 포함됩니다.

**BYTSENT**

이 세션 동안 송신된 바이트 수(채널이 시작된 이후로). 여기에는 메시지 채널 에이전트가 송신한 제어 정보가 포함됩니다.

**LSTMSGDA**

마지막 메시지가 송신되었거나 MQI 호출이 처리된 날짜(LSTMSGTI 참조).

**LSTMSGTI**

마지막 메시지가 송신되었거나 MQI 호출이 처리된 시간. 송신자 또는 서버의 경우에는 마지막 메시지(메시지가 분할된 경우에는 메시지의 마지막 부분)가 송신된 시간입니다. 요청자 또는 수신자의 경우에는 마

지막 메시지가 대상 큐에 넣어진 시간입니다. 서버 연결 채널의 경우에는 마지막 MQI 호출이 완료된 시간입니다.

### CURMSGSGS

송신 채널의 경우, 현재 배치에서 송신된 메시지 수. 수신 채널의 경우, 이는 현재 배치에서 수신된 메시지 수입니다. 배치가 예약되면, 송신 및 수신 채널 둘 다에 대해 값이 0으로 재설정됩니다.

2. 채널이 시작된 이후에 메시지를 송신했는지 판별하십시오. 송신한 경우 언제 마지막 메시지를 송신했는지 판별하십시오.
3. 채널이 아직 완료되지 않은 배치를 시작한 경우에는 CURMSGSGS의 0이 아닌 값이 표시하는 것처럼, 채널의 다른 쪽 끝이 배치를 수신확인할 때까지 채널이 대기 중인 것일 수 있습니다. 출력에서 SUBSTATE 필드를 보고 261 페이지의 표 31의 내용을 참조하십시오.

표 31. 송신자 및 수신자 MCA 하위 상태		
송신자 SUBSTATE	수신자 SUBSTATE	참고사항
MQGET	RECEIVE	휴면 중인 채널의 정상 상태.
SEND	RECEIVE	SEND는 일반적으로 일시적 상태입니다. SEND가 표시될 경우 이는 통신 프로토콜 버퍼가 채워졌음을 나타냅니다. 네트워크 문제 점을 나타내는 것일 수 있습니다.
RECEIVE		송신자가 일정 기간 동안 RECEIVE 하위 상태로 표시되는 경우 송신자가 배치 완료 또는 하트비트에 대한 응답을 대기 중인 것입니다. 배치 완료에 시간이 오래 걸리는 이유를 확인할 수 있습니다.

**참고:** 특히, 채널의 하위 상태가 엑시트 처리와 연관된 경우 채널이 메시지를 충분히 빠르게 처리할 수 있는지 여부를 판별할 수도 있습니다.

## 배치 완료에 시간이 오래 걸리는 이유 확인

일괄처리가 완료되는 데 오랜 시간이 걸리는 몇 가지 이유를 보려면 이 페이지를 사용하십시오.

### 이 태스크 정보

송신자 채널은 메시지 배치를 송신한 후 채널이 파이프라인되었는지 여부와 무관하게, 수신자로부터 해당 배치의 확인을 대기합니다. 다음은 송신자 채널이 대기하는 시간에 영향을 미칠 수 있는 요소입니다.

### 프로시저

- 네트워크가 느린지 확인하십시오.  
느린 네트워크는 배치를 완료하는 데 걸리는 시간에 영향을 미칠 수 있습니다. NETTIME 필드의 지표를 생성하는 측정용 배치 끝에서 측정됩니다. 하지만 느린 네트워크의 영향을 받은 첫 번째 배치는 배치 끝에서 측정되었기 때문에 NETTIME 값의 변경이 표시되지 않습니다.
- 채널이 메시지 재시도를 사용 중인지 확인하십시오.  
수신자 채널에서 대상 큐에 메시지를 넣는 데 실패하는 경우 해당 메시지를 즉시 데드레터 큐에 넣지 않고 메시지 재시도 처리를 사용할 수도 있습니다. 재시도 처리는 배치 속도를 느리게 할 수 있습니다. MQPUT 시도 사이에 채널의 상태가 메시지 재시도 간격이 패스되기를 대기 중임을 나타내는 STATUS(PAUSED)가 됩니다.

