

MQSeries® Everyplace



소개

버전 1

MQSeries® Everyplace



소개

버전 1

주의!

이 책과 이 책이 지원하는 제품을 사용하기 전에 71 페이지의 『부록. 주의사항』의 "주의사항"을 읽으십시오.

사용권 경고

MQSeries Everyplace 버전 1은 사용자가 MQSeries Everyplace 응용프로그램을 작성하고 그것을 수행할 환경을 조성하게 해주는 도구모음입니다.

도구모음을 구매한 사용권 조건에 따라 사용할 수 있는 환경이 달라집니다.

장치(클라이언트)로 사용하기 위해 *MQSeries Everyplace*를 구입한 경우, ***MQSeries Everyplace*** 채널 관리자나 ***MQSeries Everyplace*** 채널 리스너를 작성하기 위해서 이를 사용할 수는 **없습니다**.

MQSeries Everyplace 채널 관리자나 ***MQSeries Everyplace*** 채널 리스너의 존재로 게이트웨이 사용권을 요구하는 게이트웨이(서버) 환경이 정의됩니다.

초판(2000년 6월)

이 책은 새 개정판에 달리 언급될 때까지는 MQSeries Everyplace 버전 1.0 및 모든 후속 릴리스와 수정판에 적용됩니다.

© Copyright International Business Machines Corporation 2000. All rights reserved.

목차

그림	v	이벤트 로그	35
표	vii	MQSeries Everyplace 네트워크	36
이 책에 관하여	ix	구성 및 확장성	36
이 책의 사용자	ix	비동기 메시지 전달	38
사전 지식	ix	동기 메시지 전달	39
이 책에서 사용되는 용어	ix	보안	39
제1장 개요	1	MQSeries Everyplace 지역 보안	41
버전 1.0의 주의사항	1	MQSeries Everyplace 큐 기반 보안	41
제2장 전체 조건	3	메시지 레벨 보안	43
제3장 MQSeries 제품군	5	레지스트리	45
제4장 요구사항	11	MQSeries Everyplace 인증 가능 항목	45
기능	11	개인 레지스트리 및 신임장	46
응용프로그램	12	자동 등록	46
고객 요구사항	12	공용 레지스트리 및 인증서 복제	47
제5장 제품 개념	15	레지스트리 서비스의 응용프로그램 사용	47
소개	15	기본 Mini Certificate 발행 서비스	47
메시지 오브젝트	16	보안 인터페이스	48
덤프 데이터 형식	20	구성 및 조정	49
큐	21	규칙	49
큐 관리자	26	연결 유형	51
큐 관리자 조작	28	피어-투-피어 연결	52
관리	30	클라이언트-서버 연결	52
관리 메시지	30	여러 연결 유형	53
선택적 관리	32	클래스	53
모니터링 및 관련 조치	32	응용프로그램 로드	53
동적 채널	33	제6장 MQSeries Everyplace 및 MQSeries	
어댑터	34	네트워크	55
전화 접속 연결 관리	35	MQSeries에 대한 인터페이스	55
추적	35	메시지 변환	62
		기능	64
		호환성	64
		보증 전달	66

제7장 응용프로그램 및 유틸리티	67	등록상표	72
Postcard	67	용어	73
MQSeries Everyplace 탐색기	67	참고 문헌	77
제8장 프로그래밍 인터페이스	69	색인	79
부록. 주의사항	71		

그림

1. MQSeries 제품군	5	5. 단순한 MQSeries Everyplace 네트워크	37
2. 단순 호스트 및 분산 구성	6	6. 스타 MQSeries Everyplace 네트워크	38
3. 일반적인 워크스테이션 구성	7	7. MQSeries 브릿지 오브젝트 계층	57
4. 일반적인 장치 구성	9		

표

1. 버전 1 지원 소프트웨어 환경	3	11. 관리 메시지 클래스	30
2. MQSeries Everyplace 및 MQSeries 요 소	9	12. 관리 메시지의 일반 구조	31
3. 필드 오브젝트 및 구성 필드 등록정보	17	13. 인증, 암호화 및 압축 지원	40
4. 속성 오브젝트 등록정보	18	14. 연결 유형	51
5. 메시지 오브젝트 등록정보	19	15. 클래스 옵션	53
6. 준비된 메시지 오브젝트 필드	19	16. 브릿지 오브젝트 등록정보	57
7. 큐 등록정보	24	17. 브릿지 등록정보	58
8. 로컬 큐 관리자 등록정보	27	18. MQSeries 큐 관리자 프록시 등록정보	58
9. 연결(리모트 큐 관리자) 등록정보	28	19. 클라이언트 연결 서비스 등록정보	59
10. MQSeries Everyplace 큐에서의 메시지 조작	29	20. 리스너 등록정보	60
		21. MQSeries 리모트 큐 등록정보	61

이 책에 관하여

이 책은 MQSeries Everyplace에 대한 일반적인 소개입니다. 여기에서는 제품 개념 및 다른 MQSeries 제품과의 관계에 대해 다룹니다.

MQSeries Everyplace API 및 이를 사용하여 MQSeries Everyplace 응용프로그램을 작성하는 방법에 대한 내용은 *MQSeries Everyplace Programming Reference* 및 *MQSeries Everyplace Programming Guide*를 참조하십시오.

이 책의 사용자

이 책은 센서, 전화, PDA(Personal Digital Assistants) 및 랩탑 컴퓨터와 같은 경량 장치에서의 보안 메시징 사용에 관심을 가지고 있는 사람을 위한 것입니다.

사전 지식

이 정보를 읽는 데 어떤 사전 지식도 필요하지는 않지만, 보안 메시징 개념을 미리 알고 있으면 더 유리합니다.

이러한 이해가 없을 경우, 다음의 MQSeries 서적들을 읽으면 유용할 것입니다.

- *MQSeries An Introduction to Messaging and Queuing*
- *MQSeries for Windows NT[®] V5R1 Quick Beginnings*

이 책들은 온라인 MQSeries 라이브러리의 서적 섹션에서 소프트 카피로 사용할 수 있습니다. 이것은 MQSeries 웹 사이트, URL 주소 <http://www.ibm.com/software/ts/MQSeries/library/>를 통해 볼 수 있습니다.

이 책에서 사용되는 용어

다음 용어들이 이 책에서 사용됩니다.

MQSeries 제품군

5 페이지의 『제3장 MQSeries 제품군』에 설명된 MQSeries 제품들의 모음을 말합니다.

MQSeries 메시징

5 페이지의 『제3장 MQSeries 제품군』에 설명된 네 가지의 메시징 제품 그룹을 말합니다.

MQSeries

다음의 세 가지 MQSeries 메시징 제품 그룹을 말합니다.

- 분산 메시징
- 호스트 메시징
- 워크스테이션 메시징

MQSeries Everyplace

네번째 MQSeries 메시징 제품 그룹, 보급 메시징을 말합니다.

장치 MQSeries Everyplace 프로그램을 실행하지만 *MQSeries Everyplace* 채널 관리자나 *MQSeries Everyplace* 채널 리스너 오브젝트가 설치되지 않은 임의 규모의 컴퓨터.

주: 사용권 부여 목적으로 사용될 때, 장치는 *MQSeries Everyplace* 클라이언트의 동의어입니다.

게이트웨이

MQSeries Everyplace 프로그램을 실행하면서 *MQSeries Everyplace* 채널 관리자나 *MQSeries Everyplace* 채널 리스너 오브젝트가 설치된 임의 규모의 컴퓨터.

주: 사용권 부여 목적으로 사용될 때, 게이트웨이는 *MQSeries Everyplace* 서버의 동의어입니다.

제1장 개요

MQSeries Everyplace는 비즈니스 수준의 메시징 제품인 MQSeries제품군 구성원입니다. 이는 센서, 전화, PDA(Personal Digital Assistant) 및 랩탑 컴퓨터와 같은 경량 장치의 메시징 필요뿐만 아니라, 이동시 접속 요구 및 부서지기 쉬운 통신 네트워크를 사용할 때 발생할 수 있는 요구사항을 충족시키기 위해 설계되었습니다. 이것은 표준 MQSeries 1회 보증 전달을 제공하고, 다른 제품군 구성원들과 메시지를 교환합니다. 많은 MQSeries Everyplace 응용프로그램이 인터넷 방화벽의 보호를 벗어나서 실행되므로, 정교한 보안 기능도 제공합니다.

경량 장치는 메시징 서브시스템이 시스템 자원을 절약하면서 사용하도록 요구하므로, MQSeries Everyplace는 시스템 footprint와 프로토콜 효율성을 위해 최적화되었습니다. 다른 메시징 MQSeries 제품군 구성원에 동일한 기능을 제공하지는 않지만, 끊임 없는 상호 작동을 제공합니다. MQSeries Everyplace에는 이동, 지역 및 원격 메시지 액세스, 신뢰할 수 없는 네트워크를 거치는 메시지에 대한 보안 및 지원에 대한 포괄적인 규정을 가지고 있습니다.

MQSeries Everyplace는 IBM® 보급 컴퓨팅 제품군의 구성원으로 다른 IBM 보급 및 무선 제품들과도 통합되도록 계속해서 설계되고 있습니다.

버전 1.0의 주의사항

- MQSeries Everyplace의 버전 1.0은 사용자가 MQSeries Everyplace 응용프로그램을 작성하고 그것을 수행할 환경을 조성하게 해주는 도구모음입니다.
- 이 릴리스에서, MQSeries Everyplace를 보급 장치에 전개하는 것은 응용프로그램 및 솔루션 제공자의 책임입니다.

제2장 전제 조건

표1에서는 MQSeries Everyplace 버전 1을 수행하는 데 사용할 수 있는 소프트웨어 환경을 보여줍니다. ¹

표 1. 버전 1 지원 소프트웨어 환경

	운영 체제
장치	EPOC
	Palm OS
	Windows CE
	Windows [®] 95 Windows 98 Windows NT v4 Windows 2000
게이트웨이	Windows NT v4 Windows 2000
<p>주:</p> <ol style="list-style-type: none"> 1. Java를 지원하는 모든 플랫폼에 사용하도록 버전 1.0이 Java에 제공됩니다. 2. 리모트 큐에 대한 동기 액세스만을 제공하는 한정된 기능 클라이언트는 Palm OS에서 C 코드 베이스로 쓸 때에만 사용 가능합니다. 3. 플랫폼에서 사용 가능한 최신 레벨의 Java 1.1을 권장합니다. 테스트한 Java 레벨에 대한 자세한 내용은 MQSeries Everyplace 웹 사이트(www.ibm.com/software/mqseries/everyplace)를 참조하십시오. 	

1. MQSeries Everyplace 장치 코드는 Java[®]를 수행하는 어떤 장치에서도 수행될 수 있으나, 표1에 나열된 운영 체제에서만 테스트되었습니다.

제3장 MQSeries 제품군

MQSeries 제품군에는 그림1에 설명된 대로 다양한 기능을 제공하는 여러 제품들이 포함됩니다.

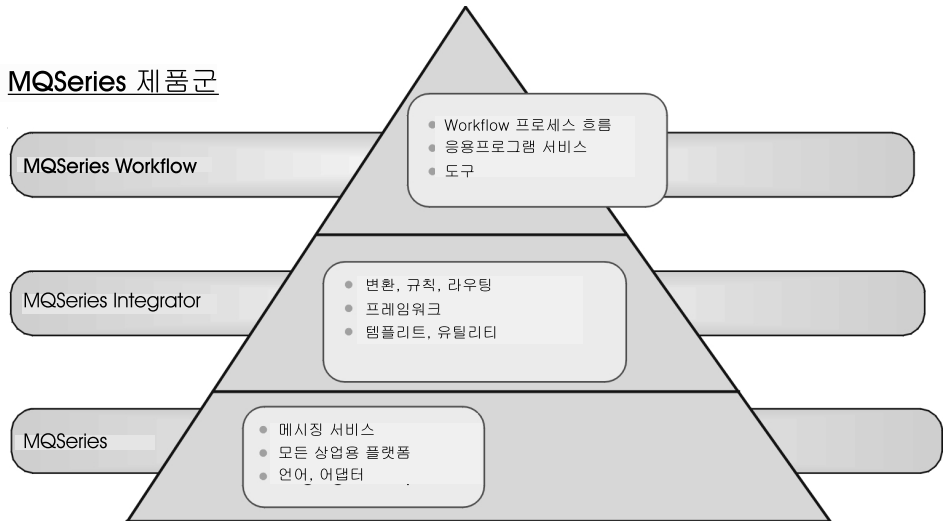


그림 1. MQSeries 제품군

- **MQSeries Workflow**는 입력과 응용프로그램을 포함하는 비즈니스 프로세스를 자동화함으로써 전사적 통합을 단순화합니다.
- **MQSeries Integrator**는 실시간의 지능형 규칙 기반 메시지 경로 지정과, 내용 전송 및 형식화를 제공하는 강력한 메시지 중개 소프트웨어입니다.
- **MQSeries Messaging**은 비즈니스 수준의 메시징을 통해 데스크탑에서 메인프레임으로의 연결성을 지원되는 35개 플랫폼에 걸쳐 제공합니다.

MQSeries Workflow 및 MQSeries Integrator 양쪽 모두 MQSeries 메시징 층에서 제공하는 연결성 이점이 있습니다.

MQSeries 제품군 메시징은 MQSeries(MQS) 및 MQSeries Everyplace 제품에 의해 제공됩니다. 각각은 하나 이상의 하드웨어 서버 플랫폼 및/또는 관련 운영 체제를 지원하도

록 설계되어 있습니다. 광범위한 플랫폼 기능과 함께, 이러한 개별적인 제품은 공통적인 기능과 설계를 가지는 제품 그룹으로 구성됩니다. 그러한 4개의 제품 그룹이 존재합니다.

- 분산 메시징: Windows NT, AIX®, AS/400®, HP-UX, Sun Solaris 및 기타 플랫폼용 MQSeries
- 호스트 메시징: OS/390®용 MQSeries
- 워크스테이션 메시징: Windows3.1, 95, 98용 MQSeries
- 보급 메시징: MQSeries Everyplace

특정 제품 또는 제품 그룹에 관계없이 메시징 자체는 큐 관리자를 기반으로 합니다. 큐 관리자는 메시지를 각각 저장할 수 있는 큐들을 관리합니다. 응용프로그램은 로컬 큐 관리자와 통신한 후 메시지를 큐에서 가져오거나 큐로 보냅니다. 리모트 큐 관리자 소유의 리모트 큐에 메시지가 넣어지면, 메시지는 채널을 통해 리모트 큐 관리자로 전송됩니다. 이러한 방식으로 메시지들은 목적지에 도달하기 전에 하나 또는 그 이상의 중간 큐 관리자를 뛰어넘을 수 있습니다. 메시징의 본질은 수신 응용프로그램, 혹은 필요한 경우 가는 도중의 큐 처리 메시지와 송신 응용프로그램의 결합도를 줄이는 것입니다. 모든 MQSeries 메시징 제품들은 세부적으로 여러 면에서 차이가 있을지라도 큐 관리자, 큐, 메시지 및 채널의 동일한 기본 요소와 관련됩니다.

MQSeries 호스트 및 분산 메시징 제품들은 서로 다른 수많은 네트워크 구성을 지원하는 데 사용되며, 그림 2에서 보여주는 식으로 모두 클라이언트 및 서버와 관련이 있습니다.² 그림2에서 일부 예를 보여줍니다.

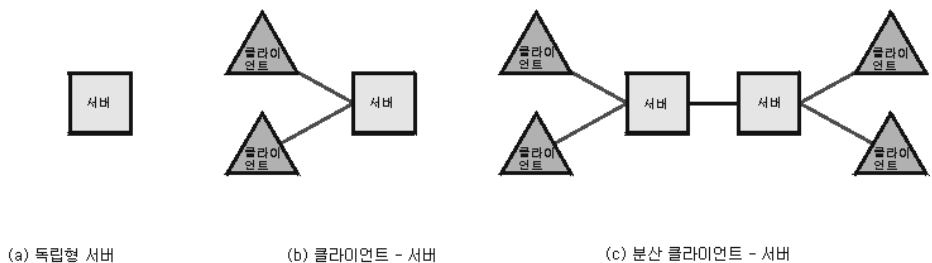


그림 2. 단순 호스트 및 분산 구성

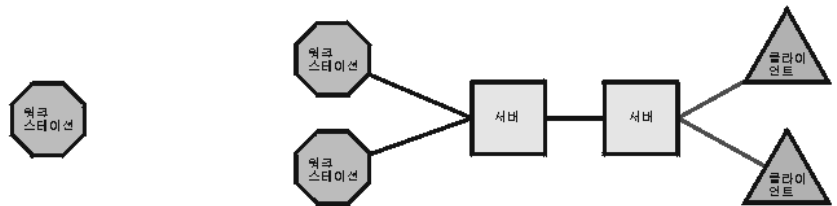
2. 이 용어들은 MQSeries에서 일반적인 사용과 일치하지 않는 보다 특별한 의미를 가진다는 점에 유의하십시오. 이 문서에서 이 용어들은 항상 MQSeries 의미로 사용됩니다.

가장 간단한 케이스에서는 큐 관리자를 실행하는 독립형 서버가 구성됩니다. 하나 이상의 응용프로그램들이 큐를 통해 메시지들을 교환하면서 그 서버에서 실행됩니다. 대체 구성이 클라이언트-서버입니다. 여기에서 큐 관리자는 서버에만 존재하지만 각 클라이언트들은 클라이언트 채널을 통해 이에 액세스합니다. 클라이언트 채널은 원격 프로시더어 호출(RPC)과 유사한 것을 구현하여 고유 MQSeries 프로토콜을 지나가는 양방향 통신 링크입니다. 응용프로그램들은 서버 큐에 액세스하여 클라이언트상에서 수행할 수 있습니다. 클라이언트-서버 구성의 한 장점은 클라이언트 메시징 하부 구조가 서버 큐 관리자에 종속적이어서 경량이라는 것입니다. 단점은 클라이언트와 그 관련 서버가 동시에 작동하므로 클라이언트 채널이 항상 사용 가능해야 한다는 점입니다.

분산 클라이언트-서버 구성에서는 다중 서버가 포함된 보다 복잡한 경우를 보여줍니다. 이 구성에서는 서버들이 메시지 채널을 통해 메시지를 교환합니다. 메시지 채널은 메시지 데이터의 안전한 비동기 교환을 위해 설계된 프로토콜을 가지며, 단방향입니다. 통신이 사용 불가능하여 서버 간에 메시지가 흐르지 않더라도, 클라이언트가 처리를 계속하기 위해 서 이 메시지 채널들이 사용 가능해야 할 필요는 없습니다.

MQSeries 워크스테이션 메시징 제품들은 이 구성 옵션의 부속 세트를 제공합니다. 서버 대신에, 이들은 큐 관리자가 있는 워크스테이션은 지원하지만 클라이언트 접속은 지원하지 않습니다. 그러나, 워크스테이션들은 MQSeries 메시지 채널을 통해 다른 워크스테이션 및 서버에도 접속할 수 있습니다. 그리하여 워크스테이션들은 일반적으로 경량 서버로 간주되며, 비동기 기능이 필요한 클라이언트 대신에 사용됩니다.

그림3에서는 두 개의 일반적인 워크스테이션 구성을 보여줍니다. (b)에서 워크스테이션 응용프로그램은 서버 및 클라이언트와 독립적으로 수행할 수 있습니다.



(a) 독립형 워크스테이션

(b) 분산 클라이언트/워크스테이션/서버

그림 3. 일반적인 워크스테이션 구성

보급형 메시징 제품 MQSeries Everyplace는 장치와 게이트웨이를 제공하여 구성을 지원합니다.

MQSeries Everyplace 장치는 채널 관리자 없이 MQSeries Everyplace 코드를 실행하는 컴퓨터입니다. 이는 장치가 한번에 단 하나의 다른 장치나 게이트웨이와 통신하도록 제한된다는 것을 의미합니다. MQSeries Everyplace 장치는 매우 작은(송유관의 센서와 같은) 것에서부터 더 큰(전화, 개인 데이터 지원(PDA) 또는 랩탑 컴퓨터와 같은) 것에 이르기까지 광범위합니다. 종종 그러한 장치 컴퓨터를 보급 장치라고 하며, 이것은 제품에는 없는 크기 및 기능상의 제한사항을 수반합니다.

게이트웨이는 MQSeries Everyplace 채널 관리자나 MQSeries Everyplace 채널 리스너가 구성된, MQSeries Everyplace 코드를 실행하는 컴퓨터입니다. 이것은 장치 코드의 모든 기능과, 추가로 여러 장치 게이트웨이와 동시에 통신할 수 있는 능력을 제공합니다. 게이트웨이는 또한 MQSeries Everyplace 네트워크와 MQSeries 네트워크 사이의 메시지 교환을 위한 메카니즘도 제공합니다.

처음에 한번 예측해 보기 위해 장치는 클라이언트와 서버의 많은 속성을 결합합니다. 그 장치는 비동기 조작이 가능하도록 완전한 큐 처리 기능으로 구성될 수 있습니다. 또한, 원격으로 보유된 큐에 액세스할 수도 있습니다. 이는 서버 큐에 대한 클라이언트 액세스와 유사합니다. 서버와는 달리, 장치는 클라이언트를 접속시킬 수 없습니다. 장치는 피어-투-피어 메시징 기능을 통해 직접 서로 통신할 수 있습니다. 또한 채널을 통해서도 통신하지만, 이 채널들은 MQSeries Everyplace에 고유하며 MQSeries 클라이언트 채널 및 MQSeries 메시지 채널과 구별하기 위해 동적 채널이라고 합니다. 동적 채널은 양방향으로, 동기식 및 비동기식 메시징을 포함하여 MQSeries Everyplace에서 제공하는 모든 범위의 기능을 지원합니다.