## 채널이 메시지를 충분히 빨리 처리할 수 있는지 판별

메시지가 전송 큐에 빌드되고 있지만 처리 문제점이 발견되지 않은 경우 채널이 메시지를 충분히 빨리 처리할 수 있는지 판별하십시오.

### 시작하기 전에

다음 명령을 일정 기간 반복적으로 발행하여 채널에 대한 성능 데이터를 수집하십시오.

```
DIS CHSTATUS(QM1.TO.QM2) ALL
```

## 이 태스크 정보

256 페이지의 『큐의 메시지가 사용 가능한지 확인』에 설명된 대로, 전송 큐에 커밋되지 않은 메시지가 없는지 확인한 후 채널 상태 표시 명령의 출력에서 XQTIME 필드를 확인하십시오. XQTIME 지표의 값이 일관되게 높거나 측정 기간 중에 증가하면 채널이 넣고 있는 애플리케이션을 따라잡지 못하는 것입니다.

다음 테스트를 수행하십시오.

## 프로시저

### 1. 엑시트가 처리 중인지 확인하십시오.

이 메시지를 전달 중인 채널에 엑시트가 사용되는 경우 엑시트가 메시지 처리에 소모되는 시간을 늘릴 수 있습니다. 이러한 경우인지 식별하려면 다음 검사를 수행하십시오.

a) DIS CHSTATUS(QM1.TO.QM2) ALL 명령의 출력에서 EXITTIME 필드를 확인하십시오.

엑시트에 소모된 시간이 예상보다 많으면 엑시트의 처리에서(특히 메시지, 송신, 수신 엑시트에서) 불필요한 루프나 여분의 처리를 검토하십시오. 이러한 처리는 채널에서 이동하는 모든 메시지에 영향을 줍니다.

b) DIS CHSTATUS(QM1.TO.QM2) ALL 명령의 출력에서 SUBSTATE 필드를 확인하십시오.

채널이 상당한 시간 동안 다음 하위 상태 중 하나로 유지되면 엑시트의 처리를 검토하십시오.

- SCYEXIT
- RCVEXIT
- SENDEXIT
- MSGEXIT
- MREXIT

### 2. 네트워크가 느린지 확인하십시오.

메시지가 채널에서 충분히 빠르게 이동하지 않는 경우 이는 네트워크가 느리기 때문일 수 있습니다. 이러한 경우인지 식별하려면 다음 검사를 수행하십시오.

a) DIS CHSTATUS(QM1.TO.QM2) ALL 명령의 출력에서 NETTIME 필드를 확인하십시오.

이 지표는 송신 채널이 파트너에 응답을 요청할 때 측정됩니다. 이는 각 배치의 끝에서 그리고 하트비트 중에 채널이 유휴 상태일 때 발생합니다.

b) 이 지표가 라운드트립에 예상보다 시간이 오래 걸린다고 표시하는 경우에는 다른 네트워크 모니터링 도구를 사용하여 네트워크 성능을 조사하십시오.

### 3. 채널이 압축을 사용 중인지 확인하십시오.

채널이 압축을 사용하면 메시지 처리 시간이 증가합니다. 채널이 압축 알고리즘을 하나만 사용 중인 경우 다음 검사를 수행하십시오.

a) DIS CHSTATUS(QM1.TO.QM2) ALL 명령의 출력에서 COMPTIME 필드를 확인하십시오.

이 지표는 압축 또는 압축 해제 중에 소모된 시간을 표시합니다.

b) 선택된 압축이 송신할 데이터 양을 예상하는 만큼 감소시키지 않으면 압축 알고리즘을 변경하십시오.