게이트웨이는 장치와의 통신을 위해 반드시 MQSeries Everyplace 동적 채널을 지원합니다. 게이트웨이는 서버와의 통신을 위해 선택적으로 MQSeries 클라이언트 채널을 지원할 수 있습니다. 서버처럼, 게이트웨이는 큐 관리자를 가지고 있으며, 지역 메시징 응용프로그램을 지원할 수 있습니다.

일반적인 보급장치 구성이 9 페이지의 그림4에 나와 있습니다.

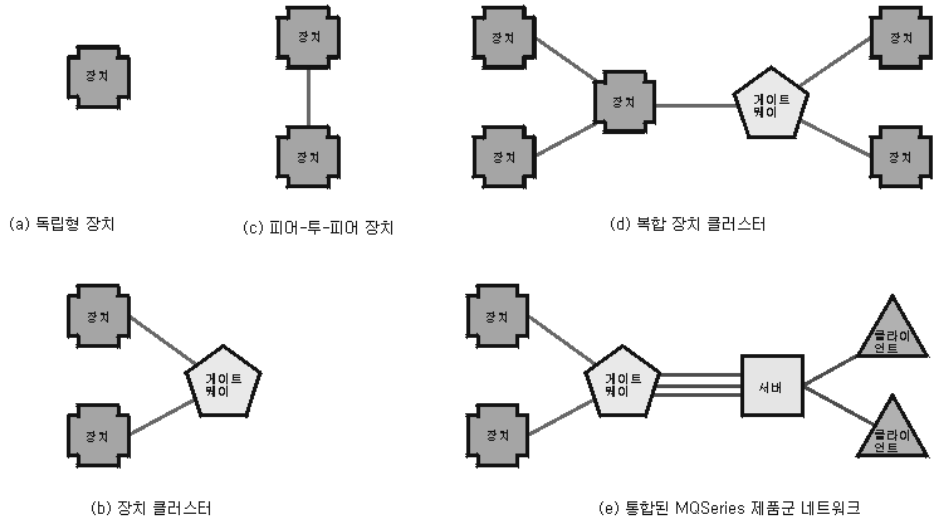


그림 4. 일반적인 장치 구성

그림 4 (b)에서, 게이트웨이는 장치들을 함께 연결하기 위해 사용됩니다. 게이트웨이 기능은 한번에 단 하나의 요청만 처리할 수 있는 장치와는 반대로, 수신되는 여러 연결 요청들을 동시에 처리할 수 있습니다. 게이트웨이와 장치 모두 요청 송신은 동시에 여러 번 할 수 있습니다. 구성 (b)에서 게이트웨이 대신 장치가 사용되었다면, 두 개의 단말기 장치가 동시에 접속할 수 있더라도 교대로 이 중간 매개 장치에 접속해야 합니다. (d)에서 장치들을 연결하기 위해 장치와 게이트웨이 둘 다가 사용됩니다. (e)에서 게이트웨이는 장치 네트워크를 MQSeries 서버(메시지가 모든 요소, 장치, 게이트웨이, 서버, 워크스테이션 및 클라이언트 사이에 이동될 수 있는 구성)에 링크하는 데 사용됩니다. 이 구성요소들의 가장 중요한 특성이 표 2에 나와 있습니다.

표 2. MQSeries Everyplace 및 MQSeries 요소

구성요소	특성	제공처
장치(MQSeries Everyplace)	<p>응용프로그램에 동적 채널을 통한 보증 메시지를 제공합니다.</p> <p>동기식 로컬 및 리모트 큐 액세스 둘 다를 허용합니다.</p> <p>리모트 큐로의 비동기 전달을 허용합니다.</p> <p>한번에 하나의 수신 요청을 처리하도록 제한됩니다.</p>	보급장치

표 2. MQSeries Everyplace 및 MQSeries 요소 (계속)

구성요소	특성	제공처
클라이언트(MQS)	<p>지역 응용프로그램에 보증 메시지를 제공합니다.</p> <p>서버에 대한 동기식 클라이언트 채널 연결을 요구합니다.</p> <p>접속된 서버에서만 큐에 대한 동기식 액세스를 허용합니다.</p> <p>접속된 서버를 통해 리모트 큐로의 비동기 전달을 허용합니다.</p>	분산 호스트
게이트웨이(MQSeries Everyplace)	<p>응용프로그램에 동적 채널을 통한 보증 메시지를 제공합니다.</p> <p>동기식 로컬 및 리모트 큐 액세스 둘 다를 허용합니다.</p> <p>리모트 큐로의 비동기 전달을 허용합니다.</p> <p>한번에 여러 개의 수신 요청을 처리할 수 있습니다.</p> <p>클라이언트 채널을 통해 여러 MQSeries 서버의 접속을 지원합니다.</p>	보급
서버(MQS)	<p>응용프로그램에 메시지 채널을 통한 보증 메시지를 제공합니다.</p> <p>동기식 로컬 큐 액세스를 허용합니다.</p> <p>리모트 큐로의 비동기 전달을 허용합니다.</p> <p>클라이언트 채널을 통해 여러 MQSeries 클라이언트의 접속을 지원합니다.</p>	분산 호스트
워크스테이션(MQS)	<p>응용프로그램에 메시지 채널을 통한 보증 메시지를 제공합니다.</p> <p>동기식 로컬 큐 액세스를 허용합니다.</p> <p>리모트 큐로의 비동기 전달을 허용합니다.</p>	워크스테이션

제4장 요구사항

이 장에서는 MQSeries Everyplace 설계 및 구현에 대한 요구사항을 설명합니다.

기능

MQSeries Everyplace는 다음과 같이 MQSeries 제품군의 메시징 범위를 확장합니다.

- PDA, 전화 및 센서와 같은 단말 장치를 지원하여, 그 장치들이 MQSeries 메시징 네트워크에 참여할 수 있도록 허용합니다. 또한 랩탑, 워크스테이션 및 특정 분산 플랫폼 등 중간 매개 장치도 지원합니다. MQSeries Everyplace는 동일한 수준의 서비스(메시지 1회 보증 전달)를 제공하며 제품군 구성원 사이에 메시지를 교환하도록 해줍니다.
- 저장된 혹은 전송 중인 메시지, 큐 및 관련 데이터를 보호하기 위한 확장된 보안 기능을 제공합니다.
- 네트워크가 불안정하거나 대역폭이 엄격히 제한되는 적대적 통신 환경에서 효율적으로 작동합니다. 효율적인 회선 프로토콜을 가지고 있어서 통신 연결이 실패할 때 자동으로 복구됩니다.
- 장치가 이동함에 따라 네트워크 연결점도 변경되도록 하여 이동 사용자를 지원합니다. 또한, 배터리 자원과 네트워크가 부족하거나 제한되는 상황에서 작동(behavior)을 제어할 수 있게 합니다.
- 적절하게 구성된 방화벽을 통해 작동합니다.
- 사용자에게 대한 관리 작업을 최소화하여, 장치에 있는 MQSeries Everyplace의 존재를 상당 부분 숨길 수 있게 합니다. 이로써, MQSeries Everyplace는 유틸리티 유형의 응용프로그램을 구축할 수 있는 적합한 기반이 됩니다.
- 응용프로그램이 제공하는 규칙이나 작동(behavior)을 수정하는 다른 클래스를 사용하여, 또는 다양한 메시지 유형을 나타내는 등의 작업을 위해 기본 오브젝트 클래스를 서브클래스화하여 쉽게 조정되고 확장됩니다.

응용프로그램

선택사항이 많고 다양하여 사용할 수 있는 MQSeries Everyplace 응용프로그램 목록에 제한이 없지만, 특정 사용자 그룹을 위해 개발된 사용자 정의 응용프로그램의 실제 숫자는 예상할 수 있습니다. 다음 목록은 검토한 예들 중에서 일부를 나열한 것입니다.

- **소비자 응용프로그램:** PDA를 사용하여 집에서 장보기, 여행자들의 항공사 선호도 수집, 휴대폰으로 금융 거래.
- **제어 응용프로그램:** 송유관 센서로부터 위성을 통해 전송된 데이터의 수집 및 통합, 조작자 검증을 위한 보안과 더불어 밸브 등 장비의 원격 조작.
- **이동 노동자:** 방문 직업인(보험 설계사), 택배 회사의 고객 영수증 빠른 발급, 주방과 정보를 교환하는 패스트 푸드점 직원, 골프 시험 점수 매기기, 경찰의 휴대용 보안 시스템 메시징 시스템, 통신이 자주 끊기는 상황에서의 설비 작업자에 대한 작업 정보, 가내 계량기 판독.
- **개인 생산성:** 우편/달력 복제, 데이터베이스 복제, 랩탑 다운사이징.

고객 요구사항

MQSeries Everyplace 설계에 영향을 준 요구사항은 다음과 같습니다.

- **관리:** 최소한의 설정 및 유지보수, 지역 및 원격 관리 지원, 특정 응용프로그램의 요구를 충족시키도록 관리 기능을 확장하고 조정하는 기능, 자동 전개 및 복구의 강조, 선택적으로 사용할 수 있는 개별 관리 요소 제공.
- **통신:** 아주 효율적인 회선 프로토콜, 최소한의 헤더, 메시지에서의 필수 필드 없애기(고유 식별자 제외), 데이터 인코딩을 변경할 수 있는 능력, 압축, 암호화 및 인증 지원, 압축성과 보안성 사이의 세부적인 조정, 방화벽을 쉽게 통과할 수 있는 능력, 호환성 있는 통신 어댑터.
- **호환성:** 서비스 및 끊임없는 메시징 교환에 대한 MQSeries 품질, 응용프로그램 변경 없이 기존 MQSeries 시스템과 통신할 수 있는 능력, MQSeries 및 MQSeries Everyplace 사이의 융통성 있는 메시지 교환 제어.
- **Footprint:** 최소한으로 구성된 장치 시스템에서 넉넉잡아 100KB 이하.

- **기능:** 동기 및 비동기 메시징 기능, 로컬 또는 리모트 큐에 보유된 메시지에 대한 액세스, 선택적 검색을 위해 메시지 내의 필드를 사용할 수 있는 능력, 큐에 대한 백업 매체의 선택적 제어.
- **규칙 지원:** 메시지를 보내는 시기, 통신 링크를 재시도하는 횟수, 너무 큰 메시지에 대해 할 일 또는 목표 큐가 가득 찼을 때의 작업(behavior) 등과 같이, 규칙을 통한 작업의 여러 측면에 대한 제어.
- **보안:** 보안, 인증 및 비거부에 대한 완전한 지원, 메시지 레벨 및 큐 레벨 보안, 보안 공격으로부터 메시징 시스템 보호, 산업 표준 알고리즘의 사용으로 호환성 있는 보안, 운영 체제 사용자 보증으로 통합할 수 있는 능력, 국제 보안 요구사항을 준수할 수 있는 기능, 메시지가 국경을 넘어갈 때 보안 지원이 변경될 수 있도록 허용.

제5장 제품 개념

소개

MQSeries Everyplace 프로그래밍 모델의 기본 요소는 메시지, 큐 및 큐 관리자입니다. MQSeries Everyplace 메시지는 응용프로그램에서 정의한 내용을 포함하는 오브젝트입니다. 메시지를 저장하면 큐에 보유하고, MQSeries Everyplace 네트워크를 통해 이동할 수 있습니다. 메시지는 목표 큐 관리자와 큐 이름 쌍을 지정하여 목표 큐로 주소가 지정됩니다. 응용프로그램은 넣기(put) 조작을 통해 메시지를 큐에 위치시키고, 대개 가져오기(get) 조작을 통해 메시지를 검색합니다. 큐는 로컬 또는 리모트 큐이며, 큐 관리자에 의해 관리됩니다. 장치 및 게이트웨이들 다 구성 데이터를 레지스트리에 저장합니다.

장치상의 응용프로그램은 장치에서 사용할 수 있는 API나 기능 중 일부 또는 모두를 사용할 수 있어서, MQSeries Everyplace 프로그래밍 인터페이스로 제한되지 않습니다. MQSeries Everyplace 장치는 동적 채널을 통해 다른 MQSeries Everyplace 장치나 MQSeries Everyplace 게이트웨이에 연결될 수도 있습니다.

게이트웨이의 응용프로그램은 또한 MQSeries Everyplace 프로그래밍 인터페이스뿐만 아니라, 게이트웨이에서 사용 가능한 API나 기능의 일부 또는 모두를 사용할 수 있습니다. 동적 채널을 통해, 게이트웨이는 다른 게이트웨이나 MQSeries Everyplace 장치에 연결될 수 있습니다. MQSeries 클라이언트 채널을 통해, 게이트웨이는 하나 이상의 MQSeries 서버에 연결될 수 있습니다. 그러나, 다른 MQSeries Everyplace 게이트웨이에는 연결될 수 없습니다. MQSeries가 전적으로 선택적이기는 하지만, MQSeries Everyplace와 MQSeries는 단일 기계에 공존할 수 있습니다.

전기능 장치 및 게이트웨이의 기능은 다음을 제외하고는 같습니다.

- 게이트웨이는 여러 개의 동시 인바운드 요청(다른 장치 또는 게이트웨이로부터의)을 처리할 수 있습니다.
- 게이트웨이는 동시에 여러 MQSeries 서버에 인터페이스로 접속할 수 있습니다.

동적 채널은 다음과 같은 네트워크 연결을 지원합니다.

- 전화 접속 연결
- 영구 연결(예를 들어, 일반 LAN, 전용 회선, 적외선 또는 무선 LAN)

통신 프로토콜은 일련의 어댑터로 구현됩니다(지원되는 프로토콜마다 하나씩). 이로써, 필요할 때 새로운 프로토콜을 추가할 수 있으며, 주어진 환경에서 메모리 footprint를 특정 구성에 맞도록 조정할 수 있습니다.

큐는 다른 어댑터 세트를 통해 스토리지 매체에 개별적으로 맵핑됩니다. 그러므로, 큐는 선택된 어댑터에 따라 파일 시스템이나 메모리에 저장됩니다.

MQSeries Everyplace 프로그래밍 인터페이스는 큐 위치에 종속되지 않고 응용프로그램을 작성할 수 있도록 설계되었습니다. 그러므로, 로컬 큐에 액세스하도록 설계된 프로그램은 리모트 큐 관리자로부터의 변경 없이 실행될 수 있어야 합니다(보안 고려사항을 강제로 만족시키고 특정 MQSeries Everyplace 조작이 리모트 큐에 대해 지원되지 않음을 승인한 상태에서). 이러한 독립성에는 모든 관리 기능의 사용이 포함됩니다.

메시지 오브젝트

MQSeries Everyplace 메시지 오브젝트는 기본적으로 MQSeries에서 지원되는 메시지와 다릅니다. MQSeries에서, 메시지는 바이트 배열로, 메시지 헤더와 메시지 본문으로 나뉩니다. 메시지 헤더는 MQSeries에 의해 인식되며, 큐에 대한 응답 ID, 큐 관리자에 대한 응답, 메시지 ID 및 상관 ID와 같은 필수 불가결한 정보를 포함하고 있습니다. 메시지 본문은 인식되지 않습니다.

반대로, MQSeries Everyplace의 메시지는 필드 오브젝트로도 알려진 MQSeries Everyplace 오브젝트에서 계승된 메시지 오브젝트입니다. 여기서 메시지는 헤더나 메시지 본문 개념이 없는 진정한 의미의 오브젝트입니다. 메시지 오브젝트의 실제 본질은 기본 필드 오브젝트가 인식될 경우에만 명백해집니다. MQSeries Everyplace에서 광범위하게 사용되는 이러한 필드 오브젝트는 필드의 누적으로, 하나의 필드는 이름, 데이터 유형 및 데이터 자체로 구성됩니다. 필드 이름은 무제한 길이의 ASCII 코드 문자열(예약 문자 제외)입니다.

필드 유형은 다음 중 하나입니다.

- **ASCII** 문자열 또는 ASCII 문자열의 동적 배열
- **Boolean** 값
- **바이트**, 고정 배열 또는 바이트 값의 동적 배열
- **Double Floating point**, 고정 배열 또는 double floating point 값의 동적 배열
- 필드 오브젝트나 필드 오브젝트의 동적 배열(필드 오브젝트의 네스팅이 지원됨)
- **Floating point** 값, 고정 배열 또는 floating point 값의 동적 배열
- **정수**(4 바이트), 고정 배열 또는 정수의 동적 배열
- **Long** 정수(8 바이트), 고정 배열 또는 long 정수의 동적 배열
- **Short** 정수(2 바이트), 고정 배열 또는 short 정수의 동적 배열
- **UNICODE** 문자열 또는 UNICODE 문자열의 동적 배열

필드 오브젝트는 유형이 프로그래밍 오브젝트 클래스 이름에 해당하는 유형을 가지고 있습니다. 이 오브젝트 클래스의 후손은 메시지 오브젝트로 응용프로그램에서 사용되며, 그 유형은 요구될 때(예를 들어, 메시지 오브젝트가 채널에서 이동된 후) 올바른 오브젝트 클래스를 인스턴스화하기 위해 MQSeries Everyplace에서 사용됩니다.

필드 오브젝트는 필드들을 나열하거나 존재를 확인하는 등 여러 메소드를 제공합니다. 마찬가지로, 필드 오브젝트의 동일성을 비교할 수 있습니다. 오브젝트들은 링크를 통한 오브젝트 전송용의 데이터를 제공하고, 전송 후에 오브젝트를 복원하는 등의 작업에 사용되는 바이트 배열로 해당되는 필드 항목을 덤프 및 복원하는 능력을 가지고 있습니다. 덤프 및 복원 방법은 필드 오브젝트가 다른 방식으로 순차화하도록(예를 들어, 전송 시 필드 내용에 대해 데이터베이스를 조화하여) 대체될 수 있습니다. 표3에 필드 오브젝트와 구성 필드에 대한 등록정보가 나와 있습니다.

표3. 필드 오브젝트 및 구성 필드 등록정보

등록정보	존재 여부	
	필드 오브젝트	필드
연관된 속성 오브젝트	선택적	
구성 필드	예	
숨겨짐		예
이름		예

표 3. 필드 오브젝트 및 구성 필드 등록정보 (계속)

등록정보	존재 여부	
	필드 오브젝트	필드
유형	예	예
값		예

필드의 숨겨진 등록정보는 비교 조작 시 필드를 무시할 수 있도록 합니다.

속성 오브젝트에는 인증, 암호화 및 압축을 수행하기 위한 메커니즘이 있으며, 필드 오브젝트와 연관될 수 있습니다.

- **인증:** 액세스를 제어합니다.
- **압축:** 스토리지 요구사항을 감소시킵니다(전송 및/또는 저장용).
- **암호화:** 오브젝트가 덤프될 때 내용을 보호합니다(복원 허용).

속성 오브젝트는 MQSeries Everyplace 보안 모델의 기초가 되며, 내용에 대한 선택적 액세스를 허용하고 다른 여러 용도 중에서도 백업 스토리지에서 데이터를 보호할 수 있게 합니다. 표4에는 속성 오브젝트들에 대한 등록정보가 나열되어 있습니다. 규칙 값은(존재할 경우) 어떤 조작을 허용할 지 제어합니다.

표 4. 속성 오브젝트 등록정보

등록정보	존재 여부
암호화기	선택적(일부 상황에서는 필수)
인증자	선택적
압축기	선택적
규칙	선택적
유형	선택적

메시지 오브젝트는 필드 오브젝트에서 도출되며 MQSeries Everyplace가 생성하는 UID(고유 식별자)를 포함합니다. 이 UID는 메시지 오브젝트를 고유하게 식별하며 다음으로 구성됩니다.

- 생성한 큐 관리자의 이름(오브젝트를 수신할 때 큐 관리자에 의해 추가됨). 이 이름은 전적으로 고유해야 합니다.
- 메시지 오브젝트가 작성된 시간(작성 시 추가됨)

메시지 오브젝트는 필드 오브젝트로서 계승한 등록정보 외에도 표5에 나열된 기본 등록정보를 가지고 있습니다.

표 5. 메시지 오브젝트 등록정보

등록정보	설명
Msg_OriginQMgr	메시지를 보낸 큐 관리자의 이름
Msg_Time	메시지 오브젝트가 응용프로그램에 의해 작성된 시간

이 두 등록정보가 메시지 오브젝트의 고유 식별자(UID)를 구성합니다.

메시지 정보 내용을 처리하기 위해 보통 다른 필드도 포함되지만, 다른 MQSeries Everyplace 큐 관리자에 보내질 메시지에서는 다른 어떤 정보도 필요하지 않습니다. 일반적으로 메시지는 기본 메시지 오브젝트 클래스의 후손이므로, 목적에 적합한 추가 필드를 수반합니다. 이러한 추가 필드들 중에서 번호는 큐 관리자에 응답과 같이 광범위한 응용프로그램에 공통됩니다. 이에 따라, MQSeries Everyplace는 그 필드들에 대한 지원 조치를 제공합니다.

표6에 지원되는 필드들이 나열되어 있습니다.

표 6. 준비된 메시지 오브젝트 필드

필드 이름	사용
Msg_CorrelID	응답을 원래 메시지와 연관시키기 위해 사용되는 바이트 문자열
Msg_ExpireTime	메시지 삭제 가능 시간(전달되지 않은 경우에도)
Msg_LockID	메시지 잠금을 해제하는 데 필요한 키
Msg_MsgID	원래 메시지와 상관을 위해 응용프로그램에서 사용됨
Msg_Priority	메시지의 우선순위
Msg_ReplyToQ	메시지 응답이 주소 지정되어야 하는 큐의 이름.
Msg_ReplyToQMgr	메시지 응답이 주소 지정되어야 하는 큐 관리자의 이름.
Msg_Resend	메시지가 이전 메시지의 재송신임을 나타냅니다.
Msg_Style	요청/응답 등에서 명령을 구별합니다.

모든 경우에, 1 바이트로 필드 이름을 처리해 주는 정의된 상수를 사용할 수 있습니다. 일부 필드의 경우, 더 광범위한 대비책이 있습니다. 예를 들어, 우선순위(존재할 경우)가 메시지 전송 순서에 영향을 주고, 상관 ID는 빠른 검색을 위해 해당 필드 값에 대해 큐의 색인화를 트리거하며, 만기 시간은 메시지의 만기를 트리거합니다.

메시지 오브젝트에는 정의된 여러 가지의 메소드가 있습니다. 예를 들어, 메시지 UID, 생성한 큐 관리자 이름 및 오브젝트 작성시간을 추출할 수 있는 기능이 있습니다. 다른 유용한 메소드들은 필드 오브젝트 클래스로부터 계승됩니다(예를 들어, 필드 값을 가져오고 놓기 위한 다양한 메소드). 특히 흥미로운 것은 *dump* 메소드로서, 이는 오브젝트 데이터를 바이트 문자열로 덤프하는 데 사용됩니다. MQSeries Everyplace는 메시지가 영구 스토리지에 저장되거나 동적 채널을 통해 전송될 때 이 메소드를 호출합니다. 다시 말하자면, 메시지 오브젝트 스스로 데이터 값의 외부 표시를 판별해야 하며, 이것은 여러 가지 방식으로 이용될 수 있습니다. 예를 들어, 오브젝트는 구성 필드들의 값을 간단히 덤프하거나 아니면 대신에 데이터베이스를 조회할 수 있습니다. 이를 보완하는 *restore* 메소드는 덤프된 형식으로 오브젝트를 다시 작성할 때 유사한 제어 가능성을 제공합니다. 메시지 오브젝트에 부가적인 속성 오브젝트가 있으면, 속성의 데이터 인코더, 암호화기 및 압축기가 덤프 시 호출됩니다. 마찬가지로, 디코너, 암호해독기 및 압축해제기가 복원 시 호출됩니다.

MQSeries Everyplace가 메시지 오브젝트를 이동시킬 때, 선로의 footprint를 줄이기 위해 연관된 클래스 파일은 이동시키지 않습니다. 따라서, 메시지 오브젝트가 인스턴스화되는 각 큐 관리자에서 적절한 메시지 클래스를 사용할 수 있어야 합니다.

기본 메시지 오브젝트 *dump* 메소드는 효율적인 메시지 저장과 전송을 위해 생성된 바이트 문자열의 크기를 최소화하도록 최적화되었습니다.

덤프 데이터 형식

기본 덤프 데이터 형식은 다음과 같이 필드를 코드화합니다.

```
{Length Identifier Fence {Data}} {Length Identifier Fence {Data}} { ...}
```

설명:

- *Data*: 데이터 값. 정수는 선행 0과 F를 제거하고 압축됩니다. Boolean 값에는 연관된 데이터 바이트가 없습니다.
- *Fence*: 식별자와 선택적 데이터 항목 사이의 경계를 구분하는 특수 바이트. 이 바이트는 또한 데이터 항목 유형을 나타냅니다.

- *Identifier*: 다양한 길이의 ASCII 문자열 바이트에 필드 이름을 보유하며, 끝 바이트로 종료됩니다.
- *Length*: 데이터 필드의 길이를 나타냅니다. 1과 4 사이의 바이트 수가 사용됩니다. 첫번째 바이트는 길이 필드의 길이를 나타내기 위해 처음 2 비트를 예약해 둡니다. 0 - 1,073,741,823 범위의 길이가 지원됩니다.

이렇게 하면 데이터 스트림 압축율이 높아집니다. 데이터를 압축하면 추가 절약이 가능합니다. 이전 바이트 스트림에 대한 XOR 압축은 좋은 결과가 예상되지만, 이 필드들의 변하기 쉬운 성질과 필드 순서의 변경 가능성 때문에 단순한 XOR로 항상 원하는 효과를 얻는 것은 아닙니다. MQSeries Everyplace에는 필드별로 작동되는 지능형 XOR이 포함되어 있습니다. 이로서 압축이 훨씬 향상될 수 있습니다.

큐

큐는 보통 응용프로그램에 의한 제어를 기다리는 중인 메시지 오브젝트를 보유하는 데 사용됩니다. 메시지처럼, 큐도 필드 오브젝트로부터 도출됩니다. 응1족?? 직접 큐 오브젝트에 액세스하는 것은 허용되지 않습니다.³ 그 대신, 큐 관리자가 응용프로그램과 큐 사이의 중개 역할을 합니다. 큐는 이름으로 식별되며 그 이름은 무제한 길이의 ASCII 문자열이 될 수 있지만⁴ 특정 큐 관리자 내에서 고유해야 합니다.

MQSeries Everyplace는 여러 가지의 다른 큐 유형을 지원합니다.

로컬 큐

로컬 큐는 응용프로그램이 메시지를 안전한 보안 방식으로 저장하는 데 사용됩니다. 로컬 큐에는 어댑터 클래스(보통, 디스크 어댑터 클래스)를 통해 액세스되는 메시지 저장소가 있습니다. 그러나, 메모리 어댑터 클래스는 (시스템이 고장나면 메시지가 유실되지만) 빠른 액세스를 위해 메모리에 메시지 저장소를 보유하는 MQSeries Everyplace에서 제공됩니다. 적절한 어댑터 메시지를 작성하면 큐 기준으로 어디에든지 저장할 수 있습니다(예를 들어, 관계형 데이터베이스나 쓰기 가능한 CD 등). 로컬 큐는

3. 직접 액세스는 큐 규칙 내에서만 허용됩니다.

4. 상호 작동성을 위해, 최대 이름 길이는 48자와 MQSeries 이름 지정 제한사항을 준수하는 것이 좋습니다. 길이는 사용하는 파일 시스템에서 제한될 수도 있습니다.

네트워크에 연결된 온라인이나 네트워크에 연결되지 않은 오프라인에서 사용할 수 있습니다. 액세스 및 보안은 큐에서 소유하며, 네트워크에 연결되었을 때 리모트 큐 관리자가 사용 권한을 부여하여 다른 사용자가 메시지를 큐에 보내거나 큐로부터 받을 수 있게 됩니다. 로컬 큐 액세스는 항상 동기식입니다.

리모트 큐

리모트 큐는 지역 환경에 상주하지 않고, 소유 큐(owning queue) 관리자와 실제 큐를 식별하는 정의가 지역적으로 존재합니다. 리모트 큐는 동기 또는 비동기식으로 액세스할 수 있습니다. 지역적으로 보유된 리모트 큐의 정의가 있으면, 액세스 모드는 이 정의를 기초로 합니다. 그렇지 않으면, 특성이 전개되는 것처럼 큐가 전개되고 액세스 모드가 강제로 동기화됩니다.

동기 큐는 소유 큐(owning queue) 관리자의 경로가 있는 네트워크에 연결되었을 때만 액세스할 수 있는 큐입니다. 네트워크가 구축되지 않으면, 넣기, 가져오기 및 찾아보기와 같은 조작(29 페이지의 표10 참조)으로 예외가 발생합니다. 소유 큐(owning queue)는 큐에 액세스하는 데 필요한 액세스 권한과 보안 요구사항을 제어합니다. 메시지를 송수신할 때, 오류나 재시도를 처리하는 것은 응용프로그램이며, 이 경우 MQSeries Everyplace는 1회 보증 전달을 책임지지 않습니다.

비동기 큐는 메시지를 넣을 수는 있지만 검색할 수는 없는 큐입니다. 네트워크에 연결되면, 메시지는 소유 큐(owning queue) 관리자와 큐에 전송됩니다. 그렇지 않으면 메시지는 네트워크에 연결되어 전송될 때까지 지역적으로 저장됩니다. 이렇게 하여, 응용프로그램은 장치가 오프라인일 때 큐에서 작동될 수 있습니다. 그러나, 이러한 유형의 큐는 메시지 저장소를 가지고 있어야 임시로 메시지를 저장할 수 있습니다.

저장 및 포워드 큐

이러한 유형의 큐는 그 다음(반드시 소유 큐 관리자는 아님) 큐 관리자로 전송될 수 있을 때까지 메시지를 저장합니다. 이 큐는 보통 게이트웨이에서 정의되며(반드시 그런 것은 아님) 장치는 네트워크에 연결될 때 해당되는 메시지를 수집합니다. 저장 및 포워드 큐는 여러 목표 큐 관리자에 대한 메시지를 보유할 수 있으며, 또는 목표 큐 관리자마다 하나의 저장 및

포워드 큐가 있을 수 있습니다. 송신자가 연결해제된 수신자에게 메시지를 보내려고 할 때 그 송신자는 계속 그 메시지를 수신자의 큐 관리자/큐로 주소지정합니다. 중간 서버는 수신자가 연결되지 않았음을 감지하고 그 메시지를 지역 메시지 저장소에 저장합니다. 송신 응용프로그램은 목표 큐의 연결 여부에 관계없이 메시지를 보내는데 어떤 변경도 할 필요가 없습니다.

홈 서버 큐

이러한 유형의 큐는 보통 장치(때때로 연결되는)에 상주하며 홈 서버로 알려진 큐 관리자에 있는 저장 및 포워드 큐를 향합니다. 홈 서버 큐는 장치가 네트워크에 연결될 때마다 홈 서버로부터 메시지를 가져옵니다. 큐가 서버로부터 메시지를 가져왔으면, 이를 `putMessage` 및 `confirmputMessage` 메소드 호출(28 페이지의 『큐 관리자 조작』 참조)을 사용하여 로컬 큐 관리자에 제공합니다. 그러면, 올바른 로컬 큐에 메시지를 배치하는 것은 큐 관리자의 책임이 됩니다. 서버로부터 메시지를 가져오는 `pull` 메소드는 서버가 메시지를 넣어주는 것보다 네트워크를 통한 흐름 측면에서 더 효율적일 수 있습니다. 이는 홈 서버 큐는 첫번째 메시지의 도착을 다음 메시지 요청으로 받아들이는 반면 서버가 넣어주는 방식은 메시지 송신을 위한 요청/응답과 그 흐름을 확인하기 위한 두번째 요청/응답을 요구하기 때문입니다. 홈 서버 큐에는 보통 네트워크가 연결되어 있는 동안 보류 중인 메시지를 확인하도록 하는 폴 간격이 설정되어 있습니다. 폴 간격은 관리 구성 옵션입니다.

관리 큐

이 유형의 큐는 MQSeries Everyplace 관리 메시지를 수신합니다. 선택적 관리 메시지 응답 큐는 MQSeries Everyplace 시스템이 보낸 관리 메시지에 대한 응답을 수신하기 위해서도 사용될 수 있습니다. 관리 큐는 관리 수행 방법은 알지 못하며, 관리 세부사항을 캡슐화하는 메시지를 처리합니다.

MQSeries 브릿지 큐

이것은 게이트웨이에서 정의된 리모트 큐와 MQSeries 큐 관리자에서의 목표 큐에 대한 특수화된 양식입니다. 이 양식의 큐는 MQSeries Everyplace와 MQSeries 환경 사이의 경로를 제공합니다. 변환기는 필요한 데이터나 메시지의 재형식화를 위해 사용됩니다. 아주 기본적인 변환기

는 MQSeries Everyplace에서 제공됩니다. 프로그래머는 이 변환기를 조정하여 자신의 요구사항에 맞도록 합니다.

MQSeries Everyplace는 데이터를 안전하게 큐에 저장하여, 메시지가 단지 운영 체제에 의해 버퍼링만 되는 것이 아니라 실제로 매체에 기록되도록 합니다. 그러나, MQSeries Everyplace는 메시지 및 큐에 대한 변경사항을 개별적으로 로그하지 않습니다. 매체 장애로부터 복구가 필요하다면, RAID 디스크 시스템 사용과 같은 하드웨어 솔루션이 전개되어야 합니다. 다른 방법으로는 큐가 특정의 데이터베이스 서브시스템과 같은 복구 가능한 스토리지에 맵핑되어야 합니다.

MQSeries Everyplace는 큐 관리자가 큐를 정의하도록 요구하지 않습니다. 그러나, 필요할 경우 네 가지 시스템 큐에 대한 대비가 있습니다.

- **AdminQ:** 관리 메시지 수신을 위해 필요합니다.
- **AdminReplyQ:** 관리 메시지에 대한 응답을 수신하기 위해 선택적으로 사용됩니다.
- **DeadLetterQ:** 달리 전달될 수 없는 메시지를 저장하기 위해 사용됩니다.
- **SYSTEM.DEFAULT.LOCAL.QUEUE:** MQSeries 서버에서 필수 시스템 큐와 공통 이름을 공유하는 큐.

큐 등록정보는 표7에 나와 있습니다. 그러나, 표시된 모든 등록정보가 모든 큐 유형에 적용되는 것은 아닙니다.

표 7. 큐 등록정보

등록정보	설명
Admin_Class	큐 클래스
Admin_Name	Ascii 큐 이름
Queue_Active	큐가 활성임을 나타냅니다.
Queue_AttRule	보안 조작용을 제어하는 규칙 클래스
Queue_Authenticator	인증자 클래스
Queue_BridgeName	MQ 브릿지 이름 소유
Queue_ClientConnection	클라이언트 연결 이름
Queue_CreationDate	큐가 작성된 날짜
Queue_Compressor	압축기 클래스
Queue_Cryptor	암호화기클래스

표 7. 큐 등록정보 (계속)

등록정보	설명
Queue_CurrentSize	큐의 메시지 수
Queue_Description	유니코드 설명
Queue_Expiry	메시지의 만기 시간
Queue_FileDesc	큐가 저장되는 위치
Queue_MaxMsgSize	큐에서 허용되는 최대 메시지 길이
Queue_MaxQSize	허용되는 최대 메시지 수
Queue_Mode	동기 또는 비동기
Queue_MQMgr	MQSeries 큐 관리자 프록시
Queue_Priority	메시지에 사용될 우선순위(메시지 값에 의해 대체되지 않은 경우)
Queue_QAliasNameList	큐에 대한 대체 이름
Queue_QMgrName	실제 큐를 소유하는 큐 관리자
Queue_RemoteQName	원격 MQSeries 필드 이름
Queue_Rule	큐 조작에 대한 규칙 클래스
Queue_TargetRegistry	목표 레지스트리 유형
Queue_Transporter	전송기 클래스
Queue_Transformer	변환기 클래스

관리 기능은 큐를 작성하고 삭제하며, 큐의 등록정보를 조회하거나 수정하는 데 사용됩니다.

큐는 메시지 저장소로 사용되도록 제한되지 않습니다. 프로세스 제어 응용프로그램 시나리오에서 서브클래스화된 큐를 사용할 수 있습니다. 예를 들어, 큐 오브젝트는 직접 밸브를 제어할 수 있습니다. 올바른 유형의 메시지는 밸브를 열어 흐름 수량을 변경하는 등을 실행시킵니다. 응용프로그램이 큐에서 메시지를 가져와 조치를 수행하는 것이 아니라, 큐 오브젝트 자체가 조치를 제어합니다. 다른 큐는 스프레드시트 갱신이나 텍스트의 스페이스 변환 등을 수행할 수 있습니다. 이 기술의 장점은 큐의 보안 측면이 계속 효율적으로 유지되면서 메시징을 보증한다는 것입니다. 그래서, MQSeries Everyplace는 계속해서 메시지 1회 보증 전달을 제공하고, 권한 부여된 메시지 송신자만 그러한 메시지를 전송할 수 있도록 연관된 인증자와 암호화기가 보증하며, 그 내용은 전송 시 확실하게 보안됩니다. 어떤 응용프로그램도 큐에 액세스할 수 없으므로 어떤 것도 요구되지 않습니다.

큐 관리자

MQSeries Everyplace 큐 관리자는 메시지 및 큐에 대한 응용프로그램 액세스를 제공하고 채널을 제어합니다. MQSeries Everyplace 버전 1.0에서는 단 한번에 하나의 큐 관리자만 하나의 Java 가상 기계(JVM)에서 활성화될 수 있습니다. 기계에 JVM이 여러 개이면, JVM과 같은 수의 큐 관리자가 가능합니다. 큐 관리자는 이름에 의해 식별되는데, 이 이름은 전적으로 고유해야 하며⁵ 무제한 길이의 ASCII 문자열이어야 합니다.⁶ 큐 관리자는 로컬 큐 처리를 사용하거나 사용하지 않고 구성될 수 있습니다. 모든 큐 관리자는 동기 메시징 조작을 지원합니다. 로컬 큐 처리를 사용하는 큐 관리자는 비동기 메시징 전달도 지원합니다.

비동기 및 동기 메시지 전달의 특성과 결과는 아주 다릅니다.

비동기 메시지 전달⁷

비동기 메시지 전달을 할 때, 응용프로그램은 리모트 큐로 전달하기 위해 메시지를 MQSeries Everyplace로 보냅니다. 넣기 조작 후 즉시 응용프로그램으로의 리턴이 발생합니다. MQSeries Everyplace는 전달될 수 있을 때까지 임시로 지역에 메시지를 보유하고 있습니다. 전달은 전달 책임을 맡고 있는 MQSeries Everyplace에서 행해질 수 있습니다. 이러한 조작 모드는 *1회 보증 전달*을 제공합니다. 38 페이지의 『비동기 메시지 전달』에서 자세한 내용을 살펴보십시오.

동기 메시지 전달

동기 메시징은 다음을 주소지정하기 위해 사용됩니다.

- MQSeries Everyplace 네트워크를 거쳐 경로지정되는 MQSeries Everyplace 큐 관리자상의 목표 큐.
- MQSeries Everyplace 게이트웨이에 직접 접속된 MQSeries 큐 관리자상의 목표 큐

5. 이 제한사항은 MQSeries Everyplace 또는 MQSeries에서 강제적은 아니지만, 중복되는 큐 관리자 이름이 있으면 메시지가 잘못된 큐 관리자로 전달될 수 있습니다.

6. 상호 작동성을 위해, 이름 길이를 최대 48자로 제한하는 것을 포함하여, MQSeries 큐 관리자 이름 규칙을 준수하는 것이 좋습니다. 길이는 사용하는 파일 시스템에서 제한될 수도 있습니다.

7. MQSeries Everyplace는 MQSeries에서 제공되는 영구적 및 비영구적 모드를 구별하지 않으나, 영구적 모드만 지원됩니다. 그러나, 필요하면 큐 백업 스토리지를 선택하여 성능과 복구 사이에 조정할 수 있습니다.

- 간접적으로 접속된 MQSeries 서버상의 목표 큐

동기 메시지를 사용할 때 응용프로그램은 전달할 메시지를 MQSeries Everyplace에 넣습니다. MQSeries Everyplace는 동기식으로 목표 리모트 큐에 접속하여 메시지를 위치시킵니다. 전달 후, MQSeries Everyplace는 응용프로그램으로 돌아갑니다.

리모트 큐 관리자와의 접속에는 중간 매개 장치나 게이트웨이를 통한 MQSeries Everyplace 경로지정이 포함될 수도 있습니다. 39 페이지의 『동기 메시지 전달』에서 자세한 내용을 살펴보십시오.

그러므로, 비동기 메시지 전달에서 로컬 응용프로그램은 메시지를 MQSeries Everyplace에 제공하고, 그 로컬 큐 관리자로부터의 전달은 MQSeries Everyplace의 책임입니다. 이는 네트워크나 수신 응용프로그램이 사용 가능할 필요가 없음을 의미합니다. 실제 전달 시간은 송신 응용프로그램에 알려지지 않습니다. 동기 메시지 전달에서는 네트워크가 실행중이어야 하지만, 송신 응용프로그램은 수신 응용프로그램의 큐에 메시지가 전달되었음을 압니다. 수신 응용프로그램은 비동기 또는 동기 방식에서 사용할 필요가 없습니다.

로컬 큐 관리자에는 큐의 지역 관리를 반영하는 등록정보가 있습니다. 또한, 접속해야 하는 리모트 큐 관리자마다 연결 정의를 가지고 있어야 합니다. 그래서, 간혹 연결 정의를 리모트 큐 관리자 정의라고 합니다. 이 정의에는 큐 관리자 사이의 직접 통신에 필요한 모든 정보(네트워크 주소와 같은)가 포함되거나, 단순히 통신이 간접적이고 중간 큐 관리자를 거친다는 것만 표시할 수도 있습니다. 나중 경우, 다음 홉(hop) 큐 관리자의 이름만 필요합니다.

큐 관리자 등록정보가 표8 및 28 페이지의 표9에 나와 있습니다.

표 8. 로컬 큐 관리자 등록정보

등록정보	설명
QMgr_ChnlAttrRules	채널 속성 규칙
QMgr_ChnlTimeout	채널 시간 종료
Admin_Class	큐 관리자 클래스
QMgr_Description	유니코드 설명
Admin_Name	큐 관리자 이름
QMgr_Rules	큐 관리자 조작에 대한 규칙 클래스

표 9. 연결(리모트 큐 관리자) 등록정보

등록정보	설명
Con_Adapter	어댑터 파일 설명자
Con_AdapterOptions	어댑터 옵션(실행기록 등)
Con_AdapterParm	어댑터에서 사용될 ASCII 데이터(서블릿 이름 등)
Con_Aliases	큐 관리자에 대한 대체 이름
Con_Channel	이 연결에서 사용해야 하는 채널 유형
Con_Description	유니코드 설명
Queue_QMgrName	정의의 소유자
Admin_Name	큐 관리자 이름

연결 정의에서 여러 어댑터가 지원됩니다.

큐 관리자 조작

큐 관리자는 메시징 조작과 선택적으로 메시지 큐를 지원합니다. 응용프로그램은 큐 관리자에서 수행되는 조작을 통해 메시지에 액세스합니다. 필터를 지정하지 않으면, 큐에 있는 첫번째 사용 가능한 메시지가 검색됩니다. 필터는 동일성을 위해 일치시켜 보는 필드 오브젝트로, 메시지의 필드들은 모두 선택적 검색에 사용할 수 있습니다. 모든 메시지 송신 및 검색 조작처럼 가져오기(get) 조작은 메시지 인코딩과 디코딩에 사용되는 속성 오브젝트를 선택적으로 부여받을 수 있습니다.

MQSeries와 마찬가지로, MQSeries Everyplace에서 가져오기(get)는 대개 소멸성 조작입니다. MQSeries Everyplace와 응용프로그램 사이에 보증 메시징이 필요할 경우, 가져오기(get) 다음 confirm 메소드의 순서로 해야 합니다. 먼저, 가져오기는 확인 ID(응용프로그램에 의해 선택되는 값)와 함께 발행됩니다. 이 조작은 메시지를 가져오지만, 즉시 삭제한다기보다는 큐에서 숨기는 것입니다. 원래 메시지 UID를 지정하는 후속 확인 조작은 응용프로그램에 대해 가져오기(get)가 성공하였고, 그래서 그 메시지가 삭제되었음을 나타냅니다. 가져오기에 실패하면 메시지는 복구될 수 있습니다. 넣기(put) 조작도 유사하게 작동됩니다.

UID를 지정하면, 메시지는 큐에서 검색 없이 삭제될 수 있습니다.

비과외적인 읽기가 필요할 경우, (선택적으로 필터 제어 하에) 큐에서 메시지를 찾아볼 수 있습니다. 찾아보기는 필터와 일치하는 모든 메시지 오브젝트를 검색하기

만 하고, 그 메시지들을 큐에 그대로 둡니다. 잠금 상태에서의 찾아보기도 지원됩니다. 여기에는 큐에서 일치하는 메시지를 잠그는 추가 기능이 있습니다. 메시지는 필터를 통해 식별된 그룹으로, 또는 개별적으로 잠글 수 있으며, 잠금 조작은 잠금 ID를 리턴합니다. 잠긴 메시지는 잠금 ID가 제공될 경우에만 가져오거나 삭제할 수 있습니다. 찾아보기에는 전체 메시지 또는 UID만 리턴되도록 하는 옵션이 있습니다.

응용프로그램은 지정된 시간 동안 큐에 메시지가 도착하기를 기다립니다. 선택적으로, 필터는 관심이 있는 것만 식별하는 데 사용될 수 있으며, 확인 ID도 지정할 수 있습니다. 다른 방법으로는 역시 선택적으로 필터를 사용하여 응용프로그램이 MQSeries Everyplace 메시지 이벤트를 연결대기하는 것이 있습니다. 메시지가 큐에 도착하면 리스너에게 통지됩니다.

큐는 표10에 표시된 것처럼 메시징 조작에 사용될 수 있습니다.

표 10. MQSeries Everyplace 큐에서의 메시징 조작

	로컬 큐	리모트 큐 ¹	
		동기	비동기
찾아보기(±잠금, ±필터)	예	예	
삭제	예	예	
가져오기(±필터)	예	예	
청취(±필터)	예		
넣기	예	예	예
대기(±필터)	예	예	

주:

1. 동기식 원격 대기 조작은 원격 큐의 폴(pole)을 통해 구현되므로, 실제 대기 시간은 폴 시간의 몇 배입니다.
2. ¹MQSeries Everyplace 버전 1.0과 함께 제공되는 MQSeries Everyplace MQSeries 브릿지만 '넣기(put)' 조작을 지원합니다.

큐 관리자는 초기설정 후에 즉시 선택적으로 응용프로그램(클래스)을 로드할 수 있습니다. 마찬가지로, 종료 시에도 응용프로그램을 종료할 수 있습니다. 큐 관리자는 상태나 오류를 반영하기 위해 이벤트를 발생시킵니다. 기본적으로, 이 이벤트들은 이벤트 로그에 표시됩니다.

관리

관리는 큐 및 연결과 같은 MQSeries Everyplace 자원을 구성하고 관리하는 기능을 제공합니다. 메시지 관련 기능은 응용프로그램의 책임으로 간주됩니다. 관리는 관리 메시지의 생성 및 수신을 처리하는 인터페이스를 통해 사용 가능하며, 지역 및 원격 관리를 동일 방식으로 처리하도록 설계되었습니다. 요청은 목표 큐 관리자의 관리 큐에 송신되고 응답은 필요할 경우에 수신될 수 있습니다. 지역 또는 원격 MQSeries Everyplace 응용프로그램은 helper 메소드를 통해 직접 또는 간접적으로 관리 메시지를 처리할 수 있습니다. 관리 메시지는 MQSeries Everyplace Explorer⁸를 통해 간접적으로 생성될 수도 있습니다. 이것은 시스템 관리를 위한 그래픽 사용자 인터페이스를 제공하는 관리 도구입니다.

관리 큐는 개별 자원의 관리를 수행하는 방법을 알지 못합니다. 이러한 정보는 각 자원과 해당되는 관리 메시지에 캡슐화되어 있습니다.

관리 메시지

관리 메시지는 기본 MQSeries Everyplace 메시지 오브젝트를 확장합니다. 표 11에 MQSeries Everyplace 자원 관리를 위해 제공되는 메시지 클래스가 나열되어 있습니다. 이 기본 관리 메시지들은 다른 오브젝트의 관리를 위해 서브클래스화될 수 있습니다. 예를 들어, MQQueueAdminMsg의 서브클래스를 사용하여 다른 유형의 큐를 관리할 수 있습니다. MQSeries에 대한 MQSeries Everyplace 브릿지는 이런 식으로 MQAdminMsg의 서브클래스를 사용합니다.

표 11. 관리 메시지 클래스

관리 메시지 클래스	사용
MQAdminMsg	모든 관리 메시지의 기본으로 사용되는 추상 클래스
MQQueueManagerAdminMsg	큐 관리자의 관리
MQQueueAdminMsg	로컬 큐의 관리
MQRemoteQueueAdminMsg	리모트 큐의 관리
MQAdminQueueAdminMsg	관리 큐의 관리
MQHomeServerQueueAdminMsg	홈 서버 큐의 관리

8. MQSeries Everyplace Explorer는 버전 1.0에 포함되어 있지만 WWW(<http://www-4.ibm.com/software/ts/MQSeries/>)의 MQSeries 소프트웨어 다운로드 사이트로부터 사용 가능합니다.

표 11. 관리 메시지 클래스 (계속)

관리 메시지 클래스	사용
MQeStoreAndForwardQueueAdminMsg	저장 및 포워드 큐의 관리
MQeConnectionAdminMsg	큐 관리자 사이의 연결 관리
MQeClientConnectionAdminMsg	MQS에 연결하는 데 사용되는 브릿지 클라이언트 연결 오브젝트의 관리
MQeListenerAdminMsg	MQS로부터 메시지를 수집하기 위해 사용되는 브릿지 전송 큐 리스너 오브젝트의 관리
MQeBridgeAdminMsg	MQS에 대한 브릿지의 관리
MQeMQBridgesAdminMsg	MQ 브릿지 목록의 관리
MQeMQMgrProxyAdminMsg	MQSeries 큐 관리자의 브릿지 표시 관리
MQeMQBridgeQueueAdminMsg	MQ 브릿지 큐의 관리

관리 메시지의 구조는 그것의 고유한 클래스에 따라 결정됩니다. 즉, 관리하는 자원의 성질과, 그 자원에서 수행될 조작의 세부사항에 따라 달라집니다. 그러나, 일반적으로 관리 메시지는 표12에서처럼 구조화됩니다.

표 12. 관리 메시지의 일반 구조

레벨 1 필드	레벨 2 이하의 필드	사용
Admin_Action		작성, 삭제, 조회...
Admin_Errors		필드 오브젝트 상위
	여러 필드	오류별 상세 정보
Admin_MaxAttempts		관리 조치가 시도되어야 하는 최대 횟수
Admin_Parameters		필드 오브젝트 상위
	자원	관리될 자원의 이름
	여러 필드	메시지 클래스 및 조치에 고유한 자세한 매개변수 데이터
Admin_Reason		실패 원인을 나타내는 텍스트 메시지
Msg_ReplyToQ		응답을 송신할 큐의 이름
Msg_ReplyToQMgr		응답을 송신할 큐 관리자의 이름
Admin_RC		결과를 나타내는 숫자 리턴 코드
Msg_Style		명령 또는 요청/응답
Admin_TargetQMgr		목표 자원을 소유하는 큐 관리자의 이름

세 가지 유형의 관리 메시지가 지원됩니다. 응답을 필요로 하지 않는 관리 조치를 나타내는 명령(데이터그램), 응답을 필요로 하는 요청, 그리고 응답 자체입니다. 응답은 원래 메시지 사본에서 구성되므로, 수신자가 사용하도록 송신자가 필드를 추가할 수 있습니다.

기본적인 관리 메시지 지원 외에도, 메시지 구성과 응답의 해석을 캡슐화하는 helper 클래스도 가장 공통적인 관리 조작에 대해 제공됩니다. 이 클래스들은 간단한 관리 도구를 구축하는 작업에 유용한 사용자 대화상자를 선택적으로 제공할 수 있습니다.⁹

선택적 관리

관리에 대한 액세스는 관리 큐의 인증자를 통해 제어할 수 있습니다. 지역 응용프로그램에 있어서, 제공된 인증자는 이 응용프로그램 모두가 동일한 지역 사용자를 나타내는 것으로 간주하고, 그 응용프로그램 모두에 대해 관리를 허용하거나 허용하지 않습니다. 원격 관리 응용프로그램은 관리 메시지가 흐르기 전에 채널에서 인증자를 호출하여 제어됩니다. 이렇게, 다른 원격 사용자를 구별할 수 있으므로 안전하게 작동 가능 또는 작동 불가능하게 됩니다. 모든 경우에서 임의 사용자에 대해 관리는 전체적으로 작동가능하거나 작동 불가능합니다. 더 세분화된 관리 제어 레벨이 필요할 경우, 예를 들어 어떤 관리 사용자에게 일부 큐에 대한 액세스만 제공될 경우, 추가 프로그래밍이 필요합니다. 더 정교한 인증자는 사용자 ID와 연관되는 사용권한을 추적할 수 있으며, 관리 메시지는 그 뒤로 이 권한을 기초로 처리될 수 있습니다(보안 관련 절 참조). 유사한 방식으로 조치를 허용하거나 허용하지 않기 위해 큐와 연관되는 규칙을 이용할 수도 있습니다(49 페이지의 『규칙』 참조).

모니터링 및 관련 조치

관리는 종종 오브젝트 작성 및 수정 이상의 것에도 관련됩니다. 예를 들어, 시스템 상태 모니터와 오류 상태 처리 등에 관련됩니다(조작원에게 큐가 거의 가득 찼음을 알리거나, 목표 큐에 비해 너무 큰 메시지가 도착할 때 적절한 조치를 취하는

9. 이 클래스들은 버전 1.0에 포함되어 있지 않지만, WWW(<http://www-4.ibm.com/software/ts/MQSeries/>)의 MQSeries 소프트웨어 다운로드 사이트에서 사용할 수 있습니다.

것 등). 이러한 측면들은 오브젝트가 상태를 크게 변경하거나 특정 유형의 오류 상태가 발생할 때 호출되는 클래스인 규칙을 통해 MQSeries Everyplace에서 처리됩니다. 기본 규칙 클래스 세트는 MQSeries Everyplace에서 제공되지만, 보통 이 클래스들은 사용자 정의 클래스로 대체됩니다(49 페이지의 『규칙』 참조).

동적 채널

MQSeries Everyplace는 동적 채널이라고 알려진 논리 링크를 통해 장치나 게이트웨이 큐 관리자 사이에 통신합니다. 양방향 흐름을 지원하며 필요에 따라 큐 관리자에 의해 구축됩니다. 비동기 및 동기 메시징 모두 같은 채널을 사용하며 사용되는 프로토콜은 MQSeries Everyplace에 고유합니다. 반대로 MQSeries는 보통 동기 트래픽에 클라이언트 채널을 사용하고 양방향 비동기 메시징에 메시지 채널 쌍을 사용합니다. MQSeries 클러스터 메시지 채널은 MQSeries Everyplace 동적 채널과 유사한 특성을 일부 가지고 있지만, 차이점이 몇 가지 있습니다.

동적 채널은 두 개의 큐 관리자 사이의 논리적 연결로, 데이터의 송수신을 위해 구축됩니다. 같은 파티 사이에도 여러 개의 동시 채널이 존재할 수 있으며, 사용되는 인증, 암호화, 압축 및 전송 프로토콜과 같은 특성을 가지고 있습니다. 이러한 특성은 호환성이 있으며(채널마다 다양한 버전 사용 가능) 결국 각 채널은 다음과 같은 고유한 서비스 품질 속성을 갖습니다.

- **인증자:** 사용자나 채널 인증을 수행할 수 있는 인증자 오브젝트 또는 널(null)
- **채널:** 전송 서비스를 제공하는 클래스
- **압축기:** 데이터 압축과 압축 해제를 수행할 수 있는 압축기 오브젝트나 널(null)
- **암호화기:** 암호화 및 해독을 수행할 수 있는 암호화기 오브젝트나 널(null)
- **목적지:** 이 채널에 대한 목표(예: SERVER.XYZ.COM)

인증자는 보통 채널을 설정할 때만 사용됩니다. 압축기와 암호화기는 일반적으로 모든 흐름에서 사용됩니다.

가장 간단한 암호화기유형은 MQeXorCryptor로, 이것은 데이터의 XOR을 수행하여 송신되는 데이터를 암호화합니다. 이 암호화는 안전하지는 않지만, 데이터를 보이지 않게 합니다. 반대로, MQe3DESCryptor는 트리플 DES를 구현합니다. 가장

간단한 유형의 압축기는 MQeRleCompressor로, 이것은 반복되는 문자를 계수로 대체하여 데이터를 압축합니다. 다른 인증자, 압축기 및 암호화기도 제공됩니다. 40 페이지의 표13을 참조하십시오.

채널 구축에서는 프로토콜 어댑터 스펙을 사용하여 특정 채널에 사용할 링크와 프로토콜을 결정합니다. 각 중간 노드에서, 다음 링크로의 주소지정을 알아내기 위해 채널 정의를 탐색합니다. 어떤 정의도 존재하지 않으면, 채널은 종료되고 메시지 흐름은 그 지점에서 큐 관리자로 전달됩니다.

채널은 응용프로그램이나 큐 관리자에게 직접 표시되지 않으며 필요에 따라 큐 관리자에 의해 구축됩니다. 채널에서 큐 관리자들을 링크시키고 MQSeries Everyplace가 그 특성을 조정 및 재조정하면 정보는 그에 따라 흐르게 됩니다. 변환기는 채널을 이용하여 큐 레벨 통신을 제공하는 MQSeries Everyplace 구성요소입니다. 이들 역시 응용프로그램 프로그래머나 관리자에게 표시되지 않습니다.

보증 메시징이 요구되면 MQSeries Everyplace는 메시지들을 응용프로그램에 한 번만 전달합니다. 이는 메시지가 하나의 큐 관리자에서 다른 큐 관리자로 성공적으로 전달되고, 전송 종료 시 사본을 삭제하기 전에 메시지 수신을 확인함으로써 수행됩니다. 통신 실패가 발생할 경우, 수신확인이 수신되지 않았으면 메시지는 재전송될 수 있지만(1회 전달이 1회 전송을 의미하지는 않음) 중복되는 메시지는 전달되지 않습니다.

어댑터

어댑터는 MQSeries Everyplace를 장치 인터페이스에 맵핑하기 위해 사용됩니다. 채널은 프로토콜 어댑터를 사용하여 HTTP, 원시 TCP/IP 및 다른 프로토콜을 통해 실행됩니다. 마찬가지로, 큐는 필드 스토리지 어댑터를 사용하여 메모리나 파일 시스템과 같은 스토리지 서브시스템에 인터페이스합니다. 어댑터는 MQSeries Everyplace에 대한 장치 지원 확장 및 버전화 메카니즘을 제공합니다.

파일 설명자는 어댑터를 식별하고, 로드 및 활성화하기 위해 사용되는 문자열입니다.

전화 접속 연결 관리

장치에 대한 전화 접속 네트워킹 지원은 장치 운영 체제에서 처리됩니다. 연결해 제된 장치상의 MQSeries Everyplace가 네트워킹을 사용하려고 할 때, 예를 들자 면 메시지는 송신되어야 하는데 네트워크 스택이 활성화되지 않은 경우, 운영 체 제 자체가 원격 액세스 서비스(RAS)를 초기화하기 때문입니다. 일반적으로, 이 때 전화 접속 연결 프로파일을 제공하는 패널 양식이 사용자에게 표시됩니다. 연결이 구축될 때까지, 운영 체제는 제어 하에 있습니다. 결과적으로, 장치 사용자는 운영 체제가 사용할 적절한 전화 접속 연결 프로파일이 사용 가능하도록 해야 합니다. 그러므로, MQSeries Everyplace 장치 구현 시 전화 접속 네트워킹에 대한 명시 적 지원은 필요하지 않습니다.

추적

추적은 추적 조치를 수행하는 독립적 프로그램을 실행할 때 가능합니다. MQSeries Everyplace 내에 포함된 것은 시스템 및 사용자 변화에 대한 정보, 경고 및 오류 상황은 추적하기 위한 호출입니다. 응용프로그램은 직접 추적을 호출하여 새로운 메시지를 추가하거나 기존 추적 메시지를 수정할 수 있습니다. 제공된 샘플 추적 프로그램은 선택된 메시지를 표시, 인쇄 또는 이벤트 로그에 기록할 수 있습니다. 추가 기능을 사용하여 다른 추적 프로그램을 작성하거나 다른 방식으로 출력을 형 식화하고 전달하도록 설계할 수 있습니다.

대부분의 MQSeries Everyplace 예외는 처리를 위해 응용프로그램에 전달되지만, 응용프로그램 예외 핸들러가 이를 추적으로 보낼 수도 있습니다.

이벤트 로그

MQSeries Everyplace는 로그 상태(예를 들어, 큐 관리자가 시작됨)에 대해 사용 할 수 있는 이벤트 로그 메커니즘과 인터페이스를 제공합니다. 로깅은 초기화될 수 있으며 기본적으로 파일로 기록될 수 있으나, 어디에서나 가로채기하여 경로를 다 시 지정할 수 있습니다. MQSeries Everyplace 이벤트 로그는 메시지 데이터를 기 록하지 않으므로 메시지나 큐의 복구에 사용할 수 없습니다.

MQSeries Everyplace 네트워크

MQSeries Everyplace 네트워크는 장치와 게이트웨이의 연결입니다. 여러 개의 물리적 네트워크에 다리를 놓아 분산되고 그 사이에서 메시지를 라우팅할 수 있습니다. 일반적으로, 이 네트워크는 큐 위치에 관계 없는 프로그래밍 모델과 함께 큐에 대한 동기 및 비동기 액세스를 제공합니다.

구성 및 확장성

기본 MQSeries Everyplace 네트워크 토폴로지의 선택이 37 페이지의 그림5에 나와 있습니다. 이러한 목적일 때, 각 네트워크는 동기 및 비동기 통신 기능 둘 다를 사용하여 구성됩니다.

가장 간단한 경우는 위의 (a)에서처럼 독립형 장치가 로컬 큐를 통해 동기식 응용 프로그램간 통신을 지원하는 것입니다. 그러나 더 흥미로운 것은 (b) 경우로, 이는 피어-투-피어 네트워크를 보여줍니다. 이 때, 두 장치 모두 같은 통신 프로토콜을 사용하고 또다른 접속 시도에 응답할 수 있도록 적어도 하나는 청취 기능을 갖도록 구성되어야 합니다. 이렇게 단순한 경우, 통신은 두 장치 모두 같은 네트워크에서 사용 가능한 경우에만 가능합니다. 비동기 메시징을 사용하면 응용프로그램은 장치가 연결되어 있지 않을 때 실행될 수 있지만, 동기 메시징은 장치가 실제로 연결된 경우에만 가능합니다.

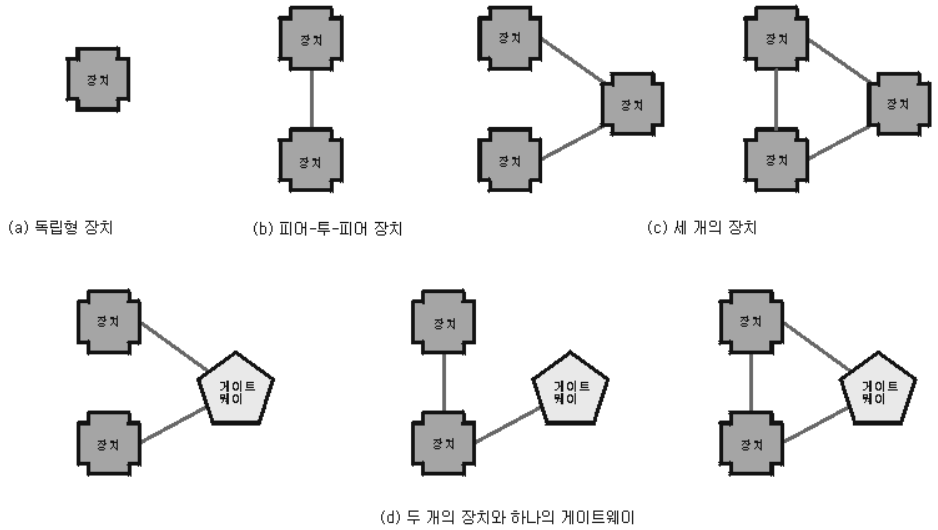


그림 5. 단순한 MQSeries Everywhere 네트워크

세 개의 장치를 사용할 경우 더 많은 가능성이 있습니다. 어느 하나가 링크 장치의 역할을 수행하거나, 세 개 모두가 서로 연결될 수 있습니다. 장치들이 피어 기준으로 메시지를 서로 교환하도록 구성된 경우, 고려해야 할 사항이 있습니다. 그 장치들이 세번째 장치를 통해 통신할 경우, 융통성은 더 커집니다. 두 개의 통신 장치는 같은 통신 프로토콜을 사용하지 않아도 되지만, 각각은 브릿징 장치에서 인식되는 것을 사용해야 합니다. 또한, 비동기 연결은 송신자와 수신자가 동시에 네트워크에 있지 않을 경우 메시지가 이동될 수 있도록 합니다(링크 장치도 사용 가능할 때 각각 네트워크에 있는 경우). 동기 통신에서는 세 개의 장치 모두가 동시에 사용 가능해야 합니다. 자주 연결해제되는 장치의 경우, 제3의 노드를 통해 이동되는 구성이 아주 적합합니다(일반적으로 중간 매개가 사용 가능한 경우).

사실상, 장치는 게이트웨이에 의해 링크됩니다. 이에 대한 세 가지의 예가 그림5 (d)에 있습니다. 링크 노드로서 게이트웨이를 선호하는 것은 게이트웨이가 여러 개의 동시 수신 연결 요청을 지원하는 데 기초한 것입니다. 두번째와 세번째를 실행할 수는 있지만 약간 특이할 경우, 표시된 첫번째 구성이 대개 사용됩니다. 세번째 구성에서는 특정 리포트 큐 관리자에 대해 하나의 경로만 구성될 수 있으므로, 두 개의 경로가 존재하더라도 하나만 선택해야 된다는 점에 유의하십시오.

대규모 네트워크의 경우 여러 개의 게이트웨이를 사용할 수 있으며, 각 게이트웨이는 여러 개의 장치를 지원합니다. 게이트웨이는 어떤 방법으로든지 상호 연결될 수 있지만, 완전한

상호 연결이 정의된 경우 장치 사이의 어떤 경로도 세 개 이상의 게이트웨이를 포함하지 않습니다. 그림6에 대규모 네트워크에 대한 예가 나와 있습니다.

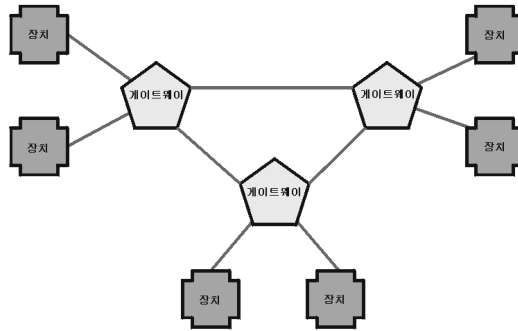


그림 6. 스타 MQSeries Everyplace 네트워크

비동기 메시지 전달

메시지가 비동기식으로 리모트 큐에 넣어질 경우, 메시지 오브젝트는 목적지 큐 관리자 및 큐 이름, 그리고 메시지의 목표 목적지가 일치하는 압축기, 인증자 및 암호화기와 함께 그 큐의 지역 정의와 연관되는 백업 저장소에 논리적으로 위치됩니다. 오브젝트가 목적지 큐에 의해 정의된 대로 영구 스토리지에 보안 형식으로 저장될 때, 오브젝트의 dump 메소드가 호출됩니다. 큐 관리자는 메시지 전달을 제어합니다. 이것은 적절한 특성을 사용하여 다음 홉(hop)의 큐 관리자로 채널을 식별(또는 구축)한 후, 목표 큐에 대한 전송기를 작성(또는 재사용)합니다. 전송기는 오브젝트를 덤프하고 결과 바이트 문자열을 전송합니다. 목표 큐 관리자와 큐 이름은 그 메시지 흐름의 일부가 아닙니다.

적합하다면 메시지는 채널을 거쳐 암호화되고 압축됩니다. 목적지 큐 관리자에 도착하였으면, 해독되고 압축해제됩니다. 목적지 큐에 위치될 결과 오브젝트와 함께 오브젝트 클래스의 restore 메소드를 사용하여 새로운 메시지 오브젝트가 작성됩니다. 메시지가 목적지 큐 관리자에 도달하지 못하면, 해독되고 압축해제된 후 전송에 적합한 특성들과 함께 저장 및 포워드 큐에 위치됩니다. 두 경우 모두에서 메시지는 목적지 큐에 정의된 대로 해당되는 각각의 큐에 보안 형식으로 보유됩니다.

비동기 메시지 전달의 특성은 메시지가 전송을 위해 큐 처리되면서, 중간 홉에서 큐 관리자에 전달되는 것입니다. 메시지는 먼저 우선순위에 따라, 그리고 나서 시간소인 순서로 중간 큐를 떠납니다.

동기 메시지 전달

동기 메시지 전달은 위에 설명된 비동기 경우와 유사하지만, 전송기와 채널을 포함하여 중간 매개 홉에서의 큐 관리자 관여가 더 낮은 레벨에서 수행됩니다. 채널은 각각의 중간 노드에서 프로토콜 스펙에 정의된 어댑터를 사용하여, 다음 링크를 식별하기 위해 종단간(end-to-end)에 구축됩니다. 관련된 파일 설명자가 더 이상 존재하지 않는 마지막 링크의 끝에서, 메시지는 처리를 위해 큐 관리자의 상위 계층으로 전달됩니다. 그러므로, 송신 노드는 메시지를 큐에 넣지 않지만, 중간 홉을 통해 채널에 따라 이를 전달하여 목표 큐에 위치시키기 위해 목적지 큐 관리자에 제공합니다.

MQSeries에 대한 링크는 게이트웨이에서 브릿지 큐를 사용합니다. 이 큐는 메시지를 MQSeries 형식으로 변환합니다. 이 메커니즘은 게이트웨이에서 종료되는 동적 채널 덕분에 장치로부터의 동기 MQSeries Everyplace 유형 메시징이 MQSeries에 대해 가능하다는 것을 의미합니다. 메시지는 클라이언트 채널을 통해 게이트웨이에서 MQSeries 서버로 실시간에 전달됩니다. 이 때, 목적지는 메시지가 MQSeries 메시지 채널을 따라 비동기식으로 전송될 것을 요구할 수 있습니다.

유사한 방식으로, 동기 메시징만 가능한 장치는 적합한 중간 매개가 사용 가능할 경우 메시지를 비동기 MQSeries Everyplace 큐에 전송할 수 있습니다.

보안

MQSeries Everyplace는 지역적으로 보류된 경우와 전송될 경우에 메시지 데이터를 보호할 수 있는 통합된 보안 기능 세트를 제공합니다.

MQSeries Everyplace 보안 기능은 세 가지 다른 범주의 보호를 제공합니다.

- 지역 보안 - 메시지(및 기타) 데이터의 지역 보호
- 큐를 기반으로 하는 보안 - 초기설정 중인 큐 관리자와 목표 큐 사이의 메시지 보호

- 메시지 레벨 보안 - 초기설정자(initiator)와 수신자 사이의 메시지들에 대한 메시지 레벨 보호

MQSeries Everyplace 지역 및 메시지 레벨 보안은 MQSeries Everyplace에서 내부적으로 사용되지만, MQSeries Everyplace 응용프로그램에 대해서도 사용할 수 있습니다. MQSeries Everyplace 큐 기반 보안은 내부 서비스입니다.

MQSeries Everyplace 보안 기능은 세 범주 모두 속성(MQeAttribute 또는 후손)을 사용하여 메시지 데이터를 보호합니다. 범주에 따라, 속성은 명시적으로 또는 암시적으로 적용됩니다.

각 속성에는 다음과 같은 오브젝트가 포함됩니다.

- 인증자
- 암호화기
- 압축기
- 키
- 목표 엔터티 이름

이 오브젝트들은 MQSeries Everyplace 보안 기능의 범주에 따라 다르게 사용되지만, 모든 경우에서 메시지 오브젝트에 딸린 속성이 호출될 때 MQSeries Everyplace 보안 기능의 보호가 적용됩니다. 이는 MQSeries Everyplace 메시지의 'dump' 메소드가 호출될 때 발생합니다(예를 들어, 메시지 데이터를 암호화하고 압축하기 위해 속성의 'encodeData' 메소드가 사용될 경우). MQSeries Everyplace 보안 기능의 비보호는 MQSeries Everyplace 메시지의 'restore' 메소드가 호출될 때 발생합니다(예를 들어, 메시지 데이터를 해독하고 압축해제하기 위해 속성의 'decodeData' 메소드가 사용될 경우).

인증을 위해 MQSeries Everyplace 버전 1.0이 지원하는 알고리즘인 암호화와 압축은 표13에 자세히 설명되어 있습니다.

표 13. 인증, 암호화 및 압축 지원

기능	알고리즘
인증	Mini Certificate 기반(WAP 포럼 WTLS Mini Certificate 에서 유래됨)
	Windows NT/2000 ID유효성 확인

표 13. 인증, 암호화 및 압축 지원 (계속)

기능	알고리즘
압축	LZW
	RLE
암호화	트리플 DES
	DES
	MARS
	RC4
	RC6
	XOR

MQSeries Everyplace 지역 보안

지역 보안은 MQSeries Everyplace 메시지(또는 MQeFields나 MQeFields 후손) 데이터를 지역적으로 보호합니다. 이는 적절한 대칭적 암호화기와 압축기로 속성을 작성하고, (암호나 통과 문구를 제공하여) 적절한 '키'를 작성하고 설정한 후, 그 키를 명시적으로 속성에 접속시키고 속성을 MQSeries Everyplace 메시지에 접속하면 가능합니다. MQSeries Everyplace는 MQeLocalSecure 클래스를 제공하여 지역 보안 설정을 보조하지만, 적절한 속성을 설정하고 암호나 통과 문구 키를 관리하는 것은 언제나 지역 보안 사용자(내부적으로는 MQSeries Everyplace, 또는 MQSeries Everyplace 응용프로그램)의 책임입니다.

MQSeries Everyplace 큐 기반 보안

큐 기반 보안은 동기 및 비동기 메시지에 적용됩니다.

동기식 큐 기반 보안

동기식 큐 기반 보안을 사용하면 응용프로그램이 MQSeries Everyplace에 대한 모든 메시지 보안 고려사항에서 벗어날 수 있습니다. 큐에는 인증, 암호화 및 압축 특성이 있으며 이 특성들은 메시지 흐름을 보호하는데(그리고, 영구 저장을 위해) 필요한 보안 레벨을 결정하는 데 사용됩니다.

메시지가 송신될 때, 목표 큐의 보안 특성이 지역 레지스트리로부터 검색됩니다. 이 특성이 없으면, 큐 관리자는 목표 큐 관리자로부터 목표 특성을 검출하려고 시도

하며 그 특성을 나중에 사용하기 위해 캐시에 넣습니다. 해당되는 큐 관리자에 대해 채널이 존재할 경우, 그 채널을 사용합니다. 그렇지 않으면, 새로운 채널이 작성됩니다. 목표 큐 속성이 검색됩니다.

필요한 서비스 품질에 따라서, 목표 큐 관리자에 대한 채널 속성이 동적으로 변경됩니다. 이것은 제정한 규칙에 따라 달라집니다. 일반적으로, 규칙은 보안 레벨의 업그레이드를 허용합니다(예를 들어, 보호 없음에서 약한 보호로, 또는 약한 보호에서 강한 보호로). 채널을 업그레이드할 수 없거나, 보안 레벨이 초과한 것으로 보이면(예를 들어, 어떤 보호도 필요하지 않으며 사용 가능한 채널이 강한 보호를 구현함) 새로운 채널이 작성됩니다. 가능하면 다시 사용되는 채널 풀(pool)이 존재하며, 이는 트래픽 요구에 따라 동적으로 특성이 변경됩니다. 채널은 필요하지 않을 경우 자동으로 소멸됩니다. 메시지는 항상 목표 큐 특성에 의해 정의된 보안 레벨로 큐에 위치됩니다.

인증은 채널 레벨에서 발생하며 메시지당 오버헤드는 최소로 유지됩니다. 동기식 큐 기반 보안은 일반적으로 대칭적 암호화와 함께 사용되기도 합니다. 빠른 암호화/해독을 가져오기 때문입니다. 그러나, 이러한 대칭적 경우에 MQSeries Everywhere는 RSA 비대칭 암호화를 초기에 사용하여 송신자 및 수신자에서 공유 키를 구축하는 데 필요한 흐름을 보호합니다. 그 지점 다음에는 대칭적 암호화가 사용되어 이동 중인 데이터의 신뢰를 보호합니다. MQSeries Everywhere는 각 채널 흐름에서 동적으로 키를 변경하여 이 데이터의 암호에 대한 공격을 더 어렵게 만듭니다. MQSeries Everywhere는 또한 데이터 송신 전에 요약을 생성하여 데이터에 추가하고, 그것을 수신할 때 재작성 및 검증함으로써 이동된 데이터의 무결성을 보장합니다.

비동기 큐 기반 보안

비동기 메시징은 putMessage 실행 시 목표 큐에 액세스 가능하다는 보증이 없는 한 위에 설명된 동기 메시징 경우와 다릅니다. 이러한 경우, 큐 관리자는 메시지를 즉시 송신할 수 없어서 이를 트랜스미션 큐에 위치시킵니다. 그러나, 이것은 목표 큐 특성에 따라 암호화됩니다. 메시지가 전송될 수 있으면, 이 메시지는 해독된 후 적합한 특성을 가지고 있는 채널로 보내집니다. 그러므로, 메시지는 전송을 기다리는 동안에도 항상 보호됩니다. 비동기 메시징에는 원격 큐 정의가 필요합니다. 그렇지 않으면 목표 큐 특성을 판별할 수 없습니다.

비동기 메시징의 경우, 시작된 곳과 목표 사이에 인증은 불가능합니다. 인증이 중요할 경우(예를 들어, 승인을 판별하거나 비거부를 확립하기 위해 수신자가 메시지가 시작된 곳을 판별해야 하거나 초기설정자가 의도한 수신자만 메시지를 처리할 수 있도록 보증하기 위해), 반드시 메시지 레벨 보안을 사용해야 합니다.

메시지 레벨 보안과 동시에 큐 기반 보안을 사용할 수 있지만, 이것은 메시지 데이터가 이미 보호되므로 필요하지 않습니다.

메시지 레벨 보안

메시지 레벨 보안은 MQSeries Everyplace 응용프로그램 초기설정 및 수신 사이에 메시지 데이터 보호를 제공합니다.

메시지 레벨 보안은 메시지 속성을 설정하고 `putMessage`를 사용하여 메시지를 목표 큐에 넣을 때 이를 제공하기 위해 MQSeries Everyplace 응용프로그램을 초기화해야 하는 응용프로그램 계층 서비스입니다. 수신하는 응용프로그램은 일치하는 메시지 레벨 속성을 설정하고 이를 수신 큐 관리자에 전달하여, 응용프로그램이 `getMessage`를 호출하여 목표 큐로부터 메시지를 가져올 때 이 속성을 사용할 수 있도록 해야 합니다.

지역 보안처럼, 메시지 레벨 보안은 메시지 오브젝트에서 속성의 응용프로그램을 이용합니다. 초기설정 중인 응용프로그램의 큐 관리자는 이 속성의 'encodeData' 메소드를 사용하여 메시지 데이터를 보호하는 'dump' 메소드로 `putMessage`를 처리합니다. 수신 응용프로그램의 큐 관리자는 이 속성의 'decodeData' 메소드를 사용하여 원래 메시지 데이터를 복구하는 'restore' 메소드로 응용프로그램의 `getMessage`를 처리합니다.

MQSeries Everyplace는 메시지 레벨 보안에 대해 두 가지의 대체 속성을 제공합니다.

MQeMAttribute

이것은 상호 트러스트가 응용프로그램 계층에서 탄탄하게 관리되고 트러스트되는 제 삼자가 필요하지 않은 비즈니스간 통신에 사용됩니다. 사용 가능한 모든 MQSeries Everyplace 대칭적 암호화기 및 압축기 선택사항을 이용할 수 있습니다. 지역 보안처럼, 속성의 키는 `putMessage` 또는 `getMessage`에서 제공되기 전에 사전설정되어야 합니다. MQeAttribute는

공용 키 내부구조(PKI)의 오버헤드 없이 메시지 기밀성을 보호하기 위해 강한 암호화를 가능하게 하는 간단하고 강력한 메시지 레벨 보호 방법을 제공합니다.

MQeMTrustAttribute

이 속성은 디지털 서명을 사용하고 기본 공용 키 내부구조를 이용하는 고급 솔루션을 제공합니다. 이 속성은 ISO9796 디지털 서명/유효성 확인을 사용하여 메시지가 알려진 송신자로부터 제공된다는 증명을 수신 응용프로그램이 할 수 있게 합니다. 제공된 속성의 암호화기를 사용하여 메시지 기밀성을 보호합니다. SHA1 요약(digest)은 메시지 무결성을 보증하고 RSA 암호화/해독은 의도한 수신자에 의해서만 메시지가 복원될 수 있도록 합니다. MQeMAttribute에서처럼, 사용 가능한 모든 MQSeries Everyplace 대칭적 암호화기 및 압축기 선택사항을 이용할 수 있습니다. 크기 최적화를 위해 선택할 경우, 사용되는 인증서는 WAP 포럼 WTLS 스펙에서 제안하는 WTLS 인증서를 기초로 한 Mini Certificate입니다. 인증(서명 확인) 및 암호화/해독에 필요한 정보의 상호 가용성이 MQSeries Everyplace 기본 내부구조를 통해 제공됩니다.

일반적인 MQeMTrustAttribute 보호 메시지의 형식은 다음과 같습니다.

RSA-enc{SymKey}, SymKey-enc {Data, DataDigest, DataSignature}

설명:

- RSA-enc:** 의도한 수신자의 공용 키로 암호화되는 RSA
- SymKey** 생성된 유사-무작위 대칭적 키
- SymKey-enc** SymKey와 함께 대칭적으로 암호화됨
- Data** 메시지 데이터
- DataDigest** 메시지 데이터의 요약
- DigSignature** 메시지 데이터에 대한 초기설정자(initiator)의 디지털 서명

메시지 레벨 보안은 큐 레벨 보안과 별개입니다.

레지스트리

레지스트리는 큐 관리자 관련 정보에 대한 기본 저장소로, 각 큐 관리자마다 하나 씩 존재합니다. 모든 큐 관리자는 다음을 보유하기 위해 레지스트리를 사용합니다.

- 큐 관리자 구성 데이터
- 큐 정의
- 리모트 큐 정의
- 리모트 큐 관리자 정의
- 사용자 데이터(구성별 보안 정보를 포함하여)

레지스트리에 대한 액세스는 보통 합법적인 큐 관리자 사용자로 제한되고 PIN으로 보호되지만, 보안보다 footprint 크기에 더 관심이 있는 사용자는 구성 가능한 옵션으로 이것을 생략할 수 있습니다.

MQSeries Everyplace 인증 가능 항목

Mini Certificate를 기반으로 하는 상호 인증을 사용하는 큐 기반 보안과 디지털 서명을 사용하는 메시지 레벨 보호에서 '인증 가능한 엔터티' 개념이 발생되었습니다. 상호 인증의 경우, 두 사용자(사람) 사이의 인증에 대해 생각하는 것이 보통이지만, 일반적으로 메시징에는 사용자 개념이 없습니다. 일반적으로 이 개념은 응용프로그램 레벨에서, 즉 메시징 서비스의 사용자에게 의해 관리됩니다. MQSeries Everyplace는 '인증의 목표' 개념을 사용자에서 '인증 가능한 엔터티'로 추상화합니다. 여기에서는 인증 가능한 엔터티가 사람이 되는 것을 배제하고 있지만, 이는 응용프로그램에서 선택되는 맵핑입니다. 내부적으로, MQSeries Everyplace는 Mini Certificate 종속 서비스의 목표가 될 수 있거나 목표를 생성할 수 있는 모든 큐 관리자를 인증 가능한 엔터티로 정의합니다. 또한, MQSeries Everyplace는 Mini Certificate를 기반으로 하는 인증자를 사용하도록 정의된 큐를 인증 가능한 엔터티가 되도록 정의합니다. 그러므로, 이러한 서비스를 지원하는 큐 관리자는 하나의 인증 가능한 엔터티, 즉 큐 관리자 또는 인증 가능한 엔터티의 세트, 즉 큐 관리자와 인증서 기반 인증자를 사용하는 모든 큐를 가지고 있습니다.

개인 레지스트리 및 신임장

모든 인증 가능한 엔터티가 유용하게 되려면 고유한 보증이 있어야 합니다. 여기에는 두 가지의 과제가 있습니다. 먼저, 보증서를 얻기 위해 어떻게 등록을 실행하는가이고, 두번째는 어디에서 보증서를 안전하게 관리할 수 있는가 입니다. 전형적으로 볼 때, 이 과제들은 기초가 되는 암호화 기술보다 해결하기가 더 어렵습니다. MQSeries Everyplace는 공용 보증서 세트를 관리할 공용 레지스트리(기본 레지스트리의 후손)와, 인증 가능한 엔터티의 개인용 보증서를 안전하게 관리할 수 있도록 하는 개인용 레지스트리(기본 레지스트리의 후손), 그리고 자동 등록을 수행하는 것 등을 인증 가능한 엔터티가 할 수 있도록 해주는 기본 서비스를 제공합니다. 개인 레지스트리는 많은 보안 또는 암호화 토큰 수준의 기본 레지스트리를 제공합니다. 예를 들어, 이것은 Mini Certificate와 같은 공용 오브젝트 및 개인 키와 같은 개인 오브젝트용의 안전한 저장소가 될 수 있습니다. 이것은 권한이 부여된 사용자만 개인 오브젝트에 액세스할 수 있게 하는 메커니즘을 제공합니다. 또한, 개인 오브젝트가 개인 레지스트리를 떠나지 않는 방식으로 서비스(예를 들어, 디지털 서명, RSA 해독)에 대한 지원을 제공합니다. 공통 인터페이스를 제공하여, 현재 지역 파일 시스템으로 제한되는 기초 장치 지원을 숨기지만, 나중에 이동성 토큰에 맵핑되도록 확장될 수 있습니다.

자동 등록

MQSeries Everyplace는 자동 등록을 지원하는 기본 서비스를 제공합니다. 이 서비스들은 인증 가능한 엔터티가 구성될 때(예를 들어, 큐 관리자가 시작되거나 새로운 큐가 정의될 경우) 자동으로 트리거됩니다. 두 경우 모두에서 등록이 트리거되고 새로운 보증서가 작성되어 인증 가능한 엔터티의 개인 레지스트리에 저장됩니다. 자동 등록 단계에는 새로운 RSA 키 쌍 생성, 개인 레지스트리에서 개인 키 보호 및 저장, 그리고 기본 Mini Certificate 서버에 대한 '새 인증서' 요청에서 공용 키를 패키지로 묶는 작업이 포함됩니다. Mini Certificate 서버가 구성되어 사용 가능할 경우, 그 서버는 인증 가능한 엔터티의 새로운 Mini Certificate를 고유한 Mini Certificate와 함께 리턴하며, 이들은 보호되는 개인 키와 함께 인증 가능한 엔터티의 개인 레지스트리에서 새로운 보증서로 저장됩니다. 자동 등록은 간단한 메커니즘을 제공하여 인증 가능한 엔터티의 보증서를 구축하는 반면, 메시지 레벨 보호(MqeMTrustAttribute, 위 참조)의 경우 의도한 수신자의 공용 키(Mini Certificate)에 대한 액세스도 요구합니다.

공용 레지스트리 및 인증서 복제

MQSeries Everyplace는 MQSeries Everyplace 구성요소 사이에 인증 가능한 엔터티 공용 보증서(Mini Certificate)를 공유할 수 있게 하는 기본 서비스를 제공합니다. 이것들은 MQeMTrust를 기반으로 하는 메시지 레벨 보안의 전제조건입니다. MQSeries Everyplace 공용 레지스트리는 Mini Certificate에 공용으로 액세스 가능한 저장소를 제공합니다. 이 레지스트리는 휴대폰의 개인 전화 디렉토리 서비스에 대해 아날로그식이며, 그 차이점은 전화 번호가 아니라 가장 자주 접속되는 인증 가능한 엔터티들의 Mini Certificate 세트라는 점입니다. 공용 레지스트리는 완전히 수동적인 서비스는 아닙니다. 보유하고 있지 않은 Mini Certificate를 제공하라는 액세스가 있으며 그것이 유효한 홈 서버 구성요소로 구성된 경우, 공용 레지스트리는 요청되는 Mini Certificate를 자동적으로 홈 서버의 공용 레지스트리로부터 페치하려고 합니다. 이 서비스들은 올바른 Mini Certificate를 올바른 시간에 이용하게 해주는 자동화된 지능형 Mini Certificate 등록 서비스를 제공하기 위해 사용됩니다.

레지스트리 서비스의 응용프로그램 사용

MQSeries Everyplace 큐 관리자는 개인 및 공용 레지스트리 서비스를 사용할 경우의 이점을 이용하도록 설계되어 있지만, 이 서비스들에 대한 액세스는 제한되지 않습니다. MQSeries Everyplace 솔루션은 고유한 인증 가능한 엔터티(예: 사용자)를 정의하고 관리할 것을 요구할 수도 있습니다. 개인 레지스트리 서비스는 새로운 인증 가능한 엔터티의 보증서를 자동 등록하고 관리하기 위해 사용하고, 공용 레지스트리 서비스는 필요한 경우 공용 보증서를 사용 가능하게 만드는 데 사용됩니다. 등록된 모든 인증 가능한 엔터티는 MQeMTrustAttribute로 보호되는 메시지 레벨 서비스의 초기설정자(initiator)나 수신자로 사용될 수 있습니다.

기본 Mini Certificate 발행 서비스

MQSeries Everyplace는 개인 레지스트리 자동 등록 요청을 만족시키도록 구성될 수 있는 기본 Mini Certificate 발행 서비스를 제공합니다. MQSeries Everyplace와 함께 제공된 도구를 사용하면, 신중하게 제어된 엔터티 이름 세트에 대해 Mini Certificate를 발행하도록 Mini Certificate 발행 서비스를 설정하고 관리할 수 있습니다. 이 보증 서비스의 특성은 다음과 같습니다.

- 등록된 인증 가능한 엔터티 세트의 관리

- Mini Certificate 보증(WAP WTLS Mini Certificate를 기초로 하는 Mini Certificate)
- Mini Certificate 저장소 관리

MQSeries Everyplace에서 제공되는 도구는 Mini Certificate 보증 서비스 관리자가 엔터티 이름과 등록된 주소를 등록하고 일회용 인증서 요청 PIN을 정의하여 제공된 엔터티에 Mini Certificate 발행 권한을 부여할 수 있게 합니다. 이것은 보통 오프라인 점점으로 요청자의 권한을 확인한 후에 수행됩니다. 인증서 요청 PIN은 의도한 사용자에게 이송됩니다(예를 들어, 새로운 은행 카드가 발행될 때 은행 카드 PIN이 사용자에게 이송되는 방식과 비슷하게). 개인 레지스트리의 사용자(예 : MQSeries Everyplace 응용프로그램이나 MQSeries Everyplace 큐 관리자)는 시작 시 이 인증서 요청 PIN을 제공하도록 구성될 수 있습니다. 개인 레지스트리가 자동 등록을 트리거할 경우, Mini Certificate 발행 서비스는 결과로 생성되는 새로운 인증서 요청을 확인하고(제시된 엔터티 이름과 인증서 요청 PIN을 사전에 등록된 값과 대조하여), 새로운 Mini Certificate를 발행한 후 등록된 인증서 요청 PIN을 재사용할 수 없도록 재설정합니다. 새로운 모든 자동 등록 Mini Certificate 요청은 보안 채널에서 처리됩니다.

Mini Certificate 발행 서비스에 의해 발행된 Mini Certificate 세트는 보증 서비스의 고유 레지스트리에서 보유됩니다. Mini Certificate가 재발행되면(예를 들어, 만기되어) 만기된 Mini Certificate는 보존됩니다.

보안 인터페이스

사용자 정의 보안 관리자에 의해 구현될 수 있는 선택적 인터페이스가 제공됩니다. 메소드는 보안 관리자가 다음 사항에 연관되는 요청을 거부하거나 권한을 부여할 수 있게 합니다.

- 클래스 별명의 추가 또는 제거
- 어댑터 정의
- 파일 설명자의 맵핑
- 채널 명령의 처리

구성 및 조정

규칙

규칙은 다양한 상태 변경이 발생할 경우 MQSeries Everyplace의 작동(behavior)을 조정하는 데 사용되는 Java 클래스입니다. 필요하다면 기본 규칙이 제공되지만, 이 규칙들은 고객 요구사항에 맞도록 응용프로그램이나 설치에 특정한 규칙으로 대체될 수 있습니다. 지원되는 규칙 유형은 기능이 아니라, 트리거되는 방법에서 다릅니다. 규칙에는 논리가 포함되므로 광범위한 기능을 수행할 수 있습니다.

속성 규칙

이 규칙 클래스는 상태를 변경할 때마다 제어를 받습니다. 예를 들어, 다음과 같은 변경이 있을 수 있습니다.

- 인증자
- 압축기
- 암호화기

규칙은 보통 변경을 허용하기도 하고, 허용하지 않기도 합니다.

MQSeries 브릿지 규칙

이 규칙 클래스는 MQSeries 대 MQSeries Everyplace 브릿지 코드 상태가 변경될 경우에 제어를 받습니다. 다음을 각각 판별하기 위한 별도의 브릿지 규칙 클래스가 있습니다.

- 리스너가 MQSeries로부터 오는 메시지를 MQSeries Everyplace로 전달할 수 없을 때 이 메시지에 대해 수행할 사항. 예를 들어, 메시지가 너무 크거나 큐가 존재하지 않을 경우에 이러한 상황이 발생합니다.
- 상태 브릿지 관리 오브젝트는 서버가 인스턴스화되면 시작해야 합니다.
- 브릿지가 MQSeries의 Sync Q(손상 복구에 사용되는 영구 저장소)에서 잘못된 것을 찾은 경우 수행할 사항. 기본 규칙은 문제점을 표시만 합니다.
- MQSeries Everyplace 메시지를 MQ 메시지로, 또는 그 반대로 변환하는 방법. MQSeries Everyplace와 MQ 메시지 사이의 변환을 수행할 변환기는 MQeRule 클래스로부터 도출되지 않으며, 그 대신 MQeTransformerInterface

인터페이스를 구현해야 합니다. 이와는 별도로, 변환기는 규칙처럼 작동하며 메시지를 형식 변환을 요구할 경우에 호출됩니다.

RAS 다이얼러 규칙

이 규칙 클래스는 RAS 다이얼러의 상태가 변경될 때마다 제어가 부여됩니다. 예를 들면, 다음과 같습니다.

- 호출할 번호로 연결할 수 없을 경우에 수행할 사항
- 오류 임계값을 초과할 경우에 수행할 사항
- 전화 접속이 시도되고 특정 유형의 연결만 시간에 따라 사용되어야 합니다(예를 들어, 피크 상태가 아니면 전화만 사용합니다).

큐 규칙

이 규칙 클래스는 연관된 큐의 상태가 변경될 때마다 제어를 받습니다. 예를 들면, 다음과 같습니다.

- 큐에 메시지 추가. 예를 들어, 임계값 초과 여부(메시지 수, 메시지 크기, 유효하지 않은 우선순위)를 볼 경우.
- 큐 특성이 할당되거나 변경될 경우
- 큐가 열리거나 닫히는 경우
- 큐가 삭제될 경우

큐 관리자 규칙

이 규칙 클래스는 큐 관리자의 상태가 변경될 때마다 제어를 받습니다. 예를 들면, 다음과 같습니다.

- 큐 관리자가 열린 경우. 예를 들어, 시간 제한된 조치가 발생할 수 있도록 백그라운드 타이머 스레드 실행을 시작합니다.
- 큐 관리자가 닫힌 경우. 예를 들어, 백그라운드 타이머 스레드를 종료합니다.
- 새로운 큐가 추가된 경우

연결 유형

MQSeries Everyplace는 클라이언트-서버¹⁰ 및/또는 피어-투-피어조작을 지원할 수 있습니다. 클라이언트는 서버와의 통신을 초기화할 수 있고, 서버는 클라이언트에 의해 초기설정된 요청에 응답할 수만 있습니다. 피어-투-피어(peer-to-peer) 조작에서, 두 개의 피어는 어느 방향으로든지 흐름을 초기설정할 수 있습니다. 이 연결 유형을 사용하려면 MQSeries Everyplace의 여러 가지 구성요소가 사용 가능하고 활성화되어야 합니다. 포함되는 구성요소는 다음과 같습니다.

- 채널 리스너: 수신되는 연결 요청을 청취합니다.
- 채널 관리자: 끝점 사이의 여러 개의 논리적 동시 통신 파이프를 지원합니다.
- 큐 관리자: 메시징 및 큐 처리 기능을 공급하여 응용프로그램을 지원합니다.

표14에서 이러한 구성요소와 연결 유형 사이의 관계를 보여줍니다. 클라이언트/서버 연결 유형은 MQSeries Everyplace이 클라이언트나 서버 모드에서 작동될 수 있는 상황에 대해 설명합니다. 서블릿 옵션은 MQSeries Everyplace가 수신되는 연결 요청을 청취할 책임이 있는 HTTP 서버에 대해 HTTP 서블릿으로 구성되는 경우에 대해 설명합니다.

표 14. 연결 유형

	큐 관리자	채널 관리자	채널 리스너
클라이언트	예		
클라이언트/서버	예	예	예
피어	예		
서버	예	예	예
서블릿	예	예	

MQSeries Everyplace 채널 관리자나 MQSeries Everyplace 채널 리스너를 사용하면 사용권 부여 목적을 위해 MQSeries Everyplace 인스턴스가 게이트웨이 임을 판별합니다.

MQSeries Everyplace 응용프로그램은 큐 관리자에서 사용되는 연결 유형을 직접 인식하지 못합니다. 그러나, 유형은 파티에 대해 사용 가능한 자원, 다른 큐 관리

10. 이 절에서, '클라이언트'와 '서버'라는 용어는 MQSeries 의미가 아니라, 일반적인 사용을 말합니다.

자와 연결할 수 있는 큐 관리자, 장치 또는 게이트웨이에서의 MQSeries Everyplace footprint, 동시에 존재할 수 있는 연결에 영향을 준다는 점에서 중요합니다.

피어-투-피어 연결

피어-투-피어 채널에는 하나의 채널에 대한 채널 관리자와 채널 리스너의 기능이 포함됩니다. 두 개의 큐 관리자 사이에 피어-투-피어 채널이 작성될 경우, 하나의 큐 관리자는 리스너로 작동되고 다른 큐 관리자는 연결 초기설정자(initiator)로 작동되어야 합니다. 피어-투-피어 연결 큐 관리자는 다른 큐 관리자와의 여러 피어-투-피어 연결을 초기설정할 수 있지만, 이것은 하나의 수신되는 연결 요청에만 응답할 수 있으므로 피어-투-피어 채널이 닫히기를 기다린 후에 다른 연결 요청에 응답해야 합니다. 하나의 피어-투-피어 채널을 통해, 참여하는 두 개의 큐 관리자가 둘 다 조치를 초기설정할 수 있습니다. 그러므로, 각 큐 관리자의 응용프로그램은 다른 큐 관리자의 큐에 액세스할 수 있습니다.

피어-투-피어 채널은 수신되는 연결 요청의 목표가 방화벽에서 승인되지 않을 수도 있으므로 방화벽을 통해 사용하지 못할 수도 있습니다.

클라이언트-서버 연결

클라이언트-서버 연결 유형에 대해 사용되는 표준 채널에는 청취하는 기능이 없지만 서버에서 독립적인 리스너에 따라 다르므로 서버는 채널 관리자가 여러 개의 동시 채널을 처리하도록 요구합니다. 클라이언트는 연결 요청과 서버 응답을 초기 설정합니다. 서버는 보통 클라이언트로부터 여러 개의 수신 요청을 처리할 수 있습니다. 표준 채널을 통해, 클라이언트는 서버에서 자원에 대한 액세스를 갖습니다. 서버상의 응용프로그램이 클라이언트상의 자원에 대한 동기 액세스를 필요로 할 경우, 역할이 반대로 되는 두번째 채널이 필요합니다. 그러나, 표준 채널은 양 방향이므로, 서버의 트랜스미션 큐에서 해당되는 클라이언트로 보내지는 메시지는 초기설정된 표준(클라이언트-서버) 채널을 통해 전달됩니다.

하나의 클라이언트가 동시에 여러 서버에 대한 클라이언트가 될 수 있습니다. 채널 관리자는 여러 개의 인바운드 채널을 처리하므로 이 구성을 지원하지 않아도 됩니다.

클라이언트-서버 연결 유형은 수신되는 연결 요청의 목표가 보통 방화벽에 대해 승인되는 것으로 식별되므로 방화벽을 통해 사용하기에 적합합니다.

여러 연결 유형

하나의 큐 관리자가 피어-투-피어 또는 클라이언트/서버 연결 중 하나를 초기설정 하고 서버나 피어로서 응답할 수 있습니다. 이 경우, 피어 채널 리스너와 표준 채널 리스너의 포트 번호는 서로 달라야 합니다.

클래스

MQSeries Everyplace는 MQSeries Everyplace의 작동이 특정 응용프로그램 요구사항에 따라 조정될 수 있도록 특정 기능에 대한 클래스 선택사항을 제공합니다. 어떤 경우에는 클래스에 대한 인터페이스가 제시되어, 추가 대안을 개발할 수 있습니다. 표15에 기능성에 대해 요약되어 있습니다. 클래스는 명시적으로, 또는 별명 사용을 통해 식별될 수 있습니다.

표 15. 클래스 옵션

클래스	대안 제공	인터페이스 제시
관리	아니오	예
인증자	예	아니오
통신 어댑터	예	예
통신 유형	예	아니오
압축기	예	아니오
암호화기	예	아니오
이벤트 로그	샘플 제공	예
메시지	아니오	예
큐 스토리지	예	아니오
규칙	기본 클래스 제공	예
추적	샘플 제공	예

응용프로그램 로드

MQSeries Everyplace가 클라이언트(또는 피어)로 작동되도록 구성된 경우, 초기 설정 응용프로그램이 다른 응용프로그램을 JVM에 로드해야 합니다. 표준 Java 기능이 이를 위해 사용되거나, MQSeries Everyplace의 일부로 포함된 클래스 로더를 사용할 수 있습니다. 그러므로, 여러 응용프로그램이 같은 JVM에서 하나의 큐 관리자에 대해 실행될 수 있습니다. 또한, 여러 JVM이 사용될 수도 있지만 이 JVM 각각에는 고유한 큐 관리자가 있어야 하고 고유한 이름도 가지고 있어야 합니다.

MQSeries Everyplace가 서버로 구성될 경우, MQSeries Everyplace 자체가 초기설정 응용프로그램입니다. MQSeries Everyplace는 사전로드 클래스 목록을 지원하므로, 큐 관리자 자체가 로드되기 전에 이 클래스들이 차례로 로드됩니다.

제6장 MQSeries Everyplace 및 MQSeries 네트워크

MQSeries Everyplace 네트워크가 MQSeries 서버나 네트워크 없이 독립형으로 존재할 수 있어도, 사실상 MQSeries Everyplace는 종종 새로운 플랫폼과 장치에 도달하도록 확장하거나 큐나 메시지를 기반으로 하는 보안 또는 동기 메시징과 같은 고급 기능을 제공하여 기존의 MQSeries 설치를 보충하는 데 사용됩니다. MQSeries Everyplace 응용프로그램 측면에서 볼 때, MQSeries 큐 및 큐 관리자는 간단한 추가 리모트 큐와 큐 관리자로 간주될 수 있습니다. 그러나, 이러한 큐가 MQSeries Everyplace 동적 채널 및 MQSeries Everyplace 큐 관리자를 통해 직접 액세스되지 않아서 여러 기능상의 제한사항이 존재하므로, MQSeries Everyplace 게이트웨이를 연관시켜야 합니다. 게이트웨이는 여러 개의 MQSeries 큐 관리자에 대해 MQSeries 클라이언트 채널을 통해 직접 또는 간접으로 메시지를 전송할 수 있습니다. 연결이 간접적일 경우, 메시지는 MQSeries 클라이언트 채널을 통해 중간 MQSeries 큐 관리자를 거쳐 MQSeries 메시지 채널을 지나 목표 큐 관리자로 보냅니다.

목적지가 MQSeries Everyplace인 MQSeries 응용프로그램으로부터의 메시지는 정적으로 MQSeries Everyplace 큐 관리자와 큐로 주소 지정됩니다. MQSeries 경로지정(리모트 큐 관리자 정의)은 MQSeries Everyplace 메시지가 특정의 MQSeries 트랜스미션 큐에 도달하는 것처럼 정의됩니다. MQSeries 채널은 트랜스미션 큐에 대해 정의되지 않습니다. 그 대신, MQSeries Everyplace 게이트웨이가 이 큐로부터 메시지를 가져와서 MQSeries Everyplace 목적지에 전달되도록 합니다. 사용되는 트랜스미션 큐 수(경로 수)를 구성할 수 있고 전달될 메시지 양을 반영하도록 설정해야 합니다.

MQSeries에 대한 인터페이스

MQSeries Everyplace 구조는 MQSeries Everyplace와 다른 메시징 시스템 사이에서 하나 이상의 선택적 브릿지 개념을 지원합니다.

MQSeries Everyplace 버전 1.0에서는 그런 종류로는 *MQSeries* 브릿지만 지원됩니다. 이 브릿지는 MQSeries Everyplace와 MQSeries 네트워크 사이에 인터페이스합니다. 이 브릿지는 MQSeries Java 클라이언트를 사용하여 하나 이상의 MQSeries 큐 관리자에 인터페이스하므로, 메시지가 MQSeries Everyplace에서 MQSeries로, 또는 그 반대로 이동될 수 있게 합니다. MQSeries Everyplace 버전 1에서는 게이트웨이마다 그러한 브릿지가 하나씩 권장되며, 각각은 여러 개의 *MQSeries* 큐 관리자 프록시(MQSeries 큐 관리자의 정의)와 연관됩니다. 큐 관리자 프록시 정의는 MQSeries Everyplace와 통신하는 각각의 MQSeries 큐 관리자에 대해 필요합니다. 이러한 정의 각각에는 하나 이상의 클라이언트 연결 서비스가 연관될 수 있습니다. 각 서비스는 하나의 MQSeries 큐 관리자에 대한 연결을 나타냅니다. 이 서비스들은 각각 큐 관리자에 대해 다른 MQSeries 서버 연결과, 선택적으로 사용자 종료(user exit)나 포트와 같은 다른 등록정보 세트를 사용할 수 있습니다.

게이트웨이 클라이언트 연결 서비스에는 게이트웨이 서비스를 사용하여 MQSeries 큐 관리자에 연결하고 MQSeries에서 MQSeries Everyplace로 메시지를 검색하는 여러 개의 리스너가 있을 수 있습니다. 리스너는 단 하나의 서비스를 사용하여 해당되는 연결을 구축하며, 각 리스너는 MQSeries 큐 관리자의 단일 트랜스미션 큐에 연결됩니다. 각 리스너는 상위 게이트웨이 큐 관리자를 통해 단일 MQSeries 트랜스미션 큐에서 MQSeries Everyplace 네트워크상의 아무 곳으로나 메시지를 이동합니다. 그러므로, 단일 게이트웨이 큐 관리자는 여러 MQSeries 메시지 소스를 MQSeries Everyplace 네트워크로 모아줄 수 있습니다.

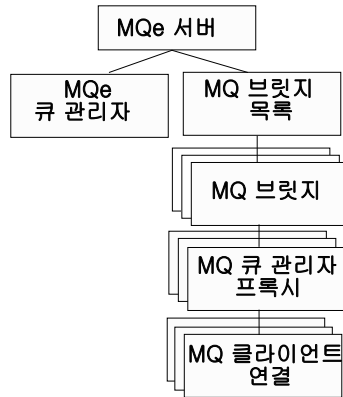


그림 7. MQSeries 브릿지 오브젝트 계층

다른 방향으로 메시지를 이동할 경우(MQSeries Everyplace에서 MQSeries로), 게이트웨이 큐 관리자는 하나 이상의 브릿지 큐 오브젝트를 구성합니다. 각 브릿지 큐 오브젝트는 직접 큐 관리자에 연결하여 해당되는 메시지를 목표 큐에 보냅니다. 이 방식에서, 게이트웨이는 단일 MQSeries Everyplace 큐 관리자를 통해 MQSeries 큐 관리자로 경로지정된 MQSeries Everyplace 메시지를 직접 또는 간접적으로 디스패치할 수 있습니다. 브릿지 오브젝트의 등록정보는 표16에 표시되어 있습니다.

표 16. 브릿지 오브젝트 등록정보

등록정보	설명
브릿지 이름	브릿지 목록
실행 상태	상태: 실행 또는 정지

브릿지 오브젝트와 다른 게이트웨이 오브젝트는 MQSeries Everyplace 큐 관리자와는 독립적으로 시작하고 정지할 수 있습니다. 그러한 게이트웨이 오브젝트가 시작(정지)된 경우, 조치는 해당되는 모든 하위 오브젝트(모든 브릿지, 큐 관리자 프록시, 클라이언트 연결 및 트랜스미션 큐 리스너)에도 적용됩니다. 브릿지 오브젝트의 등록정보는 58 페이지의 표17에 나와 있습니다.

표 17. 브릿지 등록정보

등록정보	설명
클래스	브릿지 클래스
기본 변환기	목적지 큐에 다른 어떤 변환기 클래스도 연관되지 않은 경우 메시지를 MQSeries Everyplace에서 MQSeries로(또는, 그 반대로) 변환하기 위해 사용될 기본 클래스(규칙 클래스)
하트비트 간격	브릿지 오브젝트에 대해 조치를 수행하는 데 사용될 기본 타이밍 단위
이름	브릿지 오브젝트의 이름
실행 상태	상태: 실행 또는 정지
시작 규칙 클래스	브릿지 오브젝트가 시작될 때 사용되는 규칙 클래스
MQSeries 큐 관리자 프록시 하위	이 브릿지가 소유하는 모든 큐 관리자 프록시 목록

간단한 경우에, 기본 변환기(규칙)는 모든 메시지 변환을 처리하는 데 사용할 수 있습니다. 또한, 이 기본값을 대체하는 리스너별 추가 변환기를 설정할 수 있습니다(MQSeries에서 MQSeries Everyplace로의 메시지에 대해). 특정 제어의 경우, 변환 규칙은 게이트웨이에서 브릿지 큐 정의를 사용하여 목표 큐 기준으로 설정될 수 있습니다. 이는 MQSeries Everyplace 및 MQSeries 목표 큐 둘 다에 적용됩니다.

MQSeries 큐 관리자 프록시는 단일 MQSeries 큐 관리자에 고유한 등록정보를 보유합니다. 프록시 등록정보는 표18에 나와 있습니다.

표 18. MQSeries 큐 관리자 프록시 등록정보

등록정보	설명
클래스	MQSeries 큐 관리자 프록시 클래스
MQSeries 호스트 이름	Java 클라이언트 클래스를 통해 MQSeries 큐 관리자에 대한 연결을 작성하는 데 사용되는 IP 호스트 이름. 지정하지 않으면, MQSeries 큐 관리자가 브릿지와 같은 기계에 있는 것으로 간주되고 Java 바인딩이 사용됩니다.
MQSeries 큐 관리자 프록시 이름	MQSeries 큐 관리자의 이름
소유하는 브릿지의 이름	이 MQSeries 큐 관리자를 소유하는 브릿지 오브젝트의 이름
실행 상태	상태: 실행 또는 정지
시작 규칙 클래스	MQSeries 큐 관리자 오브젝트가 시작될 때 사용되는 규칙 클래스
클라이언트 연결 하위	이 프록시가 소유하는 모든 클라이언트 연결 오브젝트의 목록

브릿지 클라이언트 연결 서비스 정의는 MQSeries 큐 관리자에 연결하는 데 필요한 자세한 정보를 보유하고 있습니다. 연결 등록정보는 표19에 나와 있습니다.

표 19. 클라이언트 연결 서비스 등록정보

등록정보	설명
어댑터 클래스	게이트웨이 어댑터로 사용될 클래스
CCSID*	사용될 정수 MQSeries CCSID 값
클래스	브릿지 클라이언트 연결 서비스 클래스
최대 연결 유효 시간	연결이 종료되기 전에 유효 상태로 있을 수 있는 최대 시간
MQSeries 암호*	Java 클라이언트에서 사용할 암호
MQSeries 포트*	Java 클라이언트 클래스를 통해 MQSeries 큐 관리자에 대한 연결을 작성하는 데 사용되는 IP 포트 번호. 지정하지 않으면, MQSeries 큐 관리자가 브릿지와 같은 기계에 있는 것으로 간주되고 Java 바인딩이 사용됩니다.
MQSeries 수신 엑시트 클래스*	클라이언트 채널의 다른 끝에서 쓰이는 수신 엑시트를 일치시키는 데 사용됩니다. 엑시트에는 데이터를 엑시트 코드에 전달할 수 있도록 하기 위한 연관된 문자열이 있습니다.
MQSeries 보안 엑시트 클래스*	클라이언트 채널의 다른 끝에서 쓰이는 보안 엑시트를 일치시키는 데 사용됩니다. 엑시트에는 데이터를 엑시트 코드에 전달할 수 있도록 하기 위한 연관된 문자열이 있습니다.
MQSeries 송신 엑시트 클래스*	클라이언트 채널의 다른 끝에서 쓰이는 송신 엑시트를 일치시키는 데 사용됩니다. 엑시트에는 데이터를 엑시트 코드에 전달할 수 있도록 하기 위한 연관된 문자열이 있습니다.
MQSeries 사용자 ID*	Java 클라이언트에서 사용할 사용자 ID
클라이언트 연결 서비스 이름	MQSeries 기계에서 서버 연결 채널의 이름
소유 큐(owning queue) 관리자 프록시의 이름	소유 큐(owning queue) 관리자 프록시의 이름
시작 규칙 클래스	브릿지 클라이언트 연결 서비스 오브젝트가 시작될 때 사용되는 규칙 클래스
동기 큐 이름	동기화를 위해 브릿지에서 사용되는 MQSeries 큐의 이름
동기 큐 제거자 규칙 클래스	동기 큐에서 메시지가 발견될 때 사용될 규칙 클래스
실행 상태	상태: 실행 또는 정지
소유하는 브릿지의 이름	이 클라이언트 연결을 소유하는 브릿지 오브젝트의 이름
MQ XmitQ 리스너 하위	이 클라이언트 연결을 사용하는 모든 리스너 오브젝트의 목록
*이 매개변수에 대한 세부사항은 <i>MQSeries Using Java</i> 문서에 수록되어 있습니다.	

어댑터 클래스는 MQSeries Everyplace에서 MQSeries로 메시지를 보내기 위해 사용되고, 동기 큐는 이 프로세스의 상태를 추적하는 데 사용됩니다. 해당되는 내용은 보중 메시지를 보장하기 위해 복구 상황에서 사용됩니다. 정상적인 종료 후에는 큐가 비워집니다. 이것은 여러 클라이언트 연결 사이에, 그리고 수신, 송신 및 보안 엑시트가 같은 여러 브릿지 정의에서 공유될 수 있습니다. 이 큐는 또한 사용 중인 리스너 등록정보에 따라 MQSeries에서 MQSeries Everyplace로 이동하는 메시지에 대한 상태를 저장하기 위해서도 사용됩니다. 동기 큐 제거자 규칙 클래스는 메시지가 동기 큐에서 발견될 경우에, MQSeries Everyplace가 메시지 확인에 실패했음을 나타내기 위해 사용됩니다.

MQSeries 시스템에 대한 브릿지 클라이언트 연결 서비스에 의해 관리되는 Java 클라이언트 연결 풀(pool)을 제어하는 데 최대 연결 유휴 시간이 사용됩니다. MQSeries 연결이 사용 부족으로 유휴 상태가 되면 타이머가 시작되고, 연결이 다시 사용되기 전에 그 타이머가 만기되면 그 유휴 연결은 버려집니다. MQSeries 연결의 작성은 확장된 조작으로, 이 프로세스는 그 연결이 과도하게 자원을 소비하지 않고 효율적으로 다시 사용되도록 합니다. 값 0은 연결 풀을 사용하지 않아야 함을 나타냅니다.

메시지를 MQSeries에서 MQSeries Everyplace로 이동하는 리스너 오브젝트의 등록정보는 표20에 나와 있습니다.

표 20. 리스너 등록정보

등록정보	설명
클래스	리스너 클래스
데드 레터 큐(QLQ) 이름	MQSeries에서 MQSeries Everyplace로 전달될 수 없는 메시지를 보유하기 위해 사용되는 큐
리스너 상태 저장 어댑터	상태 정보를 저장하기 위해 사용되는 어댑터의 클래스 이름
리스너 이름	메시지를 제공하는 MQSeries XMIT 큐의 이름
소유하는 클라이언트 연결 서비스 이름	클라이언트 연결 서비스 이름
실행 상태	상태: 실행 또는 정지
시작 규칙 클래스	리스너 오브젝트가 시작될 때 사용되는 규칙 클래스
변환기 클래스	MQSeries 메시지를 MQSeries Everyplace 메시지로 변환하는 것을 판별하는 데 사용되는 규칙 클래스
전달되지 않는 메시지 규칙 클래스	MQSeries에서 MQSeries Everyplace로 메시지가 전달될 수 없을 경우 조치를 판별하기 위해 사용되는 규칙 클래스
메시지를 기다리는 초 수	예외 상황에서 리스너 성능을 제어하기 위해 사용할 수 있는 고급 옵션

전달되지 않는 메시지 규칙 클래스는 메시지가 MQSeries에서 MQSeries Everyplace로 전달될 수 없을 경우에 취할 조치를 결정합니다. 일반적으로, 이 메시지는 MQSeries 시스템의 데드 레터 큐(QLQ)에 위치됩니다.

메시지 보증 전달을 위해, 리스너 클래스는 리스너 상태 저장 어댑터를 사용하여 상태 정보를 MQSeries Everyplace 시스템이나, MQSeries 시스템의 동기 큐에 저장합니다.

브릿지의 구성을 완료하려면, 리모트 큐 관리자와 리모트 큐 정의가 필요합니다. 리모트 MQSeries Everyplace 큐 관리자에 대한 리모트 큐 관리자 정의는 표준 MQSeries Everyplace 실체를 따릅니다. 리모트 MQSeries 큐 관리자에 대한 정의에서는 정상적인 MQSeries Everyplace 동적 채널이 사용되지 않음을 나타내기 위해 채널 정의가 널로 설정됩니다. 그 대신, 위에 설명된 대로 MQSeries 큐 관리자에 대한 연결이 정의됩니다.

MQSeries Everyplace 큐에 대한 리모트 큐 정의도 표준 실체를 따릅니다. 그러나, MQSeries 큐의 경우 MQSeries Everyplace 큐에 대해 사용되는 것과 아주 다릅니다. 표 21에 MQSeries 리모트 큐의 등록정보가 나와 있습니다.

표 21. MQSeries 리모트 큐 등록정보

등록정보	설명
별명	큐에 대한 대체 이름
인증자	널(null)이어야 함
클래스	오브젝트 클래스
클라이언트 연결	사용될 클라이언트 연결 서비스의 이름
압축기	널(null)이어야 함
암호화기	널(null)이어야 함
만기	변환기에 전달됨
최대 메시지 크기	규칙 클래스에 전달됨
모드	동기여야 함
MQ 큐 관리자 프록시	메시지가 처음 송신되어야 하는 MQSeries 큐 관리자의 이름
MQSeries 브릿지	MQSeries로 메시지를 전달할 브릿지의 이름
이름	리모트 MQSeries 큐가 MQSeries Everyplace에 알려지는 이름
소유 큐(owning queue) 관리자	정의를 소유하는 큐 관리자
우선순위	메시지에 사용될 우선순위(메시지 값에 의해 대체되지 않은 경우)
리모트 MQSeries 큐 이름	리모트 MQSeries 큐의 이름
규칙	큐 조작에 사용되는 규칙 클래스

표 21. MQSeries 리모트 큐 등록정보 (계속)

등록정보	설명
큐 관리자 목표	큐를 소유하는 MQSeries 큐 관리자
변환기	메시지를 MQSeries Everyplace 형식에서 MQSeries 형식으로 변환하는 변환기 클래스의 이름
유형	MQSeries 브릿지 큐

cryptor, 인증자 및 압축기 클래스는 큐에 전달되는 메시지에 대한 보안 레벨을 나타내는 큐 속성 세트를 정의합니다. MQSeries Everyplace에 송신된 때부터 그 메시지가 MQSeries 브릿지 큐에 전달될 때까지, 메시지는 최소한 큐 레벨 보안으로 보호됩니다. 이 보안 레벨은 MQSeries 브릿지 큐가 메시지를 MQSeries 시스템에 전달할 때 적용될 수 없어서, 클라이언트 연결에서의 보안 송신 및 수신 엑시트가 전송 동안 사용됩니다. 큐 레벨 보안이 유지되는지는 점검하지 않습니다.

MQSeries 브릿지 큐는 동기식입니다. 그러므로, 비동기 응용프로그램은 MQSeries Everyplace 저장 및 포워드/홈 서버 큐를 통하거나, 비동기 리모트 큐 정의를 통해 이 큐에 메시지를 보냅니다.

게이트웨이 관리는 보통의 MQSeries Everyplace 큐 관리자를 관리하는 것과 같은 방식으로 처리됩니다(관리 메시지를 사용). 관리되는 오브젝트에 적절하게 새로운 메시지 클래스가 정의됩니다. 30 페이지의 표11에 게이트웨이 관리 메시지 클래스가 나와 있습니다.

메시지 변환

MQSeries Everyplace 메시지는 브릿지를 통해 MQSeries에 전달되어 기본 변환기나 목표 큐에 고유한 변환기를 사용하여 MQSeries 형식으로 변환됩니다. 사용자 정의 변환기는 많은 융통성을 제공합니다. 예를 들어, MQSeries Everyplace 메시지 오브젝트 클래스의 서브클래스를 사용하여 MQSeries Everyplace 네트워크에서 특정 유형의 메시지를 나타내는 연습을 해보는 것이 좋습니다. 게이트웨이에서, 변환기는 특정 데이터를 추가하는 것 뿐만 아니라, 적당한 MQSeries 값과 임의의 필드간 매핑을 사용하여 메시지를 MQSeries 형식으로 변환함으로써 특정 데이터를 추가하여 서브클래스의 중요성을 나타낼 수도 있습니다.

MQSeries Everyplace에서 MQSeries로의 기본 변환기는 서브클래스 정보를 이용할 수 없지만, 광범위한 상황에서 유용하도록 설계되었습니다. 이 변환기는 다음과 같은 특성을 가지고 있습니다.

- **MQSeries Everyplace에서 MQSeries로의 메시지 흐름:**

MQSeries Everyplace에서 MQSeries로의 기본 변환기는 MQeMQMsgObject 클래스와 함께 작동합니다. 이 클래스는 MQ 메시지 헤더에서 찾을 수 있는 모든 필드의 표시입니다. MQeMQMsgObject를 사용할 때, 응용프로그램은 set() 메소드를 사용하여 값(우선순위 등)을 설정할 수 있습니다. 그러므로, MQeMQMsgObject(또는 MQeMQMsgObject 클래스에서 도출된 오브젝트)가 기본 MQSeries Everyplace 변환기를 통해 전달될 경우, 기본 변환기(MQeBaseTransformer)는 MQeMQMsgObject 내에서 값을 확보하고, MQSeries 메시지에서 해당되는 값을 설정합니다(예를 들어, 우선순위 값은 MQSeries 메시지에 복사됩니다).

처리되는 메시지가 MQeMQMsgObject가 아니며, MQeMQMsgObject 클래스에서 도출되지 않은 경우, 전체 MQSeries Everyplace 메시지는 MQSeries 메시지의 본문에 복사(*funnel*)됩니다. MQSeries 메시지 헤더의 메시지 형식 필드는 MQSeries 메시지가 MQSeries Everyplace "funneled" 형식으로 보유됨을 나타내도록 설정됩니다.

- **MQSeries에서 MQSeries Everyplace로의 메시지 흐름:**

MQSeries Everyplace에 대한 MQSeries 메시지는 반대 방향으로 이동하는 메시지들과 유사하게 처리됩니다. 기본 변환기는 MQSeries 헤더의 메시지 유형 필드를 검수하여 각각에 맞게 작동합니다.

MQSeries 헤더가 "funneled" MQSeries Everyplace 메시지를 나타낼 경우, MQSeries 메시지 본문은 나중에 MQSeries Everyplace 네트워크로 이동되는 원래 MQSeries Everyplace 메시지로 재구성됩니다.

메시지가 "funneled" MQSeries Everyplace 메시지가 아니면, MQSeries 메시지 헤더 내용이 추출되어 MQeMQMsgObject 오브젝트에 놓입니다. MQSeries 메시지 본문은 간단한 바이트 필드로 처리되고, MQeMQMsgObject 오브젝트에도 놓입니다. 그리고 나서, MQeMQMsgObject는 MQSeries Everyplace 네트워크로 이송됩니다.

이 MQeMQMMsgObject 클래스와 기본 변환기 행동양식(behavior)의 의미는 다음과 같습니다.

- MQSeries Everyplace 메시지가 MQSeries 네트워크를 거쳐 변경 없이 MQSeries Everyplace 네트워크로 이동할 수 있습니다.
- MQSeries 메시지가 MQSeries Everyplace 네트워크를 거쳐 변경 없이 MQSeries 네트워크로 이동할 수 있습니다.
- MQSeries Everyplace 응용프로그램은 MQSeries 응용프로그램은 변경하지 않고 기존의 MQSeries 응용프로그램을 구동할 수 있습니다.

기능

MQSeries 리모트 큐는 MQSeries Everyplace 큐 관리자로부터의 동기 MQSeries Everyplace 넣기(put) 메시징 조작에 사용할 수 있습니다. 다른 모든 메시징 조작은 비동기여야 합니다.

MQSeries Everyplace 관리 메시지는 MQSeries 큐 관리자에게 보낼 수 없습니다. AdminQ가 없어서 관리 메시지 형식이 MQSeries에서 사용되는 형식과 다릅니다.

호환성

MQSeries Everyplace 네트워크는 MQSeries와 독립적으로 존재할 수 있지만, 여러 상황에서 두 제품이 함께 응용프로그램 요구사항을 만족시켜야 합니다. MQSeries Everyplace는 아래에 요약된 측면을 포함하는 호환성으로 기존 MQSeries 네트워크에 통합할 수 있습니다.

- 주소 지정 및 이름 지정:
 - 큐 관리자/큐 주소를 사용하는 동일한 주소 지정 시멘틱
 - ASCII 이름 공간의 공통 사용.
- 응용프로그램: MQSeries Everyplace는 응용프로그램 변경 없이 기존의 MQSeries 응용프로그램을 지원할 수 있습니다.
- 채널: MQSeries Everyplace 게이트웨이는 MQSeries 클라이언트 채널을 사용합니다.

- 메시지 교환 및 내용:
 - MQSeries Everyplace 및 MQS 사이의 메시지 교환
 - 메시지 네트워크 비가시성(MQSeries Everyplace나 MQSeries로부터의 메시지는 변경 없이 상대 네트워크를 통과할 수 있음)
 - MQSeries 메시지 헤더에서 식별된 필드에 대한 상호 지원
 - 메시지 1회 보증 전달
- 샘플 응용프로그램: MQSeries Postcard 및 MQSeries Everyplace Postcard¹¹
 응용프로그램의 상호 작동성

MQSeries Everyplace 버전 1은 MQSeries의 모든 기능을 지원하지는 않습니다. 환경, 운영 체제 및 통신 고려사항 외에 더욱 중요한 차이점이 아래에 자세히 설명되어 있습니다. 그러나, MQSeries Everyplace 내에서 MQSeries Everyplace 기능을 사용하는 대체 수단이나 서브클래스 이용, 제공된 클래스의 대체, 또는 규칙, 인터페이스 및 기타 제품에 내장된 조정 기능을 이용하여 많은 응용프로그램 작업을 수행할 수 있습니다.

- 클러스터링 지원 없음
- 분배 리스트 지원 없음
- 그룹화/세그먼트화된 메시지 없음
- 로드 밸런스/윅 대기 기능 없음
- 참조 메시지 없음
- 보고 옵션 없음
- 공유 큐 지원 없음
- 트리거링 없음
- 작업 단위 지원 및 XA 조정 없음

확장성과 성능 특성은 다릅니다.

11. 이 응용프로그램은 버전 1.0에 포함되지 않지만, 웹(<http://www-4.ibm.com/software/ts/MQSeries/>)의 MQSeries 소프트웨어 다운로드 사이트에서 사용할 수 있습니다.

보증 전달

MQSeries Everyplace 및 MQSeries 둘 다 보증 전달을 제공하지만, 각각 다른 레벨의 보증입니다. 메시지가 MQSeries Everyplace에서 MQSeries로 이동할 때, 메시지 이송은 `putMessage` 및 `confirmPutMessage` 조합이 사용될 경우에만 보증됩니다(28 페이지의 『큐 관리자 조작』 참조). 메시지가 MQSeries에서 MQSeries Everyplace로 이동할 때, 메시지 이송은 MQSeries 메시지가 영구적인 것으로 정의된 경우에만 보증됩니다.

제7장 응용프로그램 및 유틸리티

주: 이 응용프로그램과 유틸리티는 버전 1.0에 포함되지 않지만, WWW의 MQSeries 소프트웨어 다운로드 사이트(<http://www-4.ibm.com/software/ts/MQSeries/>)에서 사용할 수 있습니다.

Postcard

Postcard는 독립형 MQSeries Everyplace 네트워크의 작동이나 MQSeries Everyplace와 MQSeries 네트워크간 상호 작동의 유효성을 확인하는 데 사용할 수 있는 MQSeries Everyplace 응용프로그램입니다. Postcard는 리모트 큐 관리자에서 사용자에게 텍스트 메시지가 송신될 수 있도록 하는 Java 응용프로그램입니다. 이것은 MQSeries 대상에 송신된 Postcard 메시지로 인해 Postcard가 수신되거나, 그 반대로 수행되는 것과 같이, Postcard와 상호 작동합니다.

PalmOS에서 실행되고 Java 버전과 상호 작동될 수 있는 MQSeries Everyplace Postcard 응용프로그램의 C 버전도 있습니다.

MQSeries Everyplace 탐색기

MQSeries Everyplace 탐색기는 로컬 및 리모트 큐 관리자, 큐, 메시지의 구성과 탐색에 쓰이는 Java로 작성된 관리 도구입니다. 이것은 Microsoft® 기초 클래스들을 사용하여 표준 Windows 그래픽 사용자 인터페이스를 나타내며, 그 결과로 Windows 이외 플랫폼에서는 실행할 수 없습니다. 그러나, 관리 메시지들의 송수신에서 배타적으로 작동하므로 모든 MQSeries Everyplace 큐 관리자를 관리하는 데 사용할 수 있습니다. 이것은 MQSeries Everyplace 네트워크의 분할창이 두 개인 보기를 나타냅니다. 왼쪽 분할창에는 오브젝트 트리 보기가 있고 오른쪽 분할창에는 오브젝트 세부 목록 보기가 있습니다.

MQSeries Everyplace 탐색기에는 다음과 같은 기능이 있습니다.

- 큐 관리자 등록정보 표시 또는 수정
- 연결 작성, 삭제 또는 수정 및 해당 등록정보 표시

- 큐 작성, 삭제 또는 수정 및 해당 등록정보와 내용 표시
- 메시지 찾아보기 또는 삭제, 해당 등록정보 표시 및 해당 필드 검수
- 테스트 메시지 송신
- MQSeries에 대한 MQSeries Everyplace 브릿지 구성

MQSeries Everyplace 탐색기는 일반적으로 이미 구성된 큐 관리자를 사용하며 실행을 위해 다른 클래스를 로드할 수 있습니다. 그러한 큐 관리자가 전혀 없으면, 사용자가 선택한 특성으로 이를 작성합니다.

도구의 여러 사본을 각각 고유 JVM으로 하나의 기계에서 실행할 수 있습니다. 이러한 배열로 MQSeries Everyplace 네트워크의 시뮬레이션이 가능하며 MQSeries Everyplace 네트워킹 및 조작을 조사하고 데모를 보기 위해 사용됩니다.

제8장 프로그래밍 인터페이스

MQSeries Everyplace 시스템 프로그래밍 인터페이스(SPI)는 MQSeries Everyplace에 대한 프로그래밍 인터페이스입니다. 운영 체제에 따라 두 가지로 구현할 수 있습니다. Java 버전은 모든 MQSeries Everyplace 기능에 대한 액세스를 제공하지만, MQSeries Everyplace 버전 1.0의 C 인터페이스는 부속 세트에 대한 액세스만 제공합니다. 클래스, 메소드 및 프로시저에 대한 자세한 내용은 *MQSeries Everyplace Programming Reference*에 제공되어 있고, MQSeries Everyplace 프로그래밍의 예는 *MQSeries Everyplace Programming Guide*에 제공되어 있습니다.

부록. 주의사항

이 정보는 미국에서 제공되는 제품 및 서비스용으로 작성된 것입니다. IBM은 다른 나라에서는 이 책에 기술된 제품, 서비스 또는 기능을 제공하지 않을 수도 있습니다. 자신의 국가에서 현재 사용할 수 있는 제품 및 서비스에 대한 정보는 현지 IBM 대표에게 문의하십시오. 이 책에서 IBM 제품, 프로그램 또는 서비스를 언급했다고 해서 반드시 IBM 제품, 프로그램 또는 서비스만이 사용되어야 함을 의미하지는 않습니다. IBM의 지적 재산권을 침해하지 않는 한, 기능상으로 동등한 제품, 프로그램, 서비스를 대신 사용할 수도 있습니다. 그러나 IBM 외의 제품, 프로그램 또는 서비스의 조작을 평가하고 검증하는 것은 사용자의 책임입니다.

IBM은 이 책에서 다루고 있는 특정 내용에 대해 특허를 보유하고 있거나 현재 출원중일 수 있습니다. 이 책을 제공한다고 해서 특허에 대한 사용권까지 부여하는 것은 아닙니다. 다음 주소로 서면으로 사용권 문의를 할 수 있습니다.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

지적 재산권부

다음 문단은 해당 조항이 현지법에 위배되는 영국이나 기타 국가에는 적용되지 않을 수도 있습니다. IBM은 본 문서를 침해 불가능성, 상업적 이용 가능성, 특정 용도로의 적합성을 비롯하여 어떤 종류의 명시적, 암시적 보증도 수반하지 않고 『현상대로』 제공합니다. 일부 국가에서는 특정 트랜잭션에서 명시적 또는 암시적 보증의 부인을 허용하지 않으므로, 본 문장은 여러분에게 적용되지 않을 수도 있습니다.

본 정보에는 기술상의 부정확성이나 인쇄상의 오류가 있을 수 있습니다. 정보는 주기적으로 변경됩니다. 책의 새 판에는 해당 변경사항이 반영됩니다. IBM은 언제든지 통보없이 이 정보에 기술된 제품 및/또는 프로그램을 변경하거나 개선할 수 있습니다.

본 정보에서 IBM 외의 웹 사이트를 언급한 것은 편의를 위한 것일 뿐이며 어떤 방식으로든 해당 웹 사이트를 후원하는 것이 아닙니다. 해당 웹 사이트의 자료는 본 IBM 제품 자료의 일부가 아니므로, 해당 웹 사이트 사용시 위험은 여러분 스스로 감수해야 합니다.

IBM은 여러분의 권리를 침해하지 않는 범위내에서 적절하다고 생각하는 방식으로 여러분이 제공한 정보를 사용하거나 배포할 수 있습니다.

(i) 독립적으로 작성된 프로그램 및 기타 프로그램(이 프로그램 포함) 간의 정보 교환 (ii) 교환된 정보의 상호 이용을 목적으로 정보를 원하는 프로그램 사용권자는 다음 주소로 문의하십시오.

135-270

서울특별시 강남구 도곡동 467-12, 군인공제회관빌딩

한국 아이.비.엠 주식회사

소프트웨어 사업부

해당 용어 및 조건에 관련된 정보는 유료로 사용할 수 있습니다.

이 정보에 기술된 사용권 프로그램 및 사용 가능한 모든 사용권 자료는 IBM이 IBM 고객 계약, IBM 국제 프로그램 사용권 계약 또는 동등한 계약 하에 제공한 것입니다.

등록상표

다음 항목은 IBM사의 등록상표입니다.

AIX AS/400 IBM MQSeries OS/390

Microsoft, Windows, Windows NT 및 Windows 로고는 Microsoft사의 등록상표입니다.

Java 및 모든 Java 기준 등록상표 및 로고는 Sun Microsystems, Inc.의 등록상표입니다.

기타 회사, 제품 및 서비스 이름은 다른 회사의 등록상표 또는 서비스 표시입니다.

용어

이 용어집에서는 이 책에서 사용되는 용어와 일반적인 의미가 아닌 다른 의미로 사용되는 단어들에 대해 설명합니다. 어떤 경우에는 정의가 한 용어에 적용할 수 있는 유일한 것이 아닐 수도 있지만, 그 단어가 이 책에서 사용되는 측면에서는 특수한 의미를 부여합니다.

찾는 용어가 없으면, 색인이나 *IBM Dictionary of Computing*(New York: McGraw-Hill, 1994)을 참조하십시오.

가

개인. 개인 필드는 자체 클래스를 벗어나서는 볼 수 없습니다.

게이트웨이. MQSeries Everyplace 게이트웨이(또는 서버)는 채널 관리자를 포함하여 MQSeries Everyplace 코드를 실행하는 컴퓨터입니다.

공용. 공용 클래스나 인터페이스는 어디에서나 볼 수 있습니다. 공용 메소드나 변수는 해당 클래스를 볼 수 있는 어디에서나 볼 수 있습니다.

다

동기 메시징. 프로그램이 메시지를 메시지 큐에 위치시키는 프로그램들 사이의 통신 방법. 동기 메시징을 사용할 경우, 송신하는 프로그램이 메시지에 대한 응답을 기다린 후 자체 처리를 재개합니다. 비동기 메시징과는 반대입니다.

동적 채널. 동적 채널은 MQSeries Everyplace 장치를 연결하여 양방향 방식으로 동기 및 비동기 메시지와 응답을 전송합니다.

마

메소드. 메소드는 함수 및 프로시저에 대한 오브젝트 지향 프로그래밍 용어입니다.

메시지. 메시지 큐 처리 응용프로그램에서, 메시지는 프로그램 사이에 송신된 통신입니다.

메시지 큐. 큐 참조

메시지 큐 처리. 응용프로그램 내의 각 프로그램이 큐에 메시지를 넣어서 다른 프로그램과 통신하는 프로그래밍 기술.

바

보호. 보호 필드는 자체 클래스, 서브클래스 또는 그 클래스가 일부를 이루고 있는 패키지 내에서만 볼 수 있습니다.

브릿지. MQSeries Everyplace와, MQSeries Everyplace를 포함한 다른 메시징 시스템 사이의 메시지 흐름을 허용하는 MQSeries 오브젝트.

비동기 메시징. 프로그램이 메시지를 메시지 큐에 위치시키는 프로그램들 사이의 통신 방법. 비동기 메시징을 사용할 경우, 송신하는 프로그램이 메시지에 대한 응답을 기다리지 않고 자체적으로 처리를 진행합니다. 동기 메시징과는 반대입니다.

사

서버. (1) MQSeries Everyplace 서버는 MQSeries Everyplace 채널 관리자가 구성된 장치입니다. (2) MQSeries 서버는 리모트 워크스테이션에서 실행되는 클라이언트 응용프로그램에 메시지 큐 처리 서비스를 제공하는 큐 관리자입니다. (3) 더 일반적으로 말하면, 서버는 클라이언트/서버의 특수한 두 프로그램 정보 흐름 모델로 정보에 대한 요청에 응답하는 프로그램입니다. (4) 서버 프로그램이 실행되는 컴퓨터.

서브클래스. 서브클래스는 다른 것을 확장하는 클래스입니다. 서브클래스는 해당되는 슈퍼클래스의 공용 및 보호 메소드와 변수를 계승합니다.

서블릿. 웹 서버에서만 실행되도록 설계된 Java 프로그램.

수퍼클래스. 수퍼클래스는 다른 클래스에 의해 확장되는 클래스입니다. 수퍼클래스의 공용 및 보호 메소드와 변수는 서브클래스에 대해 사용할 수 있습니다.

아

암호화기. 전송 중에 보안을 제공하기 위해 메시지를 암호화하는 프로그램.

압축기. 전송될 데이터의 볼륨을 감소시키기 위해 메시지를 압축하는 프로그램.

오브젝트. (1) Java에서, 오브젝트는 클래스의 인스턴스입니다. 클래스는 어떤 것들의 그룹을 모델화한 것이고, 오브젝트는 그 그룹의 특정 구성원을 모델화한 것입니다. (2) MQSeries에서, 오브젝트는 큐 관리자, 큐 또는 채널입니다.

웹. WWW(World Wide Web)를 참조하십시오.

웹 브라우저. WWW에서 분산되는 정보를 형식화하고 표시하는 프로그램.

응용프로그램 프로그래밍 인터페이스(API). 응용프로그램 프로그래밍 인터페이스는 프로그래머가 응용프로그램에서 사용할 수 있는 함수와 변수들로 구성됩니다.

인스턴스. 인스턴스는 오브젝트입니다. 클래스가 오브젝트를 생성하기 위해 인스턴스화될 경우, 그 오브젝트를 클래스의 인스턴스라고 합니다.

인증자. 메시지의 송신자와 수신자를 확인하는 프로그램.

인터넷. 인터넷은 공유하는 정보의 통합 공용 네트워크입니다. 실제로, 인터넷은 현재 존재하는 모든 공용 원격 통신 네트워크의 모든 자원들의 부속 세트를 사용합니다. 기술적으로, 인터넷을 통합 공용 네트워크로 구별하는 것은 TCP/IP(Transport Control Protocol/Internet Protocol)라고 하는 일련의 프로토콜 세트를 사용하는 데 있습니다.

인터페이스. 인터페이스는 추상 메소드만 포함하고 인스턴스 변수는 없는 클래스입니다. 인터페이스는 여러 개의 서로 다른 클래스들의 서브클래스에 의해 구현될 수 있는 일련의 공통 메소드 세트를 제공합니다.

차

채널. 동적 채널과 MQI 채널을 참조하십시오.

채널 관리자. 끝점 사이에 여러 개의 논리적인 동시 통신 파이프를 지원하는 MQSeries Everyplace 오브젝트.

카

캡슐화. 캡슐화는 오브젝트의 데이터를 개인화하거나 보호되도록 하여 프로그래머가 메소드 호출을 통해서만 그 데이터에 액세스하여 조작할 수 있게 하는 오브젝트 지향 프로그래밍 기술입니다.

큐. 큐는 MQSeries 오브젝트입니다. 메시지 큐 처리 응용프로그램은 메시지를 큐에 넣고, 그 큐에서 메시지를 가져올 수 있습니다.

큐 관리자. 큐 관리자는 응용프로그램에 메시지 큐 처리 서비스를 제공하는 시스템 프로그램입니다.

클라이언트. MQSeries에서, 클라이언트는 지역 사용자 응용프로그램에 대해 서버에서 서비스를 큐에 넣기 위한 액세스를 제공하는 실행시간 구성요소입니다.

클래스. 클래스는 데이터에 대해 작동할 캡슐화된 데이터 및 메소드의 모음입니다. 클래스는 클래스의 인스턴스인 오브젝트를 생성하기 위해 인스턴스화될 수 있습니다.

과

패키지. Java에서의 패키지는 특정 클래스 세트에 대한 Java 코드 액세스 부분을 제공하는 방식입니다. 특정 패키지의 일부인 Java 코드는 패키지에 있는 모든 클래스와, 클래스에 있는 개인적인 아닌 모든 메소드와 필드에 대한 액세스 권한을 갖습니다.

하

하이퍼 텍스트 마크업 언어(HTML). WWW에 표시될 정보를 정의하기 위해 사용되는 언어.

J

JDK(Java Developers Kit). Java 개발자를 위해 Sun Microsystems가 배포하는 소프트웨어 패키지. 여기에는 Java 인터프리터, Java 클래스 및 Java 개발 도구(컴파일러, 디버거, 디셈블러, 애플릿뷰어, 스태브 파일 생성기 및 문서 생성기)가 포함됩니다.

JNDI(Java Naming and Directory Service). Java 프로그래밍 언어로 지정되는 API. 이것은 Java 프로그래밍 언어로 작성된 이름 지정 및 디렉토리 기능을 응용프로그램에 제공합니다.

L

LDAP(Lightweight Directory Access Protocol). LDAP는 디렉토리 서비스를 제공하기 위한 클라이언트-서버 프로토콜입니다.

M

MQI 채널. MQI 채널은 서버 시스템에서 큐 관리자에게 MQSeries 클라이언트를 연결하여 이송 호출과 응답을 양방향 방식으로 이송합니다.

MQSeries. MQSeries는 메시지 큐 처리 서비스를 제공하는 IBM 공인 프로그램입니다.

P

PDA(Personal Digital Addistant). 포켓 크기의 개인 컴퓨터.

T

TCP/IP(Transmission Control Protocol/Internet Protocol). 지역 및 광거리 통신망 둘 다에 대한 피어-투-피어 연결 기능을 지원하는 통신 프로토콜 세트.

W

WWW(World Wide Web)(웹). WWW는 특별히 구성된 서버 컴퓨터가 표준 방식으로 인터넷을 거쳐 문서를 배포할 수 있도록 하는 인터넷 서비스로, 공통 프로토콜 세트를 기반으로 합니다.

참고 문헌

관련 서적:

- *MQSeries Everyplace* 읽어보기, GA30-1080-00
- *MQSeries Everyplace Programming Reference*, SC34-5846-00
- *MQSeries Everyplace Programming Guide*, SC34-5845-00
- *MQSeries An Introduction to Messaging and Queuing*, GC33-0805-01
- *MQSeries for Windows NT V5R1 Quick Beginnings*, GC34-5389-00

색인

[가]

개념, 제품 15
개요 1
개인 레지스트리 46
게이트웨이, MQSeries Everyplace 8, 15
고객 요구사항 12
공용 레지스트리 47
관리 메시지 30
관리 유틸리티 67
구성 49
구성, MQSeries Everyplace 36
규칙, MQSeries Everyplace 49
기능 11

[나]

네트워크, MQS 55
네트워크, MQSeries Everyplace 36, 55

[다]

덤프 데이터 형식 20
덤프 데이터의 형식 20
독자 ix
독자 층 ix
동기 메시징 26, 39
동적 채널 8, 15, 33
등록상표 72

[라]

레지스트리 45
레지스트리, MQSeries Everyplace 15
레지스트리, 개인 46

레지스트리, 공용 47
로컬 큐 21
리모트 큐 22
리스너 오브젝트 60, 61

[마]

메시지 레벨 보안 43
메시지 변환 62
메시지 오브젝트 16
메시지 전달, 보증 66
메시지, 관리 30
메시지, 비동기 38
메시징, MQSeries 5
메시징, 동기 26, 39
메시징, 비동기 26
모니터링 32

[바]

법적 주의사항 71
변환기 62
보급 메시징 6
보안 인터페이스 48
보안, MQSeries Everyplace 39
보안, 메시지 레벨 43
보안, 지역 41
보안, 큐 기반 41
보증 메시지 전달 66
분산 메시징 6
브릿지 오브젝트 57
브릿지, MQS 55
비동기 메시지 38
비동기 메시징 26

[사]

사전 지식 ix
서버, MQSeries 6

설명 1
속성 규칙 49

[아]

암호화 39
압축 39
어댑터, MQSeries Everyplace 34
엔터티, 인증 가능 45
여러 연결 유형 53
연결 유형 51
연결 유형, 여러 53
연결, 클라이언트-서버 52
연결, 피어-투-피어 52
오브젝트, MQSeries Everyplace 16
오브젝트, 메시지 16
요구사항, 고객 12
용어 ix
운영 체제, 지원 3
워크스테이션 메시징 6
유틸리티 67
유틸리티, MQSeries Everyplace 탐색기 67
응용프로그램 67
응용프로그램 로드 53
응용프로그램 MQSeries Everyplace 12
응용프로그램, Postcard 67
응용프로그램, 로드 53
이 책에 관하여 ix
이 책의 사용자 ix
이벤트 로그 35
인증 가능 항목 45
인증서 복제 47
인증서의 복제 47
인터페이스, 보안 48
인터페이스, 프로그래밍 69

[자]

자동 등록 46
장치, MQSeries Everyplace 8, 15
저장 및 포워드 큐 22
전화 접속 연결 관리 35
제반사항 3
제품 개념 15
조작, 큐 관리자 28
조정 49
주의사항, 법적 71
지역 보안 41
지원 운영 체제 3

[차]

채널 관리자 51
채널 리스너 51
채널, 동적 8, 15, 33
채널, 클라이언트 8, 15

[카]

큐 관리자 6, 51
큐 관리자 규칙 50
큐 관리자 조작 28
큐 관리자 프록시 오브젝트 58
큐 관리자, MQSeries Everyplace 26
큐 규칙 50
큐 기반 보안 41
큐, MQSeries Everyplace 21
큐, MQSeries 브릿지 23
큐, 로컬 21
큐, 리모트 22
큐, 저장 및 포워드 22
클라이언트 채널 8, 15
클라이언트, MQSeries 6
클라이언트-서버 연결 52
클래스, MQSeries Everyplace 53

[타]

통신 51

[파]

프로그래밍 인터페이스 69
피어-투-피어 연결 52
필수 운영 체제 3

[하]

호스트 메시징 6
홈 서버 큐 23
홈 서버, MQSeries Everyplace 23
확장성 36

M

Mini Certificate 45
Mini Certificate 발행 서비스 47
Mini Certificate에 대한 보증 서비스 47
MQeAttribute 43
MQeMTrustAttribute 44
MQSeries Everyplace 게이트웨이 8, 15
MQSeries Everyplace 관리 30
MQSeries Everyplace 관리 유틸리티 67
MQSeries Everyplace 구성 36
MQSeries Everyplace 규칙 49
MQSeries Everyplace 네트워크 36, 55
MQSeries Everyplace 레지스트리 15, 45
MQSeries Everyplace 보안 39
MQSeries Everyplace 어댑터 34
MQSeries Everyplace 오브젝트 16
MQSeries Everyplace 유틸리티 67
MQSeries Everyplace 응용프로그램 12, 67
MQSeries Everyplace 장치 8, 15
MQSeries Everyplace 추적 35
MQSeries Everyplace 큐 21
MQSeries Everyplace 큐 관리자 26

MQSeries Everyplace 클래스 53
MQSeries Everyplace 탐색기 유틸리티 67
MQSeries Everyplace와의 관리 30
MQSeries Integrator 5
MQSeries Workflow 5
MQSeries 네트워크 55
MQSeries 메시징 5
MQSeries 브릿지 55
MQSeries 브릿지 규칙 49
MQSeries 브릿지 큐 23
MQSeries 서버 6
MQSeries 제품군 5
MQSeries 클라이언트 6
MQSeries에 대한 인터페이스 55
MQSeries와의 호환성 64
MQSeries, 인터페이스 55
MQSeries, 호환성 64

P

Postcard 응용프로그램 67

R

RAS 다이얼러 규칙 50

S

SPI 69



GA30-1079-00