### 4. 채널이 여러 압축 알고리즘을 사용 중인 경우에는 다음 검사를 수행하십시오.

a) DIS CHSTATUS(QM1.TO.QM2) ALL 명령의 출력에서 COMPTIME, COMPHDR 및 COMPMSG 필드를 확인하십시오.

b) 압축률 또는 알고리즘 선택이 필요한 압축 또는 성능을 제공하지 않는 경우에는, 채널 정의에 지정된 압축 알고리즘을 변경하거나 특정 메시지에 대한 채널의 압축 알고리즘 선택을 대체할 메시지 엑시트를 쓸 것을 고려하십시오.

## 클러스터 채널에 대한 문제점 해결

SYSTEM.CLUSTER.TRANSMIT.QUEUE 큐에 메시지를 빌드하는 경우 문제점 진단의 첫 단계는 메시지 전달 문제가 있는 단일 또는 다중 채널을 찾는 것입니다.

## 이 태스크 정보

SYSTEM.CLUSTER.TRANSMIT.QUEUE를 사용하여 메시지 전달 문제가 있는 단일 또는 다중 채널을 찾으려면 다음 검사를 수행하십시오.

### 프로시저

1. 다음 명령을 실행하십시오.

```
DIS CHSTATUS(*) WHERE(XQMSGSA GT 1)
```

**참고:** 많은 메시지가 이동하는 사용 중인 클러스터가 있으면 전달 가능한 몇 개의 메시지만 있는 채널을 제거하기 위해 이 명령을 보다 높은 수로 발행할 것을 고려하십시오.

2. 출력에서 XQMSGSA 필드의 값이 큰 채널을 찾아보십시오. 채널이 메시지를 이동시키지 않거나 충분히 빠르게 이동시키지 않는 이유를 판별하십시오. 258 페이지의 『채널 모니터링』에 개괄된 태스크를 사용하여 빌드를 유발하는 채널에 대한 문제점을 진단하십시오.

### 전송 큐 전환 모니터링

엔터프라이즈에 미치는 영향이 최소화되도록 전송 큐를 전환하는 클러스터 송신자 채널의 프로세스를 모니터링하는 것이 중요합니다. 예를 들어, 워크로드가 많거나 여러 채널을 동시에 전환하여 이 프로세스를 시도해서는 안 됩니다.

### 채널을 전환하는 프로세스

채널을 전환하는 데 사용되는 프로세스는 다음과 같습니다.

1. 채널은 입력을 위해 새 전송 큐를 열고 이 큐에서 메시지를 받기 시작합니다 (상관 ID로 가져오기를 사용하여).
2. 백그라운드 프로세스는 큐 관리자가 이전 전송 큐에서 새 전송 큐로 채널 큐에 있는 메시지를 이동하도록 시작됩니다. 메시지가 이동되는 동안 채널의 새 메시지가 이전 전송 큐에 대기되어 시퀀스를 보존합니다. 이전 전송 큐에 채널에 대한 많은 수의 메시지가 있거나 새 메시지가 빠르게 도착하는 경우 이 프로세스를 완료하는데 시간이 걸릴 수 있습니다.
3. 이전 전송 큐의 채널에 대해 커밋되거나 커밋되지 않은 메시지가 큐에 남아 있지 않은 경우, 스위치가 완료됩니다. 이제 새 메시지가 새 전송 큐에 직접 배치됩니다.

동시에 여러 채널이 전환되는 것을 방지하기 위해 IBM WebSphere MQ는 **runswch1** 명령을 사용하여 실행 중인 아닌 하나 이상의 채널의 전송 큐를 전환하는 기능을 제공합니다.

### 스위치 작동 상태 모니터링

스위치 조작 관리자의 상태를 이해하려면 다음 조치를 수행할 수 있습니다.

- 큐 관리자 오류 로그 (AMQERR01.LOG) 를 사용하십시오. 여기서 메시지는 조작 중 다음 단계를 표시하기 위해 출력됩니다.
  - 스위치 조작이 시작되었습니다.
  - 메시지 이동이 시작되었습니다.
  - 이동을 위해 남아 있는 메시지 수에 대한 정기적 업데이트 (스위치 조작이 신속하게 완료되지 않는 경우)
  - 메시지 이동이 완료되었습니다.
  - 스위치 작동이 완료되었습니다.
- 
- 각 클러스터 송신자 채널이 현재 사용 중인 전송 큐를 조회하려면 DISPLAY CLUSQMGR 명령을 사용한다.
- 조회 모드에서 **runswch1** 명령을 실행하여 하나 이상의 채널의 전환 상태를 확인하십시오. 이 명령의 출력은 각 채널에 대해 다음을 식별합니다.
  - 채널에 오류 중인 스위치 조작이 있는지 여부
  - 채널이 전환하고 있는 전송 큐

- 이전 전송 큐에 남아 있는 메시지 수

각 명령은 하나의 호출에서 모든 채널의 상태, 구성 변경에 대한 영향 및 모든 스위치 조작이 완료되었는지 여부를 판별할 수 있기 때문에 매우 유용합니다.

### **발생할 수 있는 잠재적 문제**

전송 큐, 해당 원인 및 가장 가능성 있는 솔루션을 전환할 때 발생할 수 있는 일부 문제 목록은 [전송 큐를 전환할 때 잠재적인 문제](#) 를 참조하십시오.

## **Windows 성능 모니터**

WebSphere MQ 버전 7.0 이전 버전인 경우, Windows 성능 모니터를 사용하여 Windows 시스템에서 로컬 큐의 성능을 모니터할 수 있습니다. WebSphere MQ 버전 7.1의 경우, 성능 모니터링이 더 이상 사용 가능하지 않습니다.

252 페이지의 [『실시간 모니터링』](#)에 설명된 방법을 사용하여 지원되는 모든 플랫폼에서 큐를 모니터할 수 있습니다.

## 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 재산을 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 라이선스까지 부여하는 것은 아닙니다. 라이선스에 대한 의문사항은 다음으로 문의하십시오.

150-945  
서울특별시 영등포구  
국제금융로 10, 3IFC  
한국 아이.비.엠 주식회사  
U.S.A.

2바이트(DBCS) 정보에 관한 라이선스 문의는 한국 IBM에 문의하거나 다음 주소로 서면 문의하시기 바랍니다.

지적 재산권 라이선스 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 라이선스 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 라이선스가 부여된 프로그램 및 프로그램에 대해 사용 가능한 모든 라이선스가 부여된 자료는 IBM이 IBM 기본 계약, IBM 프로그램 라이선스 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 추정

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.

저작권 라이선스:

이 정보에는 여러 운영 플랫폼에서의 프로그래밍 기법을 보여주는 원어로 된 샘플 응용프로그램이 들어 있습니다. 귀하는 이러한 샘플 프로그램의 작성 기준이 된 운영 플랫폼의 응용프로그램 프로그래밍 인터페이스(API)에 부합하는 응용프로그램을 개발, 사용, 판매 또는 배포할 목적으로 IBM에 추가 비용을 지불하지 않고 이들 샘플 프로그램을 어떠한 형태로든 복사, 수정 및 배포할 수 있습니다. 이러한 샘플 프로그램은 모든 조건하에서 완전히 테스트된 것은 아닙니다. 따라서 IBM은 이들 샘플 프로그램의 신뢰성, 서비스 가능성 또는 기능을 보증하거나 진술하지 않습니다.

이 정보를 소프트웨어로 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.

This book contains information on intended programming interfaces that allow the customer to write programs to obtain the services of IBM WebSphere MQ.

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](http://www.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](http://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: