

MQSeries Everyplace



Einführung

Version 1

MQSeries Everyplace



Einführung

Version 1

Anmerkung

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die allgemeinen Informationen in „Anhang. Bemerkungen“ auf Seite 71 gelesen werden.

Erste Ausgabe (Juni 2000)

Diese Veröffentlichung ist eine Übersetzung des Handbuchs
MQSeries Everyplace Introduction,
IBM Form GC34-5843-00,

herausgegeben von International Business Machines Corporation, USA

(C) Copyright International Business Machines Corporation 2000

(C) Copyright IBM Deutschland Informationssysteme GmbH 2000

Informationen, die nur für bestimmte Länder Gültigkeit haben und für Deutschland, Österreich und die Schweiz nicht zutreffen, wurden in dieser Veröffentlichung im Originaltext übernommen.

Möglicherweise sind nicht alle in dieser Übersetzung aufgeführten Produkte in Deutschland angekündigt und verfügbar; vor Entscheidungen empfiehlt sich der Kontakt mit der zuständigen IBM Geschäftsstelle.

Änderungen des Textes bleiben vorbehalten.

Herausgegeben von:
SW NLS Center
Kst. 2877
Juni 2000

Inhaltsverzeichnis

Abbildungsverzeichnis	v
Tabellen	vii
Zu diesem Handbuch	ix
Zielgruppe	ix
Vorkenntnisse	ix
In diesem Handbuch verwendete Begriffe	x
Kapitel 1. Übersicht	1
Hinweise zu Version 1.0	1
Kapitel 2. Voraussetzungen	3
Kapitel 3. Die MQSeries-Produktfamilie	5
Kapitel 4. Anforderungen	11
Leistungsspektrum	11
Anwendungen	12
Kundenspezifische Anforderungen	13
Kapitel 5. Produktkonzepte	15
Einführung	15
Nachrichtenobjekte	16
Format der Ausgabedaten	20
Warteschlangen	21
WS-Manager	26
WS-Managervorgänge	28
Verwaltung	30
Verwaltungsnachrichten	30
Selektive Verwaltung	32
Überwachung und zugehörige Vorgänge	33
Dynamische Kanäle	33
Adapter	35
Verwaltung von Einwählverbindungen	35
Trace	35
Ereignisprotokoll	36
MQSeries Everyplace-Netze	36
Konfigurationen und Skalierbarkeit	36
Asynchrone Nachrichtenzustellung	39
Synchrone Nachrichtenzustellung	40
Sicherheit	40
Lokale MQSeries Everyplace-Sicherheit	42
MQSeries Everyplace-Sicherheit auf Warteschlangenebene	42
Sicherheit auf Nachrichtenebene	44
Registrierungsdatenbank	45
Authentifizierbare MQSeries Everyplace-Definitionseinheiten	46

Private Registrierungsdatenbank und Berechtigungen	46
Automatische Registrierung	47
Öffentliche Registrierungsdatenbank und Zertifikatreplikation.	47
Verwendung der Datenbankservices durch Anwendungen	48
Standardservices für die Vergabe von Mini Certificates	48
Sicherheitsschnittstelle	49
Konfiguration und Anpassung	49
Regeln	49
Verbindungsarten	51
Peer-zu-Peer-Verbindung	52
Client/Server-Verbindung	52
Mehrere Verbindungsarten	52
Klassen	53
Anwendungen laden	53
Kapitel 6. MQSeries Everyplace- und MQSeries-Netze	55
Schnittstellen zu MQSeries.	55
Nachrichtenumsetzung	64
Funktion	65
Kompatibilität	65
Gesicherte Zustellung	66
Kapitel 7. Anwendungen und Dienstprogramme	67
PostCard.	67
MQSeries Everyplace Explorer	67
Kapitel 8. Programmierschnittstellen	69
Anhang. Bemerkungen	71
Marken	72
Glossar	73
Literaturverzeichnis.	77
Index	79

Abbildungsverzeichnis

1.	MQSeries-Produktfamilie	5
2.	Einfache Host-Konfiguration und verteilte Konfigurationen	6
3.	Typische Workstation-Konfigurationen	7
4.	Typische Einheitenkonfigurationen	9
5.	Einfache MQSeries Everyplace-Netze	37
6.	MQSeries Everyplace-Sternnetz	38
7.	MQSeries-Brückenobjekthierarchie	56

Tabellen

1.	Version 1 - Unterstützte Softwareumgebungen	3
2.	MQSeries Everyplace- und MQSeries-Komponenten	9
3.	Feldobjekte und ihre Feldmerkmale	17
4.	Attributobjekt - Merkmale	18
5.	Nachrichtenobjekt - Merkmale	18
6.	Nachrichtenobjektfelder, die unterstützt werden	19
7.	Warteschlangenmerkmale	24
8.	Lokale WS-Manager - Merkmale	27
9.	Verbindungsmerkmale (Merkmale ferner WS-Manager).	28
10.	Nachrichtenvorgänge in MQSeries Everyplace-Warteschlangen	29
11.	Verwaltungsnachrichtenklassen	30
12.	Generische Struktur einer Verwaltungsnachricht	31
13.	Unterstützung für Authentifizierung, Verschlüsselung und Komprimierung.	41
14.	Verbindungsarten	51
15.	Klassensoptionen	53
16.	Merkmale des Brückenobjekts	57
17.	Brückenmerkmale	58
18.	MQSeries-Proxy-WS-Manager - Merkmale	58
19.	Client-Verbindungsservice - Merkmale	59
20.	Empfangsprogramm - Merkmale	60
21.	Ferne MQSeries-Warteschlangen - Merkmale	62

Zu diesem Handbuch

Das vorliegende Handbuch enthält eine allgemeine Einführung in MQSeries Everyplace. Themen sind die diesem Produkt zugrundeliegenden Konzepte und das Zusammenspiel mit anderen MQSeries-Produkten.

Ausführliche Informationen zur Anwendungsprogrammierschnittstelle von MQSeries Everyplace und deren Verwendung für die Erstellung von MQSeries Everyplace-Anwendungen finden Sie in den Handbüchern *MQSeries Everyplace Programming Reference* und *MQSeries Everyplace Programming Guide*.

Zielgruppe

Dieses Handbuch richtet sich an alle Benutzer, die Wert auf eine sichere Nachrichtenübertragung bei kleinen, kompakten Einheiten wie Sensoren, Telefonen, PDAs (Personal Digital Assistants) und Laptops legen.

Vorkenntnisse

Zum Verständnis des vorliegenden Handbuchs sind keine Vorkenntnisse erforderlich, Grundkenntnisse über die sichere Nachrichtenübermittlung sind jedoch von Vorteil.

Informationen hierzu finden Sie in den folgenden MQSeries-Handbüchern:

- *MQSeries An Introduction to Messaging and Queuing*
- *MQSeries for Windows NT V5R1 Einstieg*

Diese Handbücher können als Softcopy aus dem Dokumentationsbereich der MQSeries-Online-Bibliothek heruntergeladen werden. Diese finden Sie auf der MQSeries-Website unter <http://www.ibm.com/software/ts/MQSeries/library/>.

In diesem Handbuch verwendete Begriffe

Folgende feststehenden Begriffe werden in diesem Handbuch verwendet:

MQSeries-Produktfamilie

Umfasst die in „Kapitel 3. Die MQSeries-Produktfamilie“ auf Seite 5 beschriebenen MQSeries-Produkte.

MQSeries-Nachrichtenübertragung

Umfasst die vier in „Kapitel 3. Die MQSeries-Produktfamilie“ auf Seite 5 beschriebenen Produktgruppen für Nachrichtenübertragung.

MQSeries

Umfasst die folgenden drei MQSeries-Produktgruppen für Nachrichtenübertragung:

- Verteilte Nachrichtenübertragung
- Host-Nachrichtenübertragung
- Workstation-Nachrichtenübertragung

MQSeries Everyplace

Umfasst die vierte Gruppe der Produktgruppe MQSeries-Nachrichtenübertragung, die universelle (pervasive) Nachrichtenübertragung.

Einheit Ein beliebiger Computer, auf dem MQSeries Everyplace-Programme zur Ausführung kommen, auf dem jedoch *kein* **MQSeries Everyplace-Kanalmanager** oder **MQSeries Everyplace-Kanalempfangsprogramm** installiert ist.

Anmerkung: Im Hinblick auf die Lizenzvergabe entspricht *Einheit* einem *MQSeries Everyplace-Client*.

Gateway

Ein beliebiger Computer, auf dem MQSeries Everyplace-Programme zur Ausführung kommen und auf dem ein **MQSeries Everyplace-Kanalmanager** oder **MQSeries Everyplace-Kanalempfangsprogramm** installiert ist.

Anmerkung: Im Hinblick auf die Lizenzvergabe entspricht *Gateway* einem *MQSeries Everyplace-Server*.

Kapitel 1. Übersicht

MQSeries Everyplace gehört zur MQSeries-Produktfamilie für hochqualitative Nachrichtenübertragung. In dieser Hinsicht entspricht dieses Produkt nicht nur den Anforderungen kleiner Einheiten wie Sensoren, Telefonen, PDAs (Personal Digital Assistant) und Laptops, sondern auch den Anforderungen mobiler Einrichtungen und unzuverlässiger Kommunikationsnetze. Es sorgt für die bei MQSeries übliche einmalige gesicherte Nachrichtenzustellung sowie für den Nachrichtenaustausch mit anderen Produkten aus dieser Familie. Da viele MQSeries Everyplace-Anwendungen außerhalb des Schutzbereiches einer Internet-Firewall zur Ausführung kommen, stellt dieses Produkt darüber hinaus leistungsstarke Sicherheitsfunktionen zur Verfügung.

Kleine Einheiten erfordern ein Nachrichtensubsystem mit geringer Belastung der Systemressourcen; daher wurde MQSeries Everyplace im Hinblick auf die Systembelegung und Protokolleistungsfähigkeit optimiert. Es stellt zwar nicht dasselbe Leistungsspektrum wie die anderen Produkte der MQSeries-Familie zur Verfügung, kann jedoch mühelos mit diesen eingesetzt werden. MQSeries Everyplace unterstützt Mobilität, Roaming, den lokalen und fernen Zugriff auf Nachrichten sowie Sicherheit und Unterstützung für die Nachrichtenübertragung über unzuverlässige Netze.

MQSeries Everyplace gehört zur Produktfamilie IBM Pervasive Computing und ermöglicht daher die problemlose Integration mit anderen IBM Produkten dieser Richtung.

Hinweise zu Version 1.0

- Bei MQSeries Everyplace Version 1.0 handelt es sich um ein Toolkit, das Benutzern die Erstellung von MQSeries Everyplace-Anwendungen sowie einer Umgebung für deren Ausführung ermöglicht.
- In diesem Release ist der Entwickler von Anwendungen bzw. Lösungen für den Einsatz von MQSeries Everyplace auf den entsprechenden Einheiten zuständig.

Kapitel 2. Voraussetzungen

In Tabelle 1 sind die Betriebssysteme aufgeführt, auf denen MQSeries Everyplace Version 1 eingesetzt werden kann.¹

Tabelle 1. Version 1 - Unterstützte Softwareumgebungen

	Betriebssystem
Einheit	EPOC
	Palm OS
	Windows CE
	Windows Windows 95 Windows 98 Windows NT V4 Windows 2000
Gateway	Windows NT V4 Windows 2000
Anmerkungen:	
<ol style="list-style-type: none">1. Version 1.0 wurde in Java erstellt und kann auf allen Plattformen eingesetzt werden, die Java unterstützen.2. Ein Client mit eingeschränktem Funktionsumfang, der lediglich synchronen Zugriff auf ferne Warteschlangen ermöglicht, steht als C-Codebase zur Verfügung und kann nur unter Palm eingesetzt werden.3. Empfohlen wird Java Version 1.1, und zwar das für die jeweilige Plattform aktuelle Release. Ausführliche Informationen zu den bereits getesteten Java-Releases finden Sie auf der MQSeries Everyplace-Website (www.ibm.com/software/mqseries/everypplace).	

1. Der MQSeries Everyplace-Einheitencode kann auf einer beliebigen Einheit mit Java installiert werden, wurde jedoch nur unter den in Tabelle 1 aufgeführten Betriebssystemen getestet.

Kapitel 3. Die MQSeries-Produktfamilie

Zur MQSeries-Produktfamilie gehört eine Vielzahl an Produkten, die die in Abb. 1 dargestellten Merkmale und Funktionen zur Verfügung stellen.

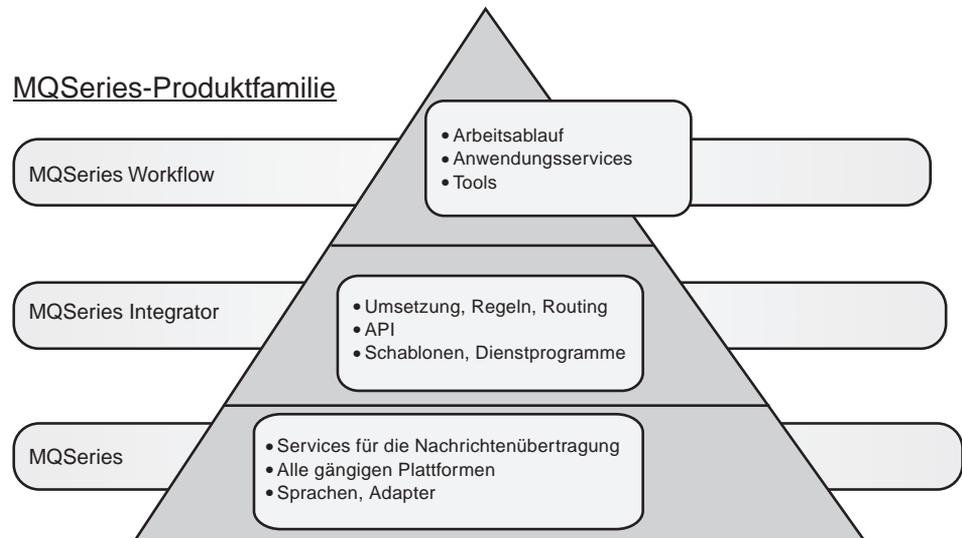


Abbildung 1. MQSeries-Produktfamilie

- **MQSeries Workflow** erleichtert die unternehmensweite Integration durch die Automatisierung von Geschäftsprozessen, an denen Menschen und Anwendungen beteiligt sind.
- **MQSeries Integrator** ist eine leistungsstarke Software für die Nachrichtensteuerung; sie ermöglicht intelligentes, regelbasiertes Nachrichten-Routing in Echtzeit sowie Datenumsetzung und -formatierung.
- **MQSeries-Nachrichtenübertragung (Messaging)** ermöglicht die Konnektivität verschiedener Systeme, vom Desktop bis zum Großrechner, wobei mehr als 35 Plattformen unterstützt werden; die Kommunikation erfolgt über eine hochqualitative Nachrichtenübertragung.

MQSeries Workflow und MQSeries Integrator bieten Konnektivität über die MQSeries-Nachrichtenübertragungsschicht.

Die MQSeries-Nachrichtenübertragung wird sowohl von MQSeries (MQS)- als auch von MQSeries Everyplace-Produkten zur Verfügung gestellt; alle Produkte unterstützen ein oder mehrere Hardware-Server-Plattformen bzw. die entsprechenden Betriebssysteme. Da die einzelnen Plattformen ein unterschiedliches Leistungsspektrum aufweisen, wurden die einzelnen Produkte in Produktgruppen zusammengefasst, denen jeweils ein gemeinsames Konzept zugrundeliegt und die einen bestimmten Bereich abdecken.

Es handelt sich hierbei um vier Produktgruppen:

- **Verteilte Nachrichtenübertragung:** MQSeries for Windows NT, AIX, AS/400, HP-UX, Sun Solaris sowie andere Plattformen
- **Host-Nachrichtenübertragung:** MQSeries for OS/390
- **Workstation-Nachrichtenübertragung:** MQSeries for Windows 3.1, 95, 98
- **Universelle (Pervasive) Nachrichtenübertragung:** MQSeries Everyplace

Die Nachrichtenübertragung selbst erfolgt bei allen Produktgruppen bzw. Produkten über Warteschlangenmanager (WS-Manager). Diese verwalten Warteschlangen, in denen Nachrichten gespeichert werden können. Anwendungen kommunizieren jeweils mit einem lokalen WS-Manager, über den sie Nachrichten in Warteschlangen stellen bzw. aus diesen abrufen. Nachrichten, die in eine ferne, einem fernen WS-Manager zugeordnete Warteschlange eingereicht werden, werden über Kanäle an diesen fernen WS-Manager übertragen. Auf diese Weise erfolgt die Übertragung von Nachrichten an das eigentliche Ziel unter Umständen über mehrere zwischengeschaltete WS-Manager. Ziel der Nachrichtenübertragung ist eine Aufhebung der Abhängigkeit zwischen sender und empfangender Anwendung, indem Nachrichten bei der Übertragung bei Bedarf in Warteschlangen zwischengespeichert werden. Alle MQSeries-Produkte, die auf dieser Nachrichtenübertragung beruhen, arbeiten mit denselben grundlegenden Komponenten, nämlich WS-Managern, Warteschlangen, Nachrichten und Kanälen, wobei jedoch zahlreiche Unterschiede zwischen den einzelnen Produkten bestehen.

MQSeries-Produkte für die Host-Nachrichtenübertragung und die verteilte Nachrichtenübertragung werden zur Unterstützung verschiedener Netzkonfigurationen mit Clients und Servern² eingesetzt. Einige Beispielkonfigurationen sind in Abb. 2 zu sehen.

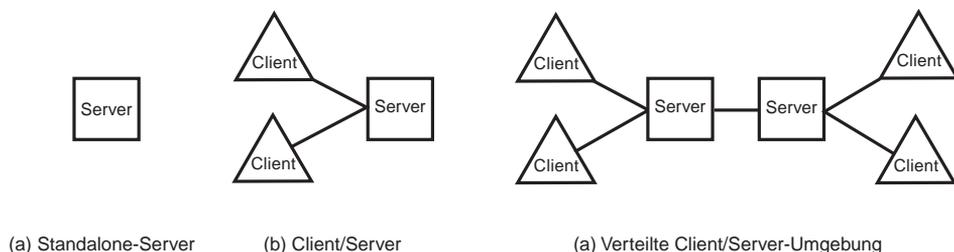


Abbildung 2. Einfache Host-Konfiguration und verteilte Konfigurationen

Bei der einfachsten Konfiguration handelt es sich um einen Standalone-Server mit einem WS-Manager. Auf diesem Server kommen eine oder mehrere Anwendungen zur Ausführung, die Nachrichten über Warteschlangen miteinander austauschen. Eine Alternative ist die Client/Server-Konfiguration. In dieser Konfiguration befindet sich der WS-Manager auf dem Server; die einzelnen Clients greifen auf diesen WS-Manager über einen Client-Kanal zu.

2. Die hier verwendeten Begriffe haben eine bestimmte Bedeutung in MQSeries, die unter Umständen von der üblichen Bedeutung abweicht. Im vorliegenden Handbuch werden diese Begriffe immer im MQSeries-Kontext verwendet.

Bei diesem Client-Kanal handelt es sich um eine bidirektionale Übertragungsverbindung mit einem spezifischen MQSeries-Protokoll, das auf einem Aufrufmechanismus ähnlich dem RPC-Verfahren (Remote Procedure Call) basiert. Die Anwendungen können auf dem Client ausgeführt werden und verfügen über Zugriff auf die Server-Warteschlangen. Ein Vorteil der Client/Server-Konfiguration ist, dass die Client-Nachrichtenübertragung wenig Ressourcen belegt, da sich der WS-Manager auf dem Server befindet. Nachteilig an dieser Konfiguration ist die Tatsache, dass die Clients und der zugehörige Server synchron arbeiten, d. h. die Client-Kanäle müssen jederzeit zur Verfügung stehen.

Ein komplexeres Szenario ist die verteilte Client/Server-Konfiguration mit mehreren Servern. In dieser Konfiguration tauschen die Server Nachrichten über Nachrichtenkanäle aus. Diese Nachrichtenkanäle sind unidirektional und arbeiten mit einem Protokoll, das für den sicheren, asynchronen Austausch von Nachrichtendaten konzipiert ist. Die Clients können die Verarbeitung fortsetzen, auch wenn diese Nachrichtenkanäle nicht verfügbar sind; allerdings ist in diesem Fall kein Nachrichtenaustausch zwischen den Servern möglich.

Die MQSeries-Produkte für Workstation-Nachrichtenübertragung stellen eine Untergruppe dieser Konfigurationsoptionen zur Verfügung. An Stelle von Servern werden Workstations mit einem WS-Manager, jedoch kein Client-Anschluss unterstützt. Allerdings ist über MQSeries-Nachrichtenkanäle die Verbindung von Workstations mit anderen Workstations oder auch Servern möglich. Daher werden Workstations als Mini-Server angesehen und an Stelle von Clients eingesetzt, wenn asynchrone Betriebsmöglichkeiten erforderlich sind.

In Abb. 3 sind zwei typische Workstation-Konfigurationen dargestellt. Bei (b) können die Workstation-Anwendungen unabhängig voneinander auf den Servern und den Clients ausgeführt werden:

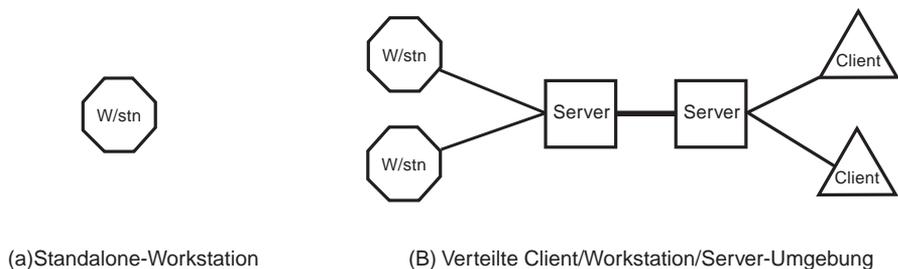


Abbildung 3. Typische Workstation-Konfigurationen

MQSeries Everyplace, das Produkt für universelle (Pervasive) Nachrichtenübertragung, unterstützt Konfigurationen über Einheiten und Gateways.

Bei der MQSeries Everyplace-Einheit handelt es sich um einen Computer mit dem MQSeries Everyplace-Code *ohne einen Kanalmanager*. Einheiten können daher nur jeweils mit einer anderen Einheit oder einem Gateway kommunizieren. Bei MQSeries Everyplace-Einheiten kann es sich um kleinere Einheiten wie z. B. einen Sensor auf einer Ölleitung, um größere Einheiten wie Telefone oder PDAs (Personal Data Assistant) bis hin zu Desktop-Maschinen und Workstations handeln. Einheitencomputer dieser Art werden häufig auch als universelle (Pervasive) Einheiten bezeichnet, obwohl diese Bezeichnung auf Einschränkungen in der Größe und dem Funktionsumfang hindeutet, die bei diesem Produkt nicht gegeben sind.

Bei einem *Gateway* handelt es sich um einen Computer mit dem MQSeries Everyplace-Code, auf dem außerdem *ein MQSeries Everyplace-Kanalmanager oder ein MQSeries Everyplace-Kanalempfangsprogramm konfiguriert ist*. Diese Konfiguration bietet neben dem Leistungsspektrum des Einheitencodes auch die Möglichkeit, mit mehreren Einheiten-Gateways gleichzeitig zu kommunizieren. Darüber hinaus stellen Gateways ein Verfahren für den Nachrichtenaustausch zwischen einem MQSeries Everyplace- und einem MQSeries-Netz zur Verfügung.

In etwa lässt sich sagen, dass Einheiten eine ganze Reihe von Client- und Server-Merkmalen bieten. Sie können für eine vollständige Unterstützung von Warteschlangen konfiguriert werden und ermöglichen damit einen asynchronen Betrieb. Sie verfügen darüber hinaus über Zugriff auf ferne Warteschlangen, was in mancher Hinsicht dem Client-Zugriff auf Server-Warteschlangen entspricht. Im Unterschied zu Servern können Einheiten allerdings nicht mit Clients verbunden werden. Einheiten können über die Peer-zu-Peer-Nachrichtenübertragung miteinander kommunizieren. Darüber hinaus ist eine Kommunikation zwischen Einheiten auch über Kanäle möglich; allerdings handelt es sich hierbei um MQSeries Everyplace-spezifische, so genannte *dynamische Kanäle*; sie unterscheiden sich von den MQSeries-Client- und MQSeries-Nachrichtenkanälen. Dynamische Kanäle sind bidirektional und unterstützen alle von MQSeries Everyplace zur Verfügung gestellten Funktionen, einschließlich der synchronen und asynchronen Nachrichtenübertragung.

Gateways unterstützen die dynamischen MQSeries Everyplace-Kanäle, um eine Kommunikation mit Einheiten zu ermöglichen. Optional unterstützen sie auch MQSeries-Client-Kanäle für die Kommunikation mit Servern. Ähnlich wie Server verfügen auch Gateways über WS-Manager, d. h. sie unterstützen lokale Anwendungen für die Nachrichtenübertragung.

Einige typische Konfigurationen für die universelle (Pervasive) Nachrichtenübertragung sind in Abb. 4 auf Seite 9 zu sehen.

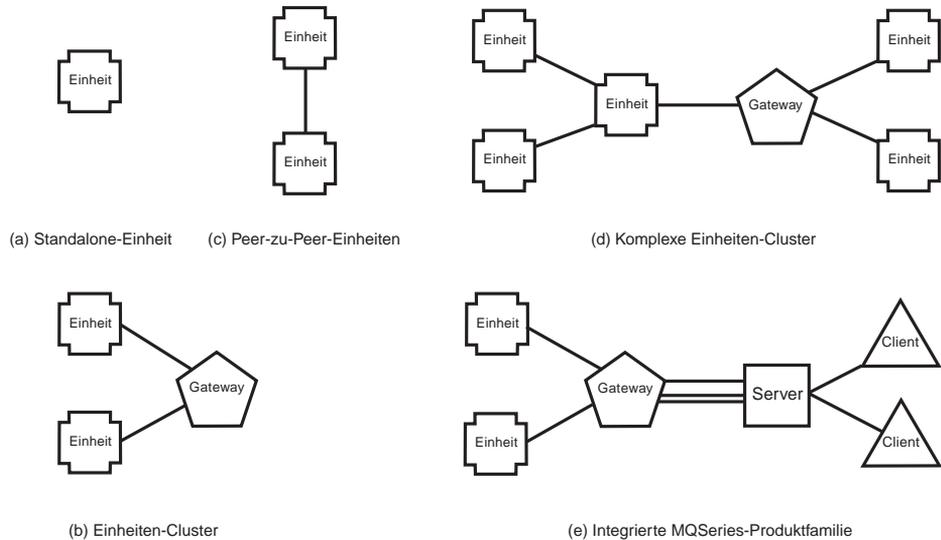


Abbildung 4. Typische Einheitenkonfigurationen

In Abb. 4 (b) wird ein Gateway für die Verbindung mehrerer Einheiten eingesetzt. Ein Gateway-Merkmal ist die Fähigkeit, mehrere gleichzeitig eintreffende Verbindungsanforderungen zu verarbeiten, während Einheiten nur jeweils eine Anforderung bearbeiten können. Dagegen können sowohl Gateways als auch Einheiten mehrere ausgehende Anforderungen gleichzeitig absetzen. Wird in Konfiguration (b) eine Einheit an Stelle eines Gateways eingesetzt, kann nur jeweils eine Terminaleinheit mit dieser zwischen geschalteten Einheit kommunizieren, auch wenn diese die beiden Terminaleinheiten gleichzeitig anruft. In (d) wird sowohl eine Einheit als auch ein Gateway zur Verbindung mehrerer Einheiten eingesetzt. In (e) schließlich verbindet ein Gateway ein Einheiten-netz mit einem MQSeries-Server; in dieser Konfiguration können Nachrichten zwischen allen Netzkomponenten, d. h. Einheiten, Gateways, Servern, Workstations und Clients, ausgetauscht werden. Die wichtigsten Merkmale dieser Komponenten sind in Tabelle 2 aufgeführt.

Tabelle 2. MQSeries Everyplace- und MQSeries-Komponenten

Komponente	Merkmale	Produktgruppe
Einheit (MQSeries Everyplace)	<p>Ermöglicht Anwendungen eine gesicherte Nachrichtenübertragung über dynamische Kanäle:</p> <ul style="list-style-type: none"> Ermöglicht den synchronen Zugriff auf lokale und ferne Warteschlangen Ermöglicht die asynchrone Nachrichtenübertragung an ferne Warteschlangen Kann nur jeweils eine eingehende Anforderung bearbeiten 	Universelle Nachrichtenübertragung

Tabelle 2. MQSeries Everyplace- und MQSeries-Komponenten (Forts.)

Komponente	Merkmale	Produktgruppe
Client (MQS)	<p>Ermöglicht lokalen Anwendungen eine gesicherte Nachrichtenübertragung:</p> <ul style="list-style-type: none"> Erfordert eine synchrone Client-Kanalverbindung zu einem Server Ermöglicht nur den synchronen Zugriff auf Warteschlangen im zugehörigen Server Ermöglicht über den zugehörigen Server die asynchrone Nachrichtenübertragung an ferne Warteschlangen 	Verteilte Host-Nachrichtenübertragung
Gateway (MQSeries Everyplace)	<p>Ermöglicht Anwendungen eine gesicherte Nachrichtenübertragung über dynamische Kanäle:</p> <ul style="list-style-type: none"> Ermöglicht den synchronen Zugriff auf lokale und ferne Warteschlangen Ermöglicht die asynchrone Nachrichtenübertragung an ferne Warteschlangen Kann mehrere gleichzeitig eingehende Anforderungen bearbeiten <p>Unterstützt die Verbindung mehrerer MQSeries-Server über Client-Kanäle</p>	Universelle Nachrichtenübertragung
Server (MQS)	<p>Ermöglicht Anwendungen eine gesicherte Nachrichtenübertragung über Nachrichtenkanäle:</p> <ul style="list-style-type: none"> Ermöglicht den synchronen Zugriff auf lokale Warteschlangen Ermöglicht die asynchrone Nachrichtenübertragung an ferne Warteschlangen <p>Unterstützt die Verbindung mehrerer MQSeries-Clients über Client-Kanäle</p>	Verteilte Host-Nachrichtenübertragung
Workstation (MQS)	<p>Ermöglicht Anwendungen eine gesicherte Nachrichtenübertragung über Nachrichtenkanäle:</p> <ul style="list-style-type: none"> Ermöglicht den synchronen Zugriff auf lokale Warteschlangen Ermöglicht die asynchrone Nachrichtenübertragung an ferne Warteschlangen 	Workstation-Nachrichtenübertragung

Kapitel 4. Anforderungen

Dieses Kapitel setzt sich mit den Anforderungen auseinander, die bei der Entwicklung und Implementierung von MQSeries Everyplace berücksichtigt wurden.

Leistungsspektrum

MQSeries Everyplace erweitert die Nachrichtenübertragungsmöglichkeiten der MQSeries-Produktfamilie wie folgt:

- Unterstützt kleine Einheiten wie PDAs, Telefone und Sensoren, die damit in ein MQSeries-Netz eingebunden werden können. Darüber hinaus unterstützt dieses Produkt auch zwischengeschaltete Einheiten wie Laptops, Workstations und bestimmte verteilte Plattformen. MQSeries Everyplace bietet die allen MQSeries-Produkten eigene Qualität, d. h. die einmalige gesicherte Nachrichtenzustellung, und ermöglicht den Nachrichtenaustausch mit anderen MQSeries-Produkten.
- Stellt umfassende Sicherheitseinrichtungen zum Schutz von Warteschlangen sowie von Nachrichten und zugehörigen Daten (während diese gespeichert sind bzw. übertragen werden) zur Verfügung.
- Ermöglicht einen effizienten Betrieb in unzuverlässigen Übertragungsumgebungen mit instabilen Netzen oder stark eingeschränkter Bandbreite. Es arbeitet mit einem leistungsstarken Übertragungsprotokoll und sorgt bei Ausfällen der Kommunikationsverbindung für eine automatische Wiederherstellung.
- Unterstützt mobile Benutzer, indem es beim Roaming den Wechsel der Netzanschlusspunkte ermöglicht. Darüber hinaus ermöglicht MQSeries Everyplace auch die Steuerung entsprechender Maßnahmen im Fall von schwachen bzw. leeren Batterien und Netzeinschränkungen bzw. -ausfällen.
- Ermöglicht den Einsatz entsprechend konfigurierter Firewalls.
- Erfordert nur einen minimalen Verwaltungsaufwand vom Benutzer; dadurch ist die Installation von MQSeries Everyplace nahezu transparent. MQSeries Everyplace stellt daher eine geeignete Basis für die Erstellung aufgabenspezifischer Anwendungen dar.
- Ermöglicht problemlose Anpassungen und Erweiterungen über anwendungsspezifische Regeln und andere Klassen, über die das Verhalten entsprechend geändert werden kann, oder über die Erweiterung von Basisobjektklassen durch Unterklassen, um beispielsweise verschiedene Nachrichtenarten zu realisieren.

Anwendungen

Bei der Vielfalt möglicher MQSeries Everyplace-Anwendungen ist es nicht möglich, eine definitive Liste aufzustellen; es steht jedoch zu erwarten, dass es sich zum größten Teil um benutzerdefinierte Anwendungen handelt, die für bestimmte Benutzergruppen entwickelt werden. Die folgende Liste bietet eine Übersicht, an welche Art von Anwendungen gedacht wurde:

- **Kundenanwendungen:** Lebensmitteleinkäufe von zu Hause aus über einen PDA, Ermittlungen der Kundenwünsche bei Fluglinien, finanzielle Transaktionen über mobile Telefone.
- **Steueranwendungen:** Erfassung und Integration der Daten von Sensoren auf Ölleitungen über Satellit, Fernsteuerung (z. B. von Ventilen) mit entsprechenden Sicherheitseinrichtungen, die die Berechtigung des Bedieners überprüfen.
- **Mobiler Arbeitseinsatz:** Vertreter (z. B. Versicherungsagenten); schnelle Übermittlung von Empfangsbestätigungen für Paketzustellungsdienste; Kommunikation in einem Schnellimbiss zwischen Angestellten und Küche; Punktezahlung bei Golfturnieren; Nachrichtensysteme für die bei der Polizei eingesetzten mobilen gesicherten Systeme; fachspezifische Informationen für Angestellte bei Versorgungseinrichtungen in Situationen, in denen es zu häufigen Verbindungsabbrüchen kommt, Zählerablesung in Privathaushalten.
- **Anwendungen für den individuellen Einsatz:** Post-/Kalenderreplikation, Datenbankreplikation, beansprucht wenig Platz auf Laptops.

Kundenspezifische Anforderungen

Folgende Anforderungen wurden u. a. bei der Konzipierung von MQSeries Everyplace berücksichtigt:

- **Verwaltung:** Minimaler Installations- und Wartungsaufwand; Unterstützung der lokalen und fernen Verwaltung; anwendungsspezifische Erweiterungs- und Anpassungsmöglichkeiten der Verwaltungsfunktionen; automatische Fehlererkennung und -behebung; unabhängige Verwaltungskomponenten, die gezielt eingesetzt werden können.
- **Kommunikation:** Leistungsstarkes Übertragungsprotokoll; minimale Header; keine obligatorischen Felder (mit Ausnahme einer eindeutigen ID); Möglichkeit zum Ändern der Datencodierung; Unterstützung von Komprimierung, Verschlüsselung und Authentifizierung; Endpunkt-zu-Endpunkt-Festlegung der Komprimierung und der Sicherheitsmerkmale; Kommunikation über eine Firewall hinweg; steckbare Kommunikationsadapter.
- **Kompatibilität:** MQSeries-Qualität und problemloser Nachrichtenaustausch mit anderen MQSeries-Produkten; Kommunikation mit vorhandenen MQSeries-Systemen, ohne dass Änderungen an Anwendungen erforderlich sind; flexible Steuerung des Nachrichtenaustausches zwischen MQSeries und MQSeries Everyplace.
- **Systemspeicherbelegung:** Weit weniger als 100 KB für ein Einheitensystem mit Minimalkonfiguration.
- **Funktionen:** Synchrone und asynchrone Nachrichtenübertragung; Zugriff auf Nachrichten in lokalen oder fernen Warteschlangen; gezielter Abruf einzelner Nachrichtenfelder; gezielte Steuerung des Backup-Datenträgers einer Warteschlange.
- **Regelunterstützung:** Steuerung des Verhaltens über Regeln, z. B. Zeitpunkt der Nachrichtenübertragung, Anzahl der Verbindungsversuche, Handhabung von zu langen Nachrichten, Maßnahmen, wenn eine Zielwarteschlange voll ist.
- **Sicherheit:** Vollständige Unterstützung von Sicherheitseinrichtungen, Authentifizierung und digitalen Unterschriften (Non-Repudiation); Sicherheit auf Nachrichten- und Warteschlangenebene; Schutz des Nachrichtenübertragungssystems vor unberechtigten Zugriffen; "steckbare" Sicherheitseinrichtungen über Algorithmen, die dem Industriestandard entsprechen; Möglichkeit zur Integration der betriebssystemspezifischen Benutzerberechtigungen; kann nationalen Sicherheitsbestimmungen entsprechend konfiguriert werden, wobei sich die Sicherheitsunterstützung bei der grenzüberschreitenden Nachrichtenübertragung jeweils ändert.

Kapitel 5. Produktkonzepte

Einführung

Die grundlegenden Komponenten des MQSeries Everyplace-Programmiermodells sind Nachrichten, Warteschlangen und WS-Manager. Bei MQSeries Everyplace-Nachrichten handelt es sich um Objekte mit anwendungsspezifischem Inhalt. Nachrichten werden in Warteschlangen gespeichert und können auf diese Weise über ein MQSeries Everyplace-Netz übertragen werden. Dabei werden sie unter Angabe des Ziel-WS-Managers und des Warteschlangennamens an die entsprechende Zielwarteschlange übertragen. Anwendungen reihen Nachrichten über Put-Vorgänge in Warteschlangen ein und rufen sie über Get-Vorgänge wieder ab. Bei Warteschlangen kann es sich entweder um lokale oder ferne Warteschlangen handeln, die von WS-Managern verwaltet werden. Für Einheiten und Gateways werden die Konfigurationsdaten in einer Registrierungsdatenbank gespeichert.

Die auf Einheiten ausgeführten Anwendungen können alle auf den Einheiten verfügbaren APIs bzw. Funktionen, nicht nur die MQSeries Everyplace-Programmierschnittstellen, verwenden. Über dynamische Kanäle besteht die Möglichkeit, MQSeries Everyplace-Einheiten mit anderen MQSeries Everyplace-Einheiten bzw. einem MQSeries Everyplace-Gateway zu verbinden.

Die Anwendungen auf Gateways können neben den MQSeries Everyplace-Programmierschnittstellen ebenfalls alle auf den Einheiten verfügbaren APIs bzw. Funktionen verwenden. Über dynamische Kanäle kann ein Gateway mit anderen Gateways bzw. anderen MQSeries Everyplace-Einheiten verbunden werden. Über MQSeries-Client-Kanäle besteht außerdem die Möglichkeit, einen Gateway mit einem oder mehreren MQSeries-Servern (jedoch keinen anderen MQSeries Everyplace-Gateways) zu verbinden. MQSeries Everyplace und MQSeries können zusammen auf einer Maschine installiert werden; die Installation von MQSeries ist jedoch optional.

Das Leistungsspektrum einer Einheit mit vollem Funktionsumfang entspricht bis auf die folgenden Ausnahmen dem eines Gateways:

- Gateways können mehrere gleichzeitig eingehende Anforderungen von anderen Einheiten bzw. Gateways bearbeiten;
- Gateways können mit mehreren MQSeries-Servern gleichzeitig kommunizieren.

Dynamische Kanäle unterstützen folgende Netzverbindungen:

- Einwählverbindungen;
- Feste Verbindungen wie beispielsweise ein herkömmliches LAN, Standleitungen, Infrarot-LANs oder drahtlose LANs.

Die Übertragungsprotokolle werden über mehrere Adapter implementiert, und zwar je ein Adapter pro Protokoll. Dadurch können bei Bedarf jederzeit neue Protokolle hinzugefügt werden, wobei die Speichergröße in einer gegebenen Umgebung jeweils an die vorhandene Konfiguration angepasst wird.

Warteschlangen werden einzeln über weitere Adapter einem Speichermedium zugeordnet. Dies ermöglicht je nach Adapter die Speicherung von Warteschlangen in einem Dateisystem oder im Speicher.

Die MQSeries Everyplace-Programmierschnittstelle ermöglicht die Erstellung von Anwendungen unabhängig vom Speicherort der Warteschlangen. Daher sollte ein Programm, das ursprünglich für den Zugriff auf lokale Warteschlangen konzipiert wurde, ohne weitere Änderungen auf einem fernen WS-Manager eingesetzt werden können (wobei natürlich alle bestehenden Sicherheitsanforderungen berücksichtigt werden müssen und in Kauf genommen werden muss, dass bestimmte MQSeries Everyplace-Vorgänge bei fernen Warteschlangen nicht unterstützt werden). Diese Unabhängigkeit trifft auch für den Einsatz von Verwaltungsfunktionen zu.

Nachrichtenobjekte

Es bestehen grundsätzliche Unterschiede zwischen MQSeries Everyplace-Nachrichtenobjekten und den von MQSeries unterstützten Nachrichten. In MQSeries handelt es sich bei Nachrichten um Bytebereiche, die sich aus Nachrichten-Header und Nachrichtendaten zusammensetzen. Der Nachrichten-Header wird von MQSeries gelesen und enthält wichtige Informationen, wie beispielsweise die Warteschlange und der WS-Manager, an die bzw. den eine Antwortnachricht gerichtet ist, die Nachrichten-ID und die Korrelations-ID; die Nachrichtendaten selbst sind für MQSeries nicht verständlich.

Bei MQSeries Everyplace hingegen handelt es sich bei Nachrichten um *Nachrichtenobjekte*, die vom MQSeries Everyplace-*Feldobjekt* übernommen werden. Bei Nachrichten handelt es sich um echte Objekte, ohne Header oder Nachrichtendaten. Nachrichtenobjekte lassen sich am besten anhand der grundlegenden Feldobjekte erläutern. Diese Feldobjekte werden in MQSeries Everyplace häufig verwendet; es handelt sich dabei um eine Gruppe von Feldern, wobei jedes Feld einen Namen, einen Datentyp und die eigentlichen Daten umfasst. Die Feldnamen sind ASCII-Zeichenfolgen (in denen die Verwendung einiger reservierter Zeichen nicht zulässig ist), für die keine Längenbegrenzung besteht.

Folgende Feldarten sind möglich:

- **ASCII-Zeichenfolge** oder dynamischer ASCII-Zeichenfolgenbereich
- **Boolescher Wert**
- **Byte**, ein fester bzw. dynamischer Bereich mit Bytewerten
- **Doppelgleitkomma**, ein fester oder dynamischer Bereich mit Doppelgleitkommawerten
- **Feldobjekt** oder ein dynamischer Bereich mit Feldobjekten (daher wird die Verschachtelung von Feldobjekten unterstützt)
- **Gleitkomma**, ein fester oder dynamischer Bereich mit Gleitkommawerten
- **Integer** (4 Byte), ein fester oder dynamischer Bereich mit ganzen Zahlen
- **Lange Integer** (8 Byte), ein fester oder dynamischer Bereich mit langen ganzen Zahlen

- **Kurze Integer** (2 Byte), ein fester oder dynamischer Bereich mit kurzen ganzen Zahlen
- **Unicode**-Zeichenfolge oder ein dynamischer Bereich mit Unicode-Zeichenfolgen

Für Feldobjekte ist eine *Feldart* festgelegt, die dem Namen der Programmierobjekt-klasse entspricht. Die "Erben" dieser Objektklasse werden von Anwendungsprogrammen als Nachrichtenobjekte verwendet; anhand der Art der Nachrichtenobjekte wird von MQSeries Everyplace bei Bedarf die richtige Objektklasse erstellt, z. B. nachdem ein Nachrichtenobjekt über einen Kanal übertragen wurde.

Feldobjekte stellen eine Anzahl von Methoden zur Verfügung; beispielsweise können Felder aufgezählt oder ihr Vorhandensein überprüft werden. Ebenso kann durch Vergleich festgestellt werden, ob Feldobjekte identisch sind. Feldobjekte können den Inhalt ihrer Felder in einen Bytebereich ausgeben (Dump) bzw. anhand eines Bytebereichs wiederherstellen; mit Hilfe dieses Merkmals werden beispielsweise die Daten für die Objektübertragung zur Verfügung gestellt und das Objekt nach der Übertragung wiederhergestellt. Diese Ausgabe- und Wiederherstellungsmethoden können außer Kraft gesetzt werden, damit sich Feldobjekte auf andere Weise serialisieren können, um beispielsweise während der Übertragung den Feldinhalt aus einer Datenbank abzurufen. In Tabelle 3 sind die Eigenschaften von Feldobjekten und/oder deren Felder aufgeführt.

Tabelle 3. Feldobjekte und ihre Feldmerkmale

Merkmal	Vorhandensein	
	Feldobjekte	Felder
Zugeordnetes Attributobjekt	Optional	
Zugehörige Felder	Ja	
Verdeckt		Ja
Name		Ja
Art	Ja	Ja
Wert		Ja

Über das Merkmal "Verdeckt" kann ein Feld von Vergleichsvorgängen ausgenommen werden.

Attributobjekte enthalten einen Mechanismus für die Ausführung von Authentifizierungen, Verschlüsselungen und Komprimierungen und können Feldobjekten zugeordnet werden.

- **Authentifizierung:** Ermöglicht die Zugriffssteuerung.
- **Komprimierung:** Reduziert den Speicherbedarf (für die Übertragung und/oder die Speicherung).
- **Verschlüsselung:** Schützt den Objektinhalt bei Ausgabe eines Objekts in einen Bytebereich (und ermöglicht die Wiederherstellung dieses Inhalts).

Attributobjekte sind von grundlegender Bedeutung für das MQSeries Everyplace-Sicherheitsmodell und ermöglichen unter anderem den selektiven Zugriff auf den Objekthalt sowie den Schutz von Daten im Sicherungsspeicher. In Tabelle 4 sind die Merkmale der Attributobjekte aufgeführt. Über den für "Regel" angegebenen Wert kann festgelegt werden, welche Vorgänge zulässig sind.

Tabelle 4. Attributobjekt - Merkmale

Merkmal	Vorhandensein
Verschlüsselungsfunktion	Optional (ist in manchen Fällen erforderlich)
Authentifizierungsfunktion	Optional
Komprimierungsfunktion	Optional
Regel	Optional
Art	Optional

Nachrichtenobjekte werden von Feldobjekten abgeleitet und enthalten eine eindeutige Kennung (UID = Unique Identifier), die von MQSeries Everyplace generiert wird. Diese UID dient als eindeutige Kennzeichnung einer Nachricht und setzt sich aus folgenden Bestandteilen zusammen:

- Dem **Namen** des Ursprungs-WS-Managers (wird vom WS-Manager bei Empfang des Objekts hinzugefügt). Hierbei muss es sich um einen global eindeutigen Namen handeln.
- Dem **Zeitpunkt**, zu dem das Nachrichtenobjekt erstellt wurde (wird bei der Erstellung hinzugefügt).

In Tabelle 5 sind die grundlegenden Merkmale aufgeführt, über die Nachrichtenobjekte neben den Merkmalen verfügen, die sie als Feldobjekte übernehmen.

Tabelle 5. Nachrichtenobjekt - Merkmale

Merkmal	Beschreibung
Msg_OriginQMgr	Der Name des WS-Managers, von dem die Nachricht gesendet wurde
Msg_Time	Der Zeitpunkt, zu dem das Nachrichtenobjekt von der Anwendung erstellt wurde

Aus diesen beiden Merkmalen setzt sich die eindeutige Kennung (UID) des Nachrichtenobjekts zusammen.

Für Nachrichten, die an einen anderen MQSeries Everyplace-WS-Manager gerichtet sind, sind keine weiteren Informationen erforderlich, obwohl in der Regel weitere Felder mit dem Informationsgehalt der Nachrichten mitgesendet werden. Bei Nachrichten handelt es sich in der Regel um die "Erben" der Basisnachrichtenobjektklasse, die je nachdem zusätzliche Felder enthalten. Bei einigen dieser Felder handelt es sich um Felder, die in einer ganzen Reihe von Anwendungen vorkommen, z. B. **Msg_ReplyToQMgr**. Daher werden diese Felder in MQSeries Everyplace unterstützt.

In Tabelle 6 sind die unterstützten Felder aufgeführt.

Tabelle 6. Nachrichtenobjektfelder, die unterstützt werden

Feldname	Verwendung
Msg_CorrelID	Bytefolge, über die eine Antwort der ursprünglichen Nachricht zugeordnet wird.
Msg_ExpireTime	Die Zeitspanne, nach der eine (nicht zugestellte) Nachricht gelöscht wird.
Msg_LockID	Der Schlüssel, der für die Freigabe der Nachricht erforderlich ist.
Msg_MsgID	Wird von der Anwendung für die Zuordnung zur ursprünglichen Nachricht verwendet.
Msg_Priority	Die Priorität der Nachricht.
Msg_ReplyToQ	Der Name der Warteschlange, an die die Antwort gesendet werden soll.
Msg_ReplyToQMgr	Der Name des WS-Managers, an den die Antwort gesendet werden soll.
Msg_Resend	Gibt an, dass es sich um die wiederholte Sendung der Nachricht handelt.
Msg_Style	Unterscheidet Befehle von Anforderungen/Antworten usw.

In allen Fällen steht eine definierte Konstante zur Verfügung, über die der Feldname in einem einzigen Byte übertragen werden kann. Einige Felder werden weitreichender unterstützt. Beispielsweise wirkt sich die Priorität (falls angegeben) auf die Reihenfolge aus, in der die Nachrichten übertragen werden, die Korrelations-ID bewirkt die Indexierung einer Warteschlange für diese Feldwerte, um einen schnellen Abruf zu ermöglichen, das Ablaufintervall bewirkt das Abfließen einer Nachricht usw.

Für Nachrichtenobjekte sind eine Reihe von Methoden definiert, wie beispielsweise die Möglichkeit, die Nachrichten-UID, den Namen des Ursprungs-WS-Managers und den Zeitpunkt der Objekterstellung zu extrahieren. Andere sinnvolle Methoden werden von der Feldobjektklasse geerbt, beispielsweise Methoden für den Abruf und das Setzen von Feldwerten. Von besonderem Interesse ist hier die Methode *dump*, bei der die Objektdaten in eine Bytefolge ausgegeben werden. Die Methode wird von MQSeries Everyplace aufgerufen, wenn eine Nachricht im permanenten Speicher abgelegt oder über einen dynamischen Kanal übertragen werden soll. Dadurch ist das Nachrichtenobjekt selbst für die Darstellung des eigenen Datenwerts verantwortlich; hierfür stehen verschiedene Möglichkeiten zur Auswahl. Beispielsweise kann ein Objekt einfach die Werte der zugehörigen Felder ausgeben oder statt dessen auch eine Datenbank abfragen. Für das Gegenstück hierzu, die Methode *restore*, stehen bei der Wiederherstellung eines Objekts anhand der Ausgabe ähnliche Steuermöglichkeiten zur Verfügung. Verfügt das Nachrichtenobjekt über ein zugeordnetes Attributobjekt, werden die Daten bei der Ausgabe (Dump) authentifiziert, verschlüsselt und komprimiert; analog hierzu werden die Daten bei der Wiederherstellung authentifiziert, entschlüsselt und dekomprimiert.

Bei der Übertragung eines Nachrichtenobjekts wird die zugehörige Klassendatei von MQSeries Everyplace nicht übertragen, um die über die Leitung übertragene Datenmenge möglichst gering zu halten. Daher muss die entsprechende Nachrichtenklasse auf jedem WS-Manager vorhanden sein, auf dem ein Exemplar des Nachrichtenobjekts erstellt werden soll.

Die Standardmethode des Nachrichtenobjekts für die Ausgabe wurde optimiert, um die Länge der generierten Bytefolge möglichst gering zu halten und so eine effizientere Nachrichtenspeicherung und -übertragung zu erreichen.

Format der Ausgabedaten

Bei der Datenausgabe im Standardformat werden die Felder wie folgt codiert:

```
{Länge Kennung Grenze {Daten}} {Länge Kennung Grenze {Daten}} { ... }
```

Dabei gilt:

- *Daten*: Der Datenwert. Ganze Zahlen werden komprimiert und alle führenden Nullen und 'F'-Werte entfernt. Booleschen Werten sind keine Datenbytes zugeordnet.
- *Grenze*: Ein besonderes Byte, das die Grenze zwischen Kennung und dem optionalen Datenelement kennzeichnet. Dieses Byte kennzeichnet darüber hinaus auch den Datenelementtyp.
- *Kennung*: Der Feldname in Form einer ASCII-Bytefolge mit variabler Länge, deren Ende mit einem Endebyte gekennzeichnet ist.
- *Länge*: Die Länge des Datenfeldes. Diese wird durch 1 bis 4 Byte angegeben. Im ersten Byte sind die ersten beiden Bits für Längenangabe des Längenfeldes reserviert. Es werden Längenangaben zwischen 0 und 1.073.741.823 unterstützt.

Daraus ergibt sich ein hochkomprimierter Datenstrom. Eine weitere Komprimierung kann durch die Datenkomprimierung erreicht werden. Die XOR-Komprimierung mit einem vorangehenden Bytestrom sieht vielversprechend aus; da die Felder jedoch variabel sind und sich ihre Reihenfolge ändern kann, kann mit einer einfachen XOR-Verknüpfung nicht immer das gewünschte Ergebnis erreicht werden. In MQSeries Everyplace steht deswegen eine intelligente XOR-Verknüpfung auf Feldebasis zur Verfügung, mit der eine bessere Komprimierung erreicht werden kann.

Warteschlangen

Warteschlangen werden in der Regel zur temporären Speicherung von Nachrichtenobjekten verwendet, die anschließend von Anwendungsprogrammen abgerufen werden. Wie Nachrichten werden auch Warteschlangen aus den Feldobjekten abgeleitet. Anwendungen erhalten keinen direkten Zugriff auf das Warteschlangenobjekt³; stattdessen fungiert der WS-Manager als Mittler zwischen Anwendungsprogrammen und Warteschlangen. Warteschlangen werden über ihren Namen gekennzeichnet, bei dem es sich um eine ASCII-Zeichenfolge mit unbegrenzter Länge handeln kann⁴, der jedoch innerhalb eines bestimmten WS-Managers eindeutig sein muss.

MQSeries Everyplace unterstützt verschiedene Warteschlangenarten:

Lokale Warteschlangen

Lokale Warteschlangen werden von Anwendungen zur sicheren Nachrichtenspeicherung verwendet. Warteschlangen verfügen über einen Nachrichtenspeicher, auf den über eine Adapterklasse, in der Regel die Datenträgeradapterklasse, zugegriffen werden kann. In MQSeries Everyplace steht jedoch eine Speicheradapterklasse zur Verfügung, die die gespeicherten Nachrichten im Speicher hält und damit einen schnelleren Zugriff ermöglicht (allerdings besteht hier im Falle eines Systemabsturzes das Risiko, dass Nachrichten verloren gehen). Durch Erstellung des entsprechenden Adapters können Nachrichten auf Warteschlangenebene an beliebigen Stellen gespeichert werden, z. B. in einer relationalen Datenbank, auf einer beschreibbaren CD-ROM usw. Lokale Warteschlangen können online und offline verwendet werden, d. h. unabhängig davon, ob sie mit einem Netz verbunden sind. Zugriff und Sicherheit werden von der Warteschlange gesteuert; ist die Warteschlange mit einem Netz verbunden, kann der Zugriff von einem fernen WS-Manager erteilt werden, so dass auch andere Anwendungen Nachrichten an die Warteschlangen senden bzw. aus dieser abrufen können. Der Zugriff auf lokale Warteschlangen erfolgt immer synchron.

3. Der direkte Zugriff ist innerhalb einer Warteschlangenregel möglich.

4. Um die Interoperabilität sicherzustellen, sollten die MQSeries-Namenseinschränkungen beachtet werden, u. a. auch die maximale zulässige Länge von 48 Zeichen. Die Länge kann auch durch das von Ihnen verwendete Dateisystem eingeschränkt werden.

Ferne Warteschlangen

Ferne Warteschlangen befinden sich nicht in der lokalen Umgebung; statt dessen ist lokal eine Definition vorhanden, die die tatsächliche Warteschlange und den zugehörigen WS-Manager angibt. Der Zugriff auf ferne Warteschlangen kann synchron oder asynchron erfolgen. Ist eine Definition der fernen Warteschlange lokal vorhanden, legt diese den Zugriffsmodus fest. Andernfalls erfolgt eine Warteschlangenermittlung, bei der die Warteschlangenmerkmale ermittelt und der Zugriffsmodus auf "synchron" gesetzt wird.

Der Zugriff auf *synchrone* Warteschlangen ist nur möglich, wenn eine Netzverbindung mit einem Pfad zum zugeordneten WS-Manager besteht. Besteht keine Netzverbindung, sind keine Einreihungs-, Abruf- und Anzeigevorgänge (Put, Get und Browse) möglich; bei einem versuchten Anzeigevorgang (siehe Tabelle 10 auf Seite 29) wird eine Ausnahmebedingung ausgegeben. Die Zugriffsberechtigungen und Sicherheitsanforderungen für den Zugriff auf die Warteschlange werden von der tatsächlichen Warteschlange des Ziel-WS-Managers gesteuert. Fehlerbehebung oder Wiederholungen beim Senden bzw. Empfangen von Nachrichten müssen von der Anwendung übernommen werden; in diesem Fall ist MQSeries Everyplace nicht mehr für die einmalige gesicherte Zustellung verantwortlich.

Bei *asynchronen* Warteschlangen handelt es sich um Warteschlangen, in die Nachrichten eingereicht, aus denen jedoch keine Nachrichten abgerufen werden können. Ist eine Netzverbindung vorhanden, werden die Nachrichten an den zugeordneten WS-Manager und die Warteschlange gesendet. Ist keine Netzverbindung vorhanden, werden die Nachrichten lokal gespeichert und übertragen, sobald eine Netzverbindung hergestellt wurde. Dies ermöglicht Anwendungen, mit der Warteschlange zu arbeiten, auch wenn die Einheit offline ist; für diese Warteschlangenart ist jedoch ein Nachrichtenspeicher für die temporäre Speicherung der Nachrichten erforderlich.

Warteschlangen zum Speichern und Weiterleiten (Store and Forward)

Diese Warteschlangenart speichert Nachrichten, bis diese an den nächsten (jedoch nicht unbedingt zugeordneten) WS-Manager weitergeleitet werden können. Diese Warteschlangenart wird in der Regel (jedoch nicht unbedingt) auf einem Gateway definiert; die Einheit ruft die Nachrichten ab, sobald eine Netzverbindung besteht. Warteschlangen zum Speichern und Weiterleiten von Nachrichten (Store and Forward) können Nachrichten für eine ganze Reihe von Ziel-WS-Managern enthalten; ebenso kann je eine Warteschlange pro Ziel-WS-Manager definiert werden. Soll eine Nachricht an einen Empfänger gesendet werden, der unter Umständen nicht mit dem Netz verbunden ist, wird die Nachricht an den WS-Manager und die Warteschlange des Empfängers adressiert; der zwischengeschaltete Server stellt fest, dass für den Empfänger keine Netzverbindung besteht und speichert die Nachricht im lokalen Nachrichtenspeicher. Es sind keine Änderungen an der sendenden Anwendung erforderlich, um eine Nachricht an eine mit dem Netz verbundene oder nicht verbundene Zielwarteschlange zu übertragen.

Ausgangs-Server-Warteschlangen

Diese Warteschlangenart befindet sich in der Regel auf einer Einheit (die hin und wieder mit dem Netz verbunden wird) und verweist auf eine Warteschlange zum Speichern und Weiterleiten auf einem WS-Manager, dem so genannten Ausgangs-Server. Sobald eine Netzverbindung für die Einheit hergestellt wird, ruft die Ausgangs-Server-Warteschlange Nachrichten aus dem Ausgangs-Server ab. Nach Abruf der Nachricht aus dem Server überträgt die Warteschlange diese Nachricht durch Aufruf der Methoden **putMessage** und **confirmputMessage** an den lokalen WS-Manager (siehe „WS-Managervorgänge“ auf Seite 28). Der WS-Manager muss dafür sorgen, dass die Nachricht in die richtige lokale Warteschlange eingereicht wird. Die Methode zum Nachrichtenabruf aus dem Server ist unter Umständen effizienter im Hinblick auf den Austausch über das Netz als die Übertragung von Nachrichten durch den Server. Dies liegt daran, dass die Ausgangs-Server-Warteschlange die Empfangsbestätigung der ersten Nachricht als Anforderung für die nächste Nachricht (falls vorhanden) verwendet, während bei der Server-Übertragung für jeden Bestätigungsablauf der Austausch von Anforderung und Antwort erforderlich ist. Für Ausgangs-Server-Warteschlangen wird in der Regel ein Abfrageintervall gesetzt, so dass der Server in regelmäßigen Abständen auf vorhandene Nachrichten abgefragt wird, solange eine Netzverbindung besteht. Bei diesem Abfrageintervall handelt es sich um eine Konfigurationsoption für die Verwaltung.

Verwaltungswarteschlangen

Diese Warteschlangenart empfängt MQSeries Everyplace-Verwaltungsnachrichten. Darüber hinaus kann eine optionale Warteschlange für Antworten auf Verwaltungsnachrichten verwendet werden, die die Antworten auf Verwaltungsnachrichten des MQSeries Everyplace-Systems empfängt. Mit der Ausführung der Verwaltungsvorgänge selbst hat die Verwaltungswarteschlange nichts zu tun, sie überträgt lediglich die Nachrichten, die in verkapselter Form die verwaltungstechnischen Informationen enthalten.

MQSeries-Brückenwarteschlangen

Hierbei handelt es sich um eine Sonderform der fernen Warteschlange, bei der sich die Warteschlangendefinition auf einem Gateway und die Zielwarteschlange auf einem MQSeries-WS-Manager befindet. Diese Warteschlangenart stellt eine Schnittstelle zwischen MQSeries Everyplace- und MQSeries-Umgebungen dar. Eventuell erforderliche Daten- oder Nachrichtenumformatierungen werden von Umsetzungsprogrammen vorgenommen. In MQSeries Everyplace steht ein Umsetzungsprogramm mit den grundlegenden Funktionen zur Verfügung; dieses Programm muss von den Programmierern entsprechend den Gegebenheiten angepasst werden.

MQSeries Everyplace speichert Daten sicher in Warteschlangen und sorgt dafür, dass Nachrichten physisch auf den Datenträger geschrieben, und nicht nur vom Betriebssystem temporär gespeichert werden. MQSeries Everyplace nimmt jedoch keine Protokollierung der Änderungen an individuellen Nachrichten und Warteschlangen vor. Für die Wiederherstellung nach einem Datenträgerausfall sind Hardwarelösungen erforderlich, wie beispielsweise die Verwendung von RAID-Plattensystemen. Ebenso besteht die Möglichkeit, Warteschlangen in einem wiederherstellbaren Speicher, wie beispielsweise in bestimmten Datenbanksystemen, abzubilden.

In MQSeries Everyplace ist es nicht erforderlich, dass ein WS-Manager über definierte Warteschlangen verfügt. Es werden jedoch bei Bedarf vier Systemwarteschlangen unterstützt:

- **AdminQ:** Diese Warteschlange ist für den Empfang von Verwaltungsnachrichten erforderlich.
- **AdminReplyQ:** Diese Warteschlange kann optional für den Empfang von Antworten auf Verwaltungsnachrichten verwendet werden.
- **DeadLetterQ:** In dieser Warteschlange werden unzustellbare Nachrichten gespeichert.
- **SYSTEM.DEFAULT.LOCAL.QUEUE:** Eine Warteschlange mit demselben Namen wie eine obligatorische Systemwarteschlange auf MQSeries-Servern.

In Tabelle 7 sind Warteschlangenmerkmale aufgeführt; allerdings gelten nicht alle Merkmale für alle Warteschlangenarten:

Tabelle 7. Warteschlangenmerkmale

Merkmale	Beschreibung
Admin_Class	Warteschlangenklasse
Admin_Name	Warteschlangenname im ASCII-Format
Queue_Active	Gibt an, dass die Warteschlange aktiv ist
Queue_AttRule	Die Regelklasse, die die sicherheitsspezifischen Vorgänge steuert
Queue_Authenticator	Authentifizierungsfunktionsklasse
Queue_BridgeName	Name der zugeordneten MQ-Brücke
Queue_ClientConnection	Name der Client-Verbindung
Queue_CreationDate	Erstellungsdatum der Warteschlange
Queue_Compressor	Komprimierungsfunktionsklasse
Queue_Cryptor	Verschlüsselungsfunktionsklasse
Queue_CurrentSize	Anzahl der Nachrichten in der Warteschlange
Queue_Description	Unicode-Beschreibung
Queue_Expiry	Ablaufzeit für Nachrichten
Queue_FileDesc	Gibt an, wo die Warteschlange gespeichert ist
Queue_MaxMsgSize	Gibt die maximal zulässige Nachrichtenlänge für die Warteschlange an
Queue_MaxQSize	Gibt die maximal zulässige Anzahl von Nachrichten an

Table 7. Warteschlangenmerkmale (Forts.)

Merkmale	Beschreibung
Queue_Mode	Gibt an, ob der Zugriffsmodus synchron oder asynchron ist
Queue_MQMgr	Der Proxy des MQSeries-WS-Managers
Queue_Priority	Priorität der Nachrichten (wird bei Angabe eines Nachrichtenwertes außer Kraft gesetzt)
Queue_QAliasNameList	Alternative Namen für die Warteschlange
Queue_QMgrName	Name des WS-Managers, der Eigner der eigentlichen Warteschlange ist
Queue_RemoteQName	Name des fernen MQSeries-Feldes
Queue_Rule	Die Regelklasse für Warteschlangenvorgänge
Queue_TargetRegistry	Die Art der Zielregistrierungsdatenbank
Queue_Transporter	Übertragungsfunktionsklasse
Queue_Transformer	Umsetzungsprogrammklasse

Verwaltungsfunktionen werden zum Erstellen und Löschen von Warteschlangen sowie zum Abfragen bzw. Ändern ihrer Merkmale verwendet.

Warteschlangen können nicht nur zum Speichern von Nachrichten eingesetzt werden. Durch Unterklassen erweiterte Warteschlangen können in Szenarios mit Prozesssteuerungsanwendungen eingesetzt werden, beispielsweise zur direkten Steuerung eines Ventils. Dabei würde über eine entsprechende Nachricht das Ventil geöffnet, der Durchfluss geändert werden usw. Andere Warteschlangen wiederum könnten für die Aktualisierung von Tabellenkalkulationen oder die Umsetzung von Text in Sprache eingesetzt werden. Dieses Verfahren bietet den Vorteil, dass die Sicherheitseinrichtungen der Warteschlangen und die gesicherte Nachrichtenübertragung weiterhin verfügbar sind. MQSeries Everyplace würde also nach wie vor die einmalige Nachrichtenzustellung sicherstellen, und eine zugeordnete Authentifizierungs- und Verschlüsselungsfunktion würden für die sichere Nachrichtenübertragung durch berechtigte Sender sorgen. Anwendungen erhielten keinen Zugriff auf die Warteschlange, und es wäre keine Zugriffsberechtigung erforderlich.

WS-Manager

Der MQSeries Everyplace-WS-Manager gestattet Anwendungen den Zugriff auf Nachrichten und Warteschlangen und steuert die vorhandenen Kanäle. In MQSeries Everyplace Version 1.0 kann auf einer einzelnen virtuellen Java-Maschine (JVM = Java Virtual Machine) nur jeweils ein WS-Manager aktiv sein. Sind mehrere JVMs vorhanden, können so viele WS-Manager eingesetzt werden wie JVMs vorhanden sind. WS-Manager werden mit Namen gekennzeichnet, bei denen es sich um systemweit eindeutige Namen handeln muss⁵ und bei denen es sich um eine ASCII-Zeichenfolge mit unbegrenzter Länge handeln kann⁶. WS-Manager können mit oder ohne Unterstützung für das Einreihen in lokale Warteschlangen konfiguriert werden. Alle WS-Manager unterstützen synchrone Nachrichtenzustellungen; WS-Manager mit lokalen Einreihvorgängen unterstützen auch die asynchrone Nachrichtenzustellung.

Zwischen der asynchronen und synchronen Nachrichtenzustellung bestehen erhebliche Unterschiede:

Asynchrone Nachrichtenzustellung:⁷

Bei der asynchronen Nachrichtenzustellung übergibt die Anwendung Nachrichten an MQSeries Everyplace, die anschließend an eine ferne Warteschlange übertragen werden. Unmittelbar nach dem Einreihvorgang (Put) erfolgt eine Rückkehr zur Anwendung. Die Nachricht wird von MQSeries Everyplace bis zu der endgültigen Zustellung temporär gespeichert. Die Übertragung kann über mehrere Zwischenschritte erfolgen, wobei MQSeries Everyplace für die Zustellung verantwortlich ist. Dieser Modus ermöglicht die *einmalige gesicherte Zustellung*. Weitere Informationen hierzu finden Sie unter „Asynchrone Nachrichtenzustellung“ auf Seite 39.

Synchrone Nachrichtenzustellung:

Die synchrone Nachrichtenzustellung wird eingesetzt für die Übertragung an:

- Zielwarteschlangen auf einem MQSeries Everyplace-WS-Manager (über ein MQSeries Everyplace-Netz)
- Zielwarteschlangen auf einem MQSeries-WS-Manager, von dem eine direkte Verbindung zu einem MQSeries Everyplace-Gateway besteht
- Zielwarteschlangen auf einem indirekt verbundenen MQSeries-Server

Bei der synchronen Nachrichtenübertragung übergibt die Anwendung die Nachricht zur Weiterleitung an MQSeries Everyplace. MQSeries Everyplace setzt sich synchron mit der fernen Zielwarteschlange in Verbindung und reiht die Nachricht ein. Im Anschluss an die Zustellung kehrt MQSeries Everyplace zur Anwendung zurück.

5. Dies ist keine Muss-Einschränkung von MQSeries Everyplace oder MQSeries; allerdings können doppelte WS-Managernamen dazu führen, dass Nachrichten an den falschen WS-Manager übertragen werden.

6. Um die Interoperabilität sicherzustellen, sollten die MQSeries-Namenseinschränkungen beachtet werden, u. a. auch die maximal zulässige Länge von 48 Zeichen. Die Länge kann auch durch das von Ihnen verwendete Dateisystem eingeschränkt werden.

7. MQSeries Everyplace unterscheidet nicht zwischen dem permanenten und nicht permanenten Modus, wie dies bei MQSeries der Fall ist, sondern unterstützt nur den permanenten Modus. Bei Bedarf besteht allerdings die Möglichkeit, einen Sicherungsspeicher für Warteschlangen einzusetzen, wobei zugunsten der Wiederherstellung Leistungseinbußen hingenommen werden.

Die Kommunikation mit dem fernen WS-Manager kann unter Umständen MQSeries Everyplace-Routing über eine Vielzahl zwischengeschalteter Einheiten bzw. Gateways erforderlich machen. Weitere Informationen hierzu finden Sie unter „Synchrone Nachrichtenzustellung“ auf Seite 40.

Bei der asynchronen Nachrichtenzustellung übergibt also die lokale Anwendung die Nachricht an MQSeries Everyplace; die Verantwortung für die weitere Übertragung von diesem lokalen WS-Manager aus liegt dann bei MQSeries Everyplace. Daher ist für die sendende bzw. empfangende Anwendung nicht unbedingt eine aktive Netzverbindung erforderlich, d. h. der Zeitpunkt der tatsächlichen Zustellung ist der sendenden Anwendung nicht bekannt. Für die synchrone Nachrichtenzustellung ist eine aktive Netzverbindung erforderlich und die sendende Anwendung weiß, dass die Nachricht an die Warteschlange der Empfängeranwendung übertragen wurde. Sowohl im asynchronen als auch im synchronen Modus muss die Empfängeranwendung nicht verfügbar sein.

Ein lokaler WS-Manager verfügt über Merkmale, die für die lokale Verwaltung von Warteschlangen erforderlich sind. Darüber hinaus benötigt er eine *Verbindungsdefinition* für jeden fernen WS-Manager, mit dem er kommunizieren muss. Daher werden diese Verbindungsdefinitionen auch als *ferne WS-Managerdefinitionen* bezeichnet. Diese Definitionen können entweder alle für die direkte Kommunikation zwischen WS-Managern erforderlichen Informationen (einschließlich einer Netzadresse) oder aber lediglich den Hinweis enthalten, dass es sich um eine indirekte Kommunikation über einen zwischengeschalteten WS-Manager handelt. Im letzteren Fall muss nur der WS-Managername des nächsten Hop angegeben werden.

Die WS-Managermerkmale sind in Tabelle 8 und Tabelle 9 auf Seite 28 aufgeführt.

Tabelle 8. Lokale WS-Manager - Merkmale

Merkmal	Beschreibung
QMgr_ChnlAttrRules	Kanalattributsregeln
QMgr_ChnlTimeout	Kanalzeitlimit
Admin_Class	WS-Managerklasse
QMgr_Description	Unicode-Beschreibung
Admin_Name	Name des WS-Managers
QMgr_Rules	Regelklasse für WS-Managervorgänge

Tabelle 9. Verbindungsmerkmale (Merkmale ferner WS-Manager)

Merkmals	Beschreibung
Con_Adapter	Der Adapterdateideskriptor
Con_AdapterOptions	Adapteroptionen (z. B. Belegungsprotokoll)
Con_AdapterParm	ASCII-Daten für einen Adapter (z. B. Name eines Servlets)
Con_Aliases	Alternative Namen für den WS-Manager
Con_Channel	Die Kanalart für die Verbindung
Con_Description	Unicode-Beschreibung
Queue_QMgrName	Der Eigner der Definition
Admin_Name	Name des WS-Managers

In einer Verbindungsdefinition werden mehrere Adapter unterstützt.

WS-Managervorgänge

WS-Manager unterstützen Vorgänge im Zusammenhang mit der Nachrichtenübertragung und können optional für die Warteschlangenverwaltung eingesetzt werden. Anwendungen erhalten Zugriff auf Nachrichten über Vorgänge, die auf einem WS-Manager ausgeführt werden. Falls kein Filter angegeben ist, wird die erste verfügbare Nachricht aus der Warteschlange abgerufen. Bei einem Filter handelt es sich um ein Feldobjekt, das auf Übereinstimmung überprüft wird; alle Felder in der Nachricht können als Filterkriterien für einen selektiven Abruf verwendet werden. Dem Abruf (Get) kann (wie allen Nachrichtensende- und Empfangsvorgängen) ein Attributobjekt zugeordnet werden, das bei der Codierung und Decodierung einer Nachricht verwendet wird.

In MQSeries Everyplace wird wie in MQSeries beim Abruf die Nachricht gelöscht. Falls eine gesicherte Nachrichtenübertragung zwischen MQSeries Everyplace und der Anwendung erforderlich ist, sollte ein Abrufvorgang mit anschließender Bestätigung verwendet werden. Dabei wird ein Abruf mit einer Bestätigungs-ID (die von der Anwendung ausgewählt wird) abgesetzt; mit diesem Vorgang wird die Nachricht abgerufen, dabei aber in der Warteschlange verdeckt, und nicht sofort gelöscht. Ein anschließender Bestätigungsvorgang unter Angabe der ursprünglichen Nachrichten-UID gibt an, dass der Abruf für die Anwendung erfolgreich war; erst dann wird die Nachricht gelöscht. Schlägt der Abruf fehl, kann die Nachricht wiederhergestellt werden. Einreihungsvorgänge (Put) verlaufen auf ähnliche Weise.

Durch Angabe der UID können Nachrichten aus einer Warteschlange *gelöscht* werden, ohne sie abzurufen.

Sind Lesevorgänge ohne Löschen (Delete) erforderlich, können die Nachrichten in Warteschlangen *angezeigt* (Browse) werden; optional kann dies auch durch Angabe eines Filters erfolgen. Bei diesem Vorgang werden alle Nachrichtenobjekte abgerufen, die mit den Filterkriterien übereinstimmen, sie sind jedoch weiterhin in der Warteschlange vorhanden. *Anzeigen bei gesetzter Sperre* (Browsing under Lock) wird ebenfalls unterstützt. Dabei werden die Nachrichten in der Warteschlange, die mit den Filterkriterien übereinstimmen, gesperrt. Nachrichten können entweder einzeln oder in über einen Filter ausgewählten Gruppen gesperrt werden; vom Sperrvorgang wird eine *Sperr-ID* zurückgegeben. Gesperrte Nachrichten können nur unter Angabe der Sperr-ID abgerufen oder gelöscht werden. Durch Angabe der entsprechenden Option wird bei einem Anzeigevorgang entweder die vollständige Nachricht oder nur die UID zurückgegeben. Anwendungen können eine angegebene Zeit auf den Empfang einer Nachricht in einer Warteschlange *warten* (Wait). Optional können über einen Filter die gewünschten Nachrichten ausgewählt werden; darüber hinaus ist auch die Angabe einer *Bestätigungs-ID* möglich. Ebenso können Anwendungen auch das Auftreten von MQSeries Everyplace-Nachrichtenergebnissen überprüfen (Listen); auch hier ist die Angabe eines Filters möglich. Empfangsprogramme werden beim Eingang von Nachrichten in einer Warteschlange entsprechend informiert.

In Tabelle 10 ist zu sehen, wie die Warteschlangen für Nachrichtenvorgänge aktiviert werden.

Tabelle 10. Nachrichtenvorgänge in MQSeries Everyplace-Warteschlangen

	Lokale Warteschlange	Ferne Warteschlange ¹	
		Synchron	Asynchron
Browse (±Sperre, ±Filter)	Ja	Ja	
Delete	Ja	Ja	
Get (±Filter)	Ja	Ja	
Listen (±Filter)	Ja		
Put	Ja	Ja	Ja
Wait (±Filter)	Ja	Ja	

Anmerkungen:

1. Der synchrone ferne *Wait*-Vorgang wird über ein Abfrageintervall der fernen Warteschlange implementiert, d. h. die tatsächliche Wartezeit entspricht einem Vielfachen des Abfrageintervalls.
2. ¹Die MQSeries Everyplace-Brücke zu MQSeries in MQSeries Everyplace Version 1.0 unterstützt nur den *Put*-Vorgang.

WS-Manager können nach der Initialisierung optional Anwendungen (Klassen) laden; in ähnlicher Weise können sie auch beim Herunterfahren Anwendungen beenden. Von WS-Managern werden zur Anzeige des Status oder von Fehlern entsprechende Ereignisse ausgegeben, die standardmäßig ins Ereignisprotokoll geschrieben werden.

Verwaltung

Die Verwaltung stellt Funktionen zur Konfigurierung und Verwaltung der MQSeries Everyplace-Ressourcen, wie beispielsweise Warteschlangen und Verbindungen, zur Verfügung. Nachrichtenspezifische Funktionen müssen von der Anwendung zur Verfügung gestellt werden. Die Verwaltung wird über eine Schnittstelle aktiviert, die die Erstellung und den Empfang verwaltungsspezifischer Nachrichten handhabt; das zugrundeliegende Konzept ermöglicht eine identische Handhabung der fernen und lokalen Verwaltung. Anforderungen werden an die Verwaltungswarteschlange auf dem Ziel-WS-Manager gesendet; bei Bedarf können Antworten zurückgegeben werden. Jedes lokale oder ferne MQSeries Everyplace-Anwendungsprogramm kann Verwaltungsnachrichten direkt oder indirekt über Hilfemethoden erstellen und verarbeiten. Verwaltungsnachrichten können außerdem indirekt über MQSeries Everyplace Explorer⁸ erstellt werden. Dabei handelt es sich um ein Verwaltungs-Tool, das eine grafische Benutzerschnittstelle (GUI) für die Systemverwaltung zur Verfügung stellt.

Die Verwaltungswarteschlange selbst weiß nichts über die Ausführung der Verwaltungsvorgänge für die einzelnen Ressourcen; diese Informationen sind in den einzelnen Ressourcen und der entsprechenden Verwaltungsnachricht verkapselt.

Verwaltungsnachrichten

Das MQSeries Everyplace-Nachrichtenobjekt kann durch Verwaltungsnachrichten erweitert werden. In Tabelle 11 sind die Nachrichtenklassen aufgeführt, die für die Verwaltung der MQSeries Everyplace-Ressourcen zur Verfügung stehen. Diese grundlegenden Verwaltungsnachrichten können durch Unterklassen erweitert werden, um so die Verwaltung anderer Objekte zu ermöglichen, beispielsweise kann eine andere Warteschlangenart unter Verwendung einer Unterklasse von **MQueQueueAdminMsg** verwaltet werden. Die MQSeries Everyplace-Brücke zu MQSeries verwendet Unterklassen von **MQueAdminMsg** auf diese Weise.

Tabelle 11. Verwaltungsnachrichtenklassen

Verwaltungsnachrichtenklassen	Verwendung
MQueAdminMsg	Eine abstrakte Klasse, die als Basis aller Verwaltungsnachrichten verwendet wird
MQueQueueManagerAdminMsg	Verwaltung von WS-Managern
MQueQueueAdminMsg	Verwaltung lokaler Warteschlangen
MQueRemoteQueueAdminMsg	Verwaltung ferner Warteschlangen
MQueAdminQueueAdminMsg	Verwaltung der Verwaltungswarteschlange
MQueHomeServerQueueAdminMsg	Verwaltung der Ausgangs-Server-Warteschlangen
MQueStoreAndForwardQueueAdminMsg	Verwaltung der Warteschlangen zum Speichern und Weiterleiten
MQueConnectionAdminMsg	Verwaltung der Verbindungen zwischen WS-Managern

8. MQSeries Everyplace Explorer steht in Version 1.0 nicht zur Verfügung, kann jedoch von der MQSeries-Site mit herunterladbarer Software im World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>) heruntergeladen werden.

Tabelle 11. Verwaltungsnachrichtenklassen (Forts.)

Verwaltungsnachrichtenklassen	Verwendung
MQeClientConnectionAdminMsg	Verwaltung eines Brücken-Client-Verbindungsobjekts, über das die Verbindung zu MQS hergestellt wird
MQeListenerAdminMSg	Verwaltung eines Empfangsprozessobjekts für die von der Brücke verwendete Übertragungswarteschlange, mit dem Nachrichten aus MQS abgerufen werden
MQeBridgeAdminMsg	Verwaltung einer Brücke zu MQS
MQeMQBridgesAdminMsg	Verwaltung einer Liste mit MQ-Brücken
MQeMQMgrProxyAdminMsg	Verwaltung der Brückendarstellung eines MQSeries-WS-Managers
MQeMQBridgeQueueAdminMsg	Verwaltung einer MQ-Brückenwarteschlange

Die Struktur einer Verwaltungsnachricht hängt von ihrer Klasse ab, d. h. von der Ressource, die sie verwaltet und dem Vorgang, der für die Ressource ausgeführt wird. Im Allgemeinen haben Verwaltungsnachrichten jedoch die in Tabelle 12 aufgeführte Struktur:

Tabelle 12. Generische Struktur einer Verwaltungsnachricht

Felder der 1. Ebene	Felder der 2. Ebene und darunter	Verwendung
Admin_Action		Erstellen, löschen, abfragen usw.
Admin_Errors		Übergeordnetes Feldobjekt
	Mehrere Felder	Ausführliche Informationen zu jedem Fehler
Admin_MaxAttempts		Gibt an, wie oft der Versuch, einen Verwaltungsvorgang auszuführen, maximal wiederholt werden soll
Admin_Parameters		Übergeordnetes Feldobjekt
	Ressource	Name der zu verwaltenden Ressource
	Mehrere Felder	Ausführliche Parameterdaten für die Nachrichtenklasse und den Vorgang
Admin_Reason		Textnachricht mit der Fehlerursache
Msg_ReplyToQ		Name der Warteschlange, an die die Antwort gesendet werden soll

Tabelle 12. Generische Struktur einer Verwaltungsnachricht (Forts.)

Felder der 1. Ebene	Felder der 2. Ebene und darunter	Verwendung
Msg_ReplyToQMgr		Name des WS-Managers, an den die Antwort gesendet werden soll
Admin_RC		Numerischer Rückkehrcode mit dem Ergebnis
Msg_Style		Befehl oder Anforderung/Antwort
Admin_TargetQMgr		Name des WS-Managers, der der Eigner der Zielressource ist

Es werden drei Arten von Verwaltungsnachrichten unterstützt, und zwar Befehle (Datagramme), die einen Verwaltungsvorgang angeben, für den keine Antwort erforderlich ist, Anforderungen, für die eine Antwort erforderlich ist, und die Antworten selbst. Die Antwort wird auf der Basis einer Kopie der ursprünglichen Nachricht erstellt; vom Sender können für den Empfänger zusätzliche Felder hinzugefügt werden.

Zusätzlich zur Unterstützung der Basisverwaltungsnachrichten werden für den Großteil der Verwaltungsvorgänge Hilfeprogrammklassen zur Verfügung gestellt, in denen die Nachrichtenstruktur und die Interpretation der Antwort verkapselt ist. Von diesen Klassen können optional Benutzerdialoge zur Verfügung gestellt werden, d. h. diese Klassen sind hilfreich für die Erstellung einfacher Verwaltungs-Tools.⁹

Selektive Verwaltung

Der Verwaltungszugriff kann über die Authentifizierungsfunktion für die Warteschlange gesteuert werden. Lokale Anwendungen werden von der Authentifizierungsfunktion als demselben Benutzer zugehörig eingestuft, d. h. der Verwaltungszugriff wird entweder allen Anwendungen oder keiner erteilt. Ferne Verwaltungsanwendungen werden durch den Aufruf der Authentifizierungsfunktion für den Kanal gesteuert; erst dann ist der Austausch von Verwaltungsnachrichten möglich. Eine Unterscheidung zwischen verschiedenen fernen Benutzern ist möglich, die daher einzeln aktiviert bzw. inaktiviert werden können. In jedem Fall wird der Verwaltungszugriff für jeden Benutzer insgesamt aktiviert oder inaktiviert. Ist eine abgestufte Verwaltungssteuerung erforderlich, bei der z. B. bestimmten Verwaltungsbenutzern der Zugriff auf einige Warteschlangen erteilt, auf andere verwehrt werden soll, ist eine zusätzliche Programmierung erforderlich. Eine erweiterte Authentifizierungsfunktion verfügt über eine Übersicht über die den verschiedenen Benutzern zugeordneten Berechtigungen; daher können Verwaltungsnachrichten anhand dieser Berechtigungen verarbeitet werden (siehe "Sicherheit"). Darüber hinaus können auch die den Warteschlangen zugeordneten Regeln in ähnlicher Weise dazu verwendet werden, bestimmte Vorgänge zuzulassen und andere nicht (siehe „Regeln“ auf Seite 49).

9. Diese Klassen stehen in Version 1.0 nicht zur Verfügung, können jedoch von der MQSeries-Site mit herunterladbarer Software im World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>) heruntergeladen werden.

Überwachung und zugehörige Vorgänge

Die Verwaltung geht in der Regel über die bloße Erstellung und Änderung von Objekten hinaus; zur Verwaltung gehören beispielsweise auch die Überwachung des Systemstatus, die Fehlerbehebung, Meldungen an den Bediener, wenn eine Warteschlange nahezu voll ist oder die Ausführung entsprechender Maßnahmen bei Eingang einer Nachricht, die zu groß für die Zielwarteschlange ist. Diese Aufgaben werden in MQSeries Everyplace über Regeln gesteuert, d. h. Klassen, die bei jeder größeren Änderung am Status eines Objekts oder beim Auftreten bestimmter Fehler aufgerufen werden. In MQSeries Everyplace steht eine Standardgruppe mit Regelklassen zur Verfügung, die jedoch durch benutzerspezifische Klassen ersetzt werden können (siehe „Regeln“ auf Seite 49).

Dynamische Kanäle

Die Kommunikation zwischen MQSeries Everyplace und WS-Managern auf Einheiten oder Gateways erfolgt über logische Verbindungen, die so genannten dynamischen Kanäle. Diese unterstützen die bidirektionale Übertragung und werden bei Bedarf vom WS-Manager eingerichtet. Die asynchrone und synchrone Nachrichtenübertragung erfolgt über dieselben Kanäle; hierzu wird ein spezifisches MQSeries Everyplace-Protokoll verwendet. In MQSeries hingegen werden in der Regel für den synchronen Austausch Client-Kanäle, für die bidirektionale asynchrone Nachrichtenübertragung ein Nachrichtenkanalpaar verwendet. Die *Cluster-Nachrichtenkanäle* in MQSeries verfügen über ähnliche Merkmale wie die dynamischen Kanäle in MQSeries Everyplace, es bestehen jedoch auch einige Unterschiede.

Bei einem dynamischen Kanal handelt es sich um die logische Verbindung zwischen zwei WS-Managern, die für das Senden bzw. Empfangen von Daten hergestellt wird. Zwischen denselben Partnern können mehrere Kanäle gleichzeitig vorhanden sein. Diese verfügen über Merkmale, wie beispielsweise Authentifizierung, Verschlüsselung, Komprimierung und das verwendete Übertragungsprotokoll. Diese Merkmale sind "steckbar" (pluggable) (für unterschiedliche Kanäle können unterschiedliche Versionen verwendet werden); daher verfügt jeder Kanal über eigene Serviceattribute:

- **Authentifizierungsfunktion:** Entweder Null, oder ein *Authentifizierungsfunktionsobjekt*, das Benutzer- oder Kanalauthentifizierungen vornehmen kann
- **Kanal:** Die Klasse, die die Übertragungsservices zur Verfügung stellt
- **Komprimierungsfunktion:** Entweder Null, oder ein *Komprimierungsfunktionsobjekt*, das Datenkomprimierungen und -dekomprimierungen vornehmen kann
- **Verschlüsselungsfunktion:** Entweder Null, oder ein *Verschlüsselungsfunktionsobjekt*, das Datenverschlüsselungen und -entschlüsselungen vornehmen kann
- **Ziel:** Das Ziel des Kanals, z. B. SERVER.XYZ.COM

Die Authentifizierungsfunktion wird in der Regel nur beim Einrichten des Kanals verwendet. Die Komprimierungsfunktion kommt bei jedem Austausch zur Anwendung.

Die einfachste Verschlüsselungsfunktion ist **MQeXorCryptor**, die die zu sendenden Daten über eine XOR-Verknüpfung verschlüsselt. Diese Verschlüsselung ist zwar nicht sicher, bewirkt jedoch, dass die Daten nicht anzeigbar sind. Hingegen verwendet **MQe3DESCryptor** die Triple DES-Verschlüsselung. Bei der einfachsten Komprimierungsfunktion handelt es sich um **MQeRleCompressor**, die Daten komprimiert, indem identische Zeichen durch einen Zähler ersetzt werden. In Tabelle 13 auf Seite 41 sind weitere Authentifizierungs-, Komprimierungs- und Verschlüsselungsfunktionen aufgeführt.

Bei der Kanalherstellung werden Protokolladapterspezifikationen verwendet, anhand derer die Verbindungen und Protokolle für einen bestimmten Kanal festgelegt werden. An jedem zwischengeschalteten Knoten werden die Kanaldefinitionen durchsucht, um die für den nächsten Verbindungsabschnitt erforderliche Adresse aufzulösen. Steht keine Definition für die Weiterleitung zur Verfügung, wird der Kanal beendet und alle Nachrichten, die über diesen Kanal gesendet werden, werden an den an diesem Punkt vorhandenen WS-Manager übergeben.

Kanäle sind für Anwendungsprogrammierer und Administratoren nicht direkt sichtbar und werden vom WS-Manager bei Bedarf erstellt. Kanäle verbinden WS-Manager, und ihre Merkmale werden von MQSeries Everyplace je nach den zu übertragenden Informationen vereinbart bzw. neu vereinbart. Bei Übertragungsprogrammen handelt es sich um MQSeries Everyplace-Komponenten, die unter Verwendung der Kanäle eine Kommunikation auf Warteschlangenebene ermöglichen. Auch diese Übertragungsprogramme sind für den Anwendungsprogrammierer oder Administrator nicht sichtbar.

Ist eine gesicherte Nachrichtenübertragung erforderlich, werden die Nachrichten nur einmal von MQSeries Everyplace an die Anwendung übertragen. Dies wird dadurch erreicht, dass die Kopie einer Nachricht auf der Senderseite erst gelöscht wird, nachdem die Nachricht erfolgreich von einem WS-Manager zum anderen übertragen und bestätigt wurde. Wird im Fall eines Kommunikationsfehlers keine Empfangsbestätigung erhalten, kann eine Nachricht noch einmal übertragen werden (einmalige Zustellung ist nicht gleichbedeutend mit einmaliger Übertragung), Nachrichten werden jedoch nicht mehrfach zugestellt.

Adapter

Adapter werden für die Zuordnung von MQSeries Everyplace zu Einheitschnittstellen verwendet. Kanäle verwenden die Protokolladapter, um HTTP, natives TCP/IP sowie weitere Protokolle für die Übertragung zu verwenden. Ähnlich nutzen Warteschlangen die Feldspeicheradapter für den Zugang zu Speichersubsystemen wie beispielsweise den Hauptspeicher oder das Dateisystem. Adapter stellen für MQSeries Everyplace eine Möglichkeit zur Erweiterung der Einheitenunterstützung und zur Unterstützung der Versionskompatibilität zur Verfügung.

Bei dem *Dateideskriptor* handelt es sich um eine Zeichenfolge, über die ein Adapter erkannt, geladen und aktiviert wird.

Verwaltung von Einwählverbindungen

Für die Unterstützung der Netzeinwahl für Einheiten ist das Betriebssystem der Einheit verantwortlich. Versucht MQSeries Everyplace auf einer nicht verbundenen Einheit einen Netzzugriff, z. B. um eine Nachricht zu übertragen, und ist der Netzstapel nicht aktiv, wird RAS (Remote Access Services) vom Betriebssystem selbst initialisiert. In der Regel wird dabei eine Anzeige mit einem Wählverbindungsprofil für den Benutzer geöffnet. Das Betriebssystem übernimmt die Steuerung, bis eine Verbindung hergestellt wurde. Daher muss der Einheitenbenutzer sicherstellen, dass für das Betriebssystem die entsprechenden Wählverbindungsprofile vorhanden sind. Aus diesem Grund ist auf MQSeries Everyplace-Einheiten keine explizite Unterstützung für die Netzeinwahl erforderlich.

Trace

Ein unabhängiges Programm ist für die Trace-Durchführung zuständig. In MQSeries Everyplace sind Aufrufe für den Trace von Informationsmeldungen, Warnungen und Fehlerbedingungen für das System und den Benutzer integriert. Anwendungen können einen Trace direkt aufrufen und neue Nachrichten hinzufügen bzw. vorhandene Trace-Nachrichten ändern. Das zur Verfügung gestellte Trace-Beispielprogramm ermöglicht die Anzeige ausgewählter Nachrichten, deren Druck oder deren Weiterleitung an das Ereignisprotokoll. Sie können andere Trace-Programme mit zusätzlichen Funktionen oder mit anderen Formatierungs- und Zustellungsmöglichkeiten erstellen.

Der Großteil der MQSeries Everyplace-Ausnahmebedingungen wird zur Verarbeitung an die Anwendung übergeben; die Ausnahmebehandlungsroutine der Anwendung kann diese Ausnahmebedingungen an das Trace-Programm weiterleiten.

Ereignisprotokoll

MQSeries Everyplace stellt Ereignisprotokollmechanismen und -schnittstellen zur Verfügung, mit denen Statusinformationen (z. B. "WS-Manager gestartet") protokolliert werden können. Standardmäßig werden die Protokolleinträge in eine Datei geschrieben; es besteht jedoch die Möglichkeit, die Protokolleinträge abzufangen und an ein anderes Ziel weiterzuleiten. Im MQSeries Everyplace-Ereignisprotokoll werden keine Nachrichtendaten protokolliert, und es kann nicht zur Wiederherstellung von Nachrichten oder Warteschlangen verwendet werden.

MQSeries Everyplace-Netze

Bei MQSeries Everyplace-Netzen handelt es sich um verbundene Einheiten und Gateways. Sie können mehrere physische Netze umfassen, zwischen denen Nachrichten ausgetauscht werden. Im Allgemeinen ermöglichen sie den synchronen und asynchronen Zugriff auf Warteschlangen, und zwar mit einem Programmiermodell, das nicht von der tatsächlichen Position der Warteschlange abhängig ist.

Konfigurationen und Skalierbarkeit

In Abb. 5 auf Seite 37 sind einige der gängigsten MQSeries Everyplace-Netztopologien zu sehen. Bei diesen wird davon ausgegangen, dass sie die synchrone und asynchrone Kommunikation unterstützen.

Die einfachste Konfiguration (a) ist eine Standalone-Einheit, die über lokale Warteschlangen die synchrone Kommunikation zwischen Anwendungen unterstützt. Eine interessantere Konfiguration in Form eines Peer-zu-Peer-Netzes stellt (b) dar. In dieser Konfiguration müssen beide Einheiten dasselbe Kommunikationsprotokoll verwenden, und mindestens eine der Einheiten muss über ein Empfangsprogramm verfügen, damit sie auf Kommunikationsanforderungen der anderen Einheit reagieren kann. In dieser einfachen Konfiguration ist eine Kommunikation nur möglich, wenn sich beide Einheiten in demselben Netz befinden. Die asynchrone Nachrichtenübertragung ermöglicht die Ausführung von Anwendungen, auch wenn keine Netzverbindung für die Einheiten besteht; die synchrone Nachrichtenübertragung hingegen ist nur möglich, wenn beide Einheiten mit dem Netz verbunden sind.

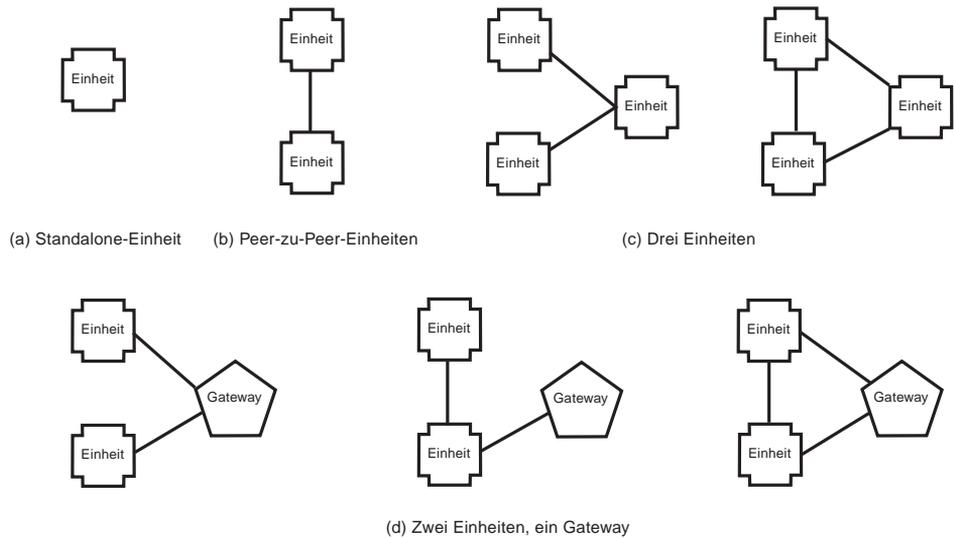


Abbildung 5. Einfache MQSeries Everyplace-Netze

Mehr Möglichkeiten sind bei drei Einheiten gegeben. Dabei können entweder zwei Einheiten über die dritte Einheit verbunden werden, oder alle drei Einheiten werden miteinander verbunden. Sind die Einheiten für den Nachrichtenaustausch auf Peer-zu-Peer-Basis konfiguriert, gelten die oben angegebenen Überlegungen. Größere Flexibilität ermöglicht das Szenario, bei dem die Kommunikation über eine dritte Einheit erfolgt. Die beiden miteinander kommunizierenden Einheiten müssen nicht dasselbe Übertragungsprotokoll verwenden, sie müssen lediglich ein Protokoll verwenden, das von der dritten Einheit, die beide verbindet, verstanden wird. Darüber hinaus ermöglicht die asynchrone Nachrichtenübertragung, dass Nachrichten auch dann ausgetauscht werden können, wenn Sender und Empfänger nicht gleichzeitig mit dem Netz verbunden sind (vorausgesetzt, Sender oder Empfänger sind jeweils zu einem Zeitpunkt mit dem Netz verbunden, zu dem die Verbindungseinheit ebenfalls verfügbar ist). Für die synchrone Kommunikation müssen alle drei Einheiten gleichzeitig verfügbar sein. Diese Konfiguration, bei der die Kommunikation über eine dritte Einheit erfolgt, ist vorteilhaft für Einheiten, die häufig nicht mit dem Netz verbunden sind; allerdings muss die zwischengeschaltete Einheit nach Möglichkeit immer verfügbar sein.

In der Praxis werden Einheiten häufig über ein Gateway miteinander verbunden; drei Konfigurationsmöglichkeiten sind in Abb. 5 auf Seite 37 (d) zu sehen. Der Gateway als Verbindungsknoten hat den Vorteil, dass er mehrere gleichzeitige Verbindungsanforderungen verarbeiten kann. Die erste Konfiguration ist die gängigste; die zweite und dritte Konfiguration sind denkbar, wenn auch etwas ungewöhnlich. Im dritten Szenario kann für einen bestimmten WS-Manager nur eine Route konfiguriert werden, d. h. obwohl augenscheinlich zwei Routes vorhanden sind, darf nur eine ausgewählt werden.

In größeren Netzen können mehrere Gateways eingesetzt werden, von denen jeder mehrere Einheiten unterstützt. Es bestehen verschiedene Möglichkeiten, Gateways miteinander zu verbinden; bei der Definition einer vollständigen Verbindung allerdings sind bei den Routes zwischen Einheiten höchstens zwei Gateways beteiligt. In Abb. 6 ist ein Beispiel für ein großes Netz zu sehen.

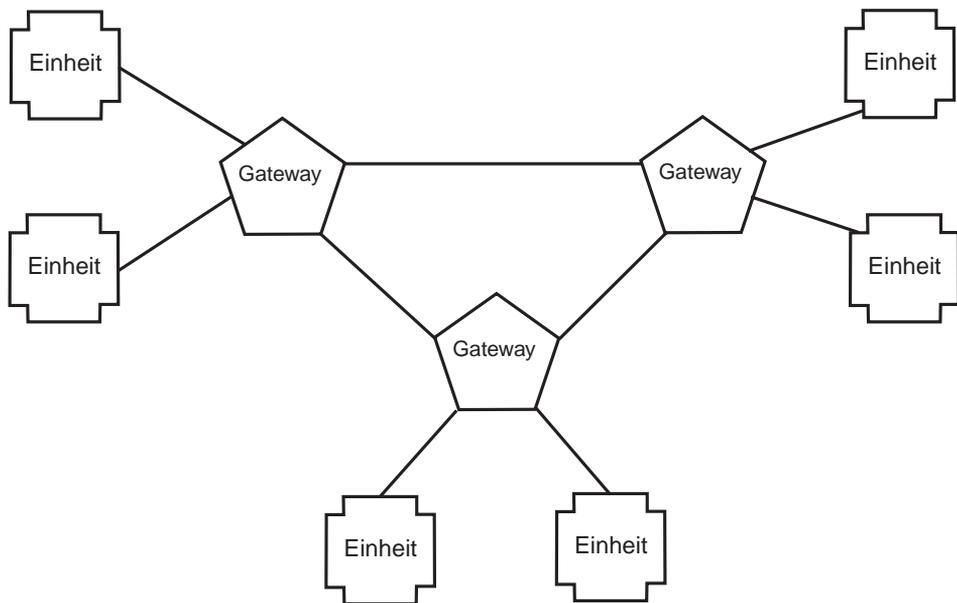


Abbildung 6. MQSeries Everyplace-Sternnetz

Asynchrone Nachrichtenzustellung

Bei der asynchronen Einreihung einer Nachricht in eine ferne Warteschlange wird das Nachrichtenobjekt lokal im Sicherungsspeicher abgelegt, der für die lokale Definition dieser Warteschlange angelegt wurde; hier werden darüber hinaus auch die Namen des Ziel-WS-Managers und der Zielwarteschlange sowie die Merkmale der Komprimierungs-, Authentifizierungs- und Verschlüsselungsfunktion, die mit denen der Zielwarteschlange übereinstimmen, abgelegt. Die Methode **dump** des Objekts wird aufgerufen, wenn das Objekt wie in der Zielwarteschlange definiert in einem sicheren Format im permanenten Speicher abgelegt wird. Die Nachrichtenzustellung wird vom WS-Manager gesteuert. Er ermittelt für den nächsten Hop einen Kanal mit den entsprechenden Merkmalen (bzw. richtet ihn ein) zum WS-Manager und erstellt anschließend ein Übertragungsprogramm (bzw. verwendet es wieder) für die Übertragung zur Zielwarteschlange. Das Übertragungsprogramm gibt das Objekt in eine Bytefolge aus, die es anschließend überträgt. Der Name des Ziel-WS-Managers und der Warteschlange sind nicht Bestandteil des Nachrichtenflusses.

Falls erforderlich, wird die Nachricht verschlüsselt und komprimiert über den Kanal übertragen. Bei der Ankunft im Ziel-WS-Manager wird sie wieder entschlüsselt und dekomprimiert. Anschließend wird über die Wiederherstellungsmethode der Objektklasse ein neues Nachrichtenobjekt erstellt, das in die Zielwarteschlange eingereiht wird. Wurde die Nachricht nicht an den Ziel-WS-Manager übergeben, wird sie entschlüsselt und dekomprimiert und in eine Warteschlange für Speicherung und Weiterleitung mit den entsprechenden Merkmalen für die weitere Übertragung eingereiht. In beiden Fällen wird die Nachricht wie von der Zielwarteschlange vorgegeben in einem sicheren Format in die Warteschlange eingereiht.

Ein Merkmal der asynchronen Nachrichtenzustellung ist die Übergabe der Nachrichten an den WS-Manager auf zwischengeschalteten Hops, wo sie in Warteschlangen für die weitere Übertragung eingereiht werden. Aus diesen zwischengeschalteten Warteschlangen werden Nachrichten erst nach Priorität und an zweiter Stelle erst nach ihrer Zeitmarke weitergeleitet.

Synchrone Nachrichtenzustellung

Die synchrone Nachrichtenzustellung verläuft ähnlich wie die oben beschriebene asynchrone Zustellung, allerdings sind die WS-Manager auf den zwischengeschalteten Hops auf einer wesentlich niedrigeren Ebene (bei der das Übertragungsprogramm und die Kanäle eine Rolle spielen) an dem Vorgang beteiligt. Von Endpunkt zu Endpunkt wird ein Kanal eingerichtet, und zwar unter Verwendung der in der Protokollspezifikation definierten Adapter auf den einzelnen zwischengeschalteten Hops, um den jeweils nächsten Verbindungsabschnitt zu ermitteln. Am Ende des letzten Verbindungsabschnitts, an dem keine weiteren Dateideskriptoren mehr vorhanden sind, wird die Nachricht zur Verarbeitung an die höheren Schichten des WS-Managers übergeben. Der sendende Knoten reiht also die Nachricht nicht selbst in die Warteschlange ein, sondern überträgt sie über einen Kanal über zwischengeschaltete Hops und übergibt sie schließlich dem Ziel-WS-Manager, der die Nachricht in die Zielwarteschlange einreicht.

Die Verbindung zu MQSeries erfolgt über eine Brückenwarteschlange auf dem Gateway, auf dem die Nachricht in ein MQSeries-Format umgesetzt wird. Dieses Verfahren ermöglicht wie in MQSeries Everyplace die synchrone Nachrichtenübertragung von einer Einheit zu MQSeries, wobei der dynamische Kanal am Gateway abgeschlossen wird. Die Nachricht wird in Echtzeit über einen Client-Kanal vom Gateway an einen MQSeries-Server übertragen. Von dort kann die Zielwarteschlange bei Bedarf die asynchrone Weiterleitung über MQSeries-Nachrichtenkanäle anfordern.

Auf ähnliche Weise kann eine Einheit, auf der lediglich die synchrone Nachrichtenübertragung unterstützt wird, Nachrichten an eine asynchrone MQSeries Everyplace-Warteschlange übertragen, vorausgesetzt, es ist eine entsprechende zwischengeschaltete Stelle vorhanden.

Sicherheit

In MQSeries Everyplace stehen integrierte Sicherheitseinrichtungen zur Verfügung, die den Schutz von Nachrichten im gespeicherten Zustand und bei der Übertragung ermöglichen.

Die MQSeries Everyplace-Sicherheitseinrichtungen können in drei Kategorien eingeteilt werden:

- **Lokale Sicherheit:** Lokaler Schutz von Nachrichtendaten und anderen Daten
- **Sicherheit auf Warteschlangenebene:** Schutz von Nachrichten zwischen dem Ursprungs-WS-Manager und der Zielwarteschlange
- **Sicherheit auf Nachrichtenebene:** Schutz auf Nachrichtenebene für Nachrichten zwischen Sender und Empfänger

Die lokale Sicherheit und die Sicherheit auf Nachrichtenebene werden von MQSeries Everyplace intern genutzt, können jedoch auch MQSeries Everyplace-Anwendungen zugänglich gemacht werden. Bei der Sicherheit auf Warteschlangenebene handelt es sich um einen internen Service.

Die MQSeries Everyplace-Sicherheitseinrichtungen aller drei Kategorien schützen die Nachrichtendaten unter Verwendung eines Attributs (**MQAttribute** oder ein Erbe). Je nach Kategorie wird das Attribut explizit oder implizit verwendet.

Jedes Attribut enthält folgende Objekte:

- Authentifizierungsfunktion
- Verschlüsselungsfunktion
- Komprimierungsfunktion
- Schlüssel
- Name der Zieldefinitionseinheit

Diese Objekte werden je nach MQSeries Everyplace-Sicherheitseinrichtung unterschiedlich verwendet; in allen Fällen wird jedoch der Schutzmechanismus der jeweiligen MQSeries Everyplace-Sicherheitseinrichtung aktiviert, wenn das einem Nachrichtenobjekt zugeordnete Attribut aufgerufen wird. Dies ist der Fall, wenn die Methode **dump** einer MQSeries Everyplace-Nachricht aufgerufen wird (wenn die Methode **encodeData** des Attributs verwendet wird, beispielsweise um die Nachrichtendaten zu verschlüsseln und zu komprimieren). Der Schutz der MQSeries Everyplace-Sicherheitseinrichtung wird aufgehoben, wenn die Methode **restore** der MQSeries Everyplace-Nachricht verwendet wird (wenn die Methode **decodeData** des Attributs verwendet wird, beispielsweise um die Nachrichtendaten zu dekomprimieren und zu entschlüsseln).

Die in MQSeries Everyplace Version 1.0 für die Authentifizierungs-, Verschlüsselungs- und Komprimierungsfunktion unterstützten Algorithmen sind in Tabelle 13 aufgeführt.

Tabelle 13. Unterstützung für Authentifizierung, Verschlüsselung und Komprimierung

Funktion	Algorithmus
Authentifizierung	Auf der Basis von Mini Certificates (abgeleitet vom WAP Forum WTLS Mini Certificate)
	Überprüfung der Windows NT/2000-Identität
Komprimierung	LZW
	RLE
Verschlüsselung	Triple DES
	DES
	MARS
	RC4
	RC6
	XOR

Lokale MQSeries Everyplace-Sicherheit

Lokale Sicherheit schützt die MQSeries Everyplace-Nachrichtendaten (oder **MQeFields** bzw. Erben von **MQeFields**) lokal. Hierzu wird ein Attribut mit einer entsprechenden symmetrischen Verschlüsselungs- und Komprimierungsfunktion erstellt sowie ein entsprechender "Schlüssel" (über ein Kennwort bzw. einen Ausdruck) erstellt und konfiguriert. Anschließend wird dieser Schlüssel dem Attribut und dieses wiederum der MQSeries Everyplace-Nachricht zugeordnet. In MQSeries Everyplace steht für die Konfiguration der lokalen Sicherheitseinrichtungen die Klasse **MQeLocalSecure** zur Verfügung; für die Konfiguration eines entsprechenden Attributs sowie die Verwaltung des Kennworts bzw. Ausdrucks ist jedoch der Benutzer der lokalen Sicherheitseinrichtungen (MQSeries Everyplace intern oder eine MQSeries Everyplace-Anwendung) zuständig.

MQSeries Everyplace-Sicherheit auf Warteschlangenebene

Sicherheit auf Warteschlangenebene ist für synchrone und asynchrone Nachrichten möglich.

Sicherheit auf Warteschlangenebene für synchrone Nachrichten

Bei Verwendung der Sicherheit auf Warteschlangenebene für synchrone Nachrichten können Anwendungen Sicherheitsüberlegungen MQSeries Everyplace überlassen. Warteschlangen verfügen über die Merkmale Authentifizierung, Verschlüsselung und Komprimierung, mit denen die für den Schutz des Nachrichtenflusses (und der Nachrichten im permanenten Speicher) erforderliche Sicherheitsstufe ermittelt werden kann.

Beim Senden einer Nachricht werden die Sicherheitsmerkmale der Zielwarteschlange aus der lokalen Registrierungsdatenbank abgerufen. Sind keine Sicherheitsmerkmale vorhanden, versucht der WS-Manager, die Merkmale der Zielwarteschlange über den Ziel-WS-Manager zu ermitteln und für die spätere Verwendung im Cache zu speichern. Besteht ein Kanal zu diesem WS-Manager, wird dieser verwendet; andernfalls wird ein neuer Kanal erstellt. Anschließend werden die Attribute der Zielwarteschlange abgerufen.

Je nach der geforderten Servicequalität können die Attribute des Kanals zum Ziel-WS-Manager dynamisch geändert werden. Die möglichen Änderungen hängen von den definierten Regeln ab. In der Regel lässt eine Regel einen Upgrade der Sicherheitsstufe zu (z. B. von keinem zu einem minimalen Schutz oder von einem minimalen zu einem umfangreichen Schutz). Ist kein Upgrade des Kanals möglich oder wird die Sicherheitsstufe als zu hoch eingeschätzt (wenn z. B. kein Schutz erforderlich ist, der verfügbare Kanal jedoch umfassenden Schutz bietet), wird ein neuer Kanal erstellt. Es steht ein Kanal-Pool (der nach Möglichkeit wiederverwendet wird) mit Merkmalen zur Verfügung, die je nach den Anforderungen des Datenverkehrs dynamisch geändert werden. Kanäle, die nicht mehr erforderlich sind, werden automatisch gelöscht. Nachrichten werden immer in Warteschlangen mit der Sicherheitsstufe eingereiht, die über die Merkmale der Zielwarteschlange definiert ist.

Authentifizierungen werden auf Kanalebene ausgeführt, wobei der Aufwand pro Nachricht so gering wie möglich gehalten wird. Sicherheit auf Warteschlangenebene für die synchrone Nachrichtenübertragung wird in der Regel auch zusammen mit symmetrischen Verschlüsselungsfunktionen eingesetzt, was eine rasche Ver- und Entschlüsselung ermöglicht. In diesen Fällen verwendet MQSeries Everyplace jedoch zu Anfang die asymmetrische RSA-Verschlüsselung, um den Austausch zu schützen, mit dem ein gemeinsam verwendeter Schlüssel auf der Sender- und Empfängerseite eingerichtet wird. Anschließend wird eine symmetrische Verschlüsselung zum Schutz der übertragenen Daten verwendet. Unbefugte Datenzugriffe werden in MQSeries Everyplace außerdem noch durch die dynamische Änderung des Schlüssels für jeden Kanalaustausch erschwert. Darüber hinaus sichert MQSeries Everyplace die Datenintegrität, indem vor dem eigentlichen Übertragungsvorgang ein Auszug der Daten erstellt und angehängt wird, der auf dem Empfänger wiederhergestellt und überprüft wird.

Sicherheit auf Warteschlangenebene für asynchrone Nachrichten

Anders als bei der oben beschriebenen synchronen Nachrichtenübertragung ist bei der asynchronen Nachrichtenübertragung die Zielwarteschlange zum Zeitpunkt des **putMessage** (Einreihung der Nachricht in die Warteschlange) nicht unbedingt verfügbar. In diesem Fall kann der WS-Manager die Nachricht nicht sofort übertragen, sondern stellt sie in die Übertragungswarteschlange, wobei sie anhand der Merkmale der Zielwarteschlangen verschlüsselt wird. Zum Zeitpunkt der Übertragung wird die Nachricht entschlüsselt und mit den entsprechenden Merkmalen über den Kanal übertragen. Dadurch sind Nachrichten auch dann geschützt, wenn sie für die Übertragung anstehen. Für die asynchrone Nachrichtenübertragung ist eine Definition der fernen Warteschlange erforderlich, andernfalls können die Merkmale der Zielwarteschlangen nicht ermittelt werden.

Bei der asynchronen Nachrichtenübertragung ist keine Authentifizierung zwischen Ursprung und Ziel möglich. Ist eine Authentifizierung erforderlich, z. B. damit ein Empfänger den Sender ermitteln kann (um über die Annahme der Nachricht zu entscheiden oder den Urheber eindeutig über die digitale Unterschrift (Non-Repudiation) zu identifizieren) oder ein Sender sicherstellen kann, dass die Nachricht nur vom betreffenden Empfänger verarbeitet werden kann, ist Sicherheit auf Nachrichtenebene erforderlich.

Die Sicherheit auf Warteschlangenebene kann zusammen mit der Sicherheit auf Nachrichtenebene eingesetzt werden, ist jedoch nicht erforderlich, da der Schutz der Nachrichtendaten gewährleistet ist.

Sicherheit auf Nachrichtenebene

Die Sicherheit auf Nachrichtenebene schützt die Nachrichtendaten, die zwischen einer sendenden und einer empfangenden MQSeries Everyplace-Anwendung übertragen werden.

Die Sicherheit auf Nachrichtenebene ist ein Service der Anwendungsschicht; die sendende MQSeries Everyplace-Anwendung muss ein Attribut auf Nachrichtenebene einrichten, das beim Einreihen der Nachricht in eine Zielwarteschlange mit **putMessage** übergeben wird. Die empfangende Anwendung muss ein entsprechendes Attribut auf Nachrichtenebene einrichten und dem empfangenden WS-Manager übergeben; dieses Attribut muss verfügbar sein, wenn die Anwendung die Nachricht unter Aufruf von **getMessage** aus der Zielwarteschlange abrufen.

Wie die lokalen Sicherheitseinrichtungen macht sich auch die Sicherheit auf Nachrichtenebene die Anwendung eines Attributs auf ein Nachrichtenobjekt zunutze. Der WS-Manager der sendenden Anwendung führt beim **putMessage**-Aufruf die Methode **dump** aus, wobei die Nachrichtendaten mit der Methode **encodeData** des Attributs geschützt werden. Der WS-Manager der empfangenden Anwendung führt beim Aufruf **getMessage** der Anwendung die Methode **restore** aus, bei der die ursprünglichen Nachrichtendaten mit der Methode **decodeData** des Attributs wiederhergestellt werden.

MQSeries Everyplace stellt zwei alternative Attribute für die Sicherheit auf Nachrichtenebene zur Verfügung:

MQeMAttribute

Dieses Attribut wird für die Kommunikation zwischen Unternehmen verwendet, bei der das gegenseitige Vertrauen seine Umsetzung in der Anwendungsschicht findet und kein vertrauenswürdiger Dritter erforderlich ist. Alle verfügbaren symmetrischen MQSeries Everyplace-Verschlüsselungs- und Komprimierungsfunktionen können verwendet werden. Wie bei der lokalen Sicherheit muss der Attributsschlüssel zunächst eingerichtet werden, bevor er mit **putMessage** oder **getMessage** zur Verfügung gestellt wird. **MQeAttribute** stellt einen einfachen aber wirkungsvollen Schutzmechanismus für den Schutz auf Nachrichtenebene zur Verfügung, wobei die Nachrichtendaten durch eine effiziente Verschlüsselung geschützt werden, ohne dass hierzu eine Infrastruktur, die mit öffentlichen Schlüsseln arbeitet (PKI = Public Key Infrastructure), erforderlich ist.

MQeMTrustAttribute

Bei diesem Attribut handelt es sich um eine erweiterte Lösung, bei der digitale Unterschriften verwendet werden und die sich die standardmäßige öffentliche Schlüsselinfrastruktur zunutze macht. Es setzt die digitale Unterschrift/Überprüfung des Standards ISO9796 ein und ermöglicht damit der empfangenden Anwendung, die Identität des Nachrichtensenders zu überprüfen. Die von diesem Attribut zur Verfügung gestellte Verschlüsselungsfunktion wird zum Schutz der Nachrichtendaten eingesetzt. Ein SHA1-Auszug stellt die Nachrichtenintegrität sicher und die RSA-Verschlüsselung/Entschlüsselung gewährleistet, dass die Nachricht nur vom betreffenden Empfänger wiederhergestellt werden kann.

Wie bei **MQeAttribute** können alle verfügbaren symmetrischen MQSeries Everyplace-Verschlüsselungs- und Komprimierungsfunktionen verwendet werden. Um den Umfang so gering wie möglich zu halten, werden Mini-Zertifikate (Mini Certificates) verwendet, die auf dem vom WAP Forum WTLS Specification vorgeschlagenen WTLS Certificate basieren. Die Verfügbarkeit der für die Authentifizierung (Unterschriftenüberprüfung) sowie für die Verschlüsselung/Entschlüsselung erforderlichen Informationen an beiden Endpunkten wird über die MQSeries Everyplace-Standardinfrastruktur zur Verfügung gestellt.

Eine über **MQeTrustAttribute** geschützte Nachricht hat in der Regel folgendes Format:
RSA-enc{SymKey}, SymKey-enc {Data, DataDigest, DigSignature}

Dabei gilt:

RSA-enc:	RSA-Verschlüsselung mit dem öffentlichen Schlüssel des betreffenden Empfängers
SymKey:	Der symmetrische quasi-zufällig generierte Schlüssel
SymKey-enc:	Symmetrische Verschlüsselung mit SymKey
Data:	Nachrichtendaten
DataDigest:	Auszug der Nachrichtendaten
DigSignature	Digitale Unterschrift des Senders für die Nachrichtendaten

Die Sicherheit auf Nachrichtenebene und die Sicherheit auf Warteschlangenebene arbeiten unabhängig voneinander.

Registrierungsdatenbank

Bei der Registrierungsdatenbank handelt es sich um den Primärspeicher für WS-Managerspezifische Informationen; für jeden WS-Manager ist jeweils eine Datenbank vorhanden. Jeder WS-Manager speichert folgende Daten in dieser Datenbank:

- Konfigurationsdaten des WS-Managers
- Warteschlangendefinitionen
- Definitionen ferner Warteschlangen
- Definitionen ferner WS-Managers
- Benutzerdaten (einschließlich konfigurationsspezifischer Sicherheitsdaten)

Der Zugriff auf die Registrierungsdatenbank ist in der Regel auf den berechtigten WS-Managerbenutzer beschränkt und durch eine PIN-Nummer geschützt; diese Sicherheitseinschränkungen können jedoch über eine Option für Benutzer aufgehoben werden, für die die Systembelegung einen höheren Stellenwert als die Sicherheit einnimmt.

Authentifizierbare MQSeries Everyplace-Definitionseinheiten

Die Sicherheit auf Warteschlangenebene, bei der eine gegenseitige Authentifizierung über Mini-Zertifikate (Mini Certificates) stattfindet, und die Sicherheit auf Nachrichtenebene, bei der digitale Unterschriften verwendet werden, haben zum Konzept der "authentifizierbaren Definitionseinheit" geführt. Bei der gegenseitigen Authentifizierung denkt man in der Regel an die gegenseitige Überprüfung zweier Benutzer (Menschen), aber im Allgemeinen gibt es dieses Konzept von Benutzern bei der Nachrichtenübertragung nicht. Die Nachrichtenübertragung wird in der Regel in der Anwendungsschicht gesteuert, d. h. von den Benutzern der Nachrichtenübertragungsservices. Daher wurde in MQSeries Everyplace im Hinblick auf das "Ziel einer Authentifizierung" das Konzept vom menschlichen Benutzer durch die abstrakte "authentifizierbare Definitionseinheit" ersetzt. Dies schließt nicht aus, dass es sich bei den authentifizierbaren Definitionseinheiten auch um Personen handeln kann, dies wird jedoch von der Anwendung festgelegt. Intern werden von MQSeries Everyplace alle WS-Manager als authentifizierbare Definitionseinheiten definiert, die Mini Certificate-spezifische Services aufrufen oder für die diese Services aufgerufen werden können. Darüber hinaus werden Warteschlangen, die Mini Certificate-spezifische Authentifizierungsfunktionen verwenden, von MQSeries Everyplace als authentifizierbare Definitionseinheit definiert. So können WS-Manager, die diese Services unterstützen, über eine authentifizierbare Definitionseinheit, nämlich den WS-Manager, oder über mehrere authentifizierbare Definitionseinheiten, nämlich den WS-Manager und alle Warteschlangen, die die Mini Certificate-spezifische Authentifizierungsfunktion verwenden, verfügen.

Private Registrierungsdatenbank und Berechtigungen

Sinnvollerweise muss jede authentifizierbare Definitionseinheit über eigene Berechtigungen verfügen. Daraus ergeben sich zwei Überlegungen, zum einen, wie die Registrierung durchgeführt werden muss, damit die Berechtigungen abgerufen werden können, und zum anderen, wo diese Berechtigungen sicher verwaltet werden können. In der Regel bereitet die Lösung dieser Fragen größere Schwierigkeiten als die Implementierung des Verschlüsselungsverfahrens selbst. MQSeries Everyplace stellt Folgendes zur Verfügung: Standardservices, die authentifizierbaren Definitionseinheiten die Ausführung automatischer Registrierungen ermöglichen, eine private Registrierungsdatenbank (ein Erbe der Basisregistrierungsdatenbank) für die gesicherte Verwaltung der privaten Berechtigungen einer authentifizierbaren Definitionseinheit sowie eine öffentliche Registrierungsdatenbank (ebenfalls ein Erbe der Basisregistrierungsdatenbank) für die Verwaltung öffentlicher Berechtigungen. Die private Registrierungsdatenbank bietet die Möglichkeiten der Basisregistrierungsdatenbank mit vielen Merkmalen eines gesicherten oder verschlüsselten Tokens; so kann sie z. B. als gesichertes Repository für öffentliche Objekte wie Mini Certificates oder für private Objekte wie private Schlüssel verwendet werden. Ein besonderer Mechanismus erlaubt nur dem berechtigten Benutzer den Zugriff auf die privaten Objekte. Die private Registrierungsdatenbank stellt außerdem Unterstützung für Services (z. B. digitale Unterschriften und RSA-Verschlüsselung) zur Verfügung, über die sichergestellt wird, dass die privaten Objekte nie diese private Datenbank verlassen. Über eine einheitliche Schnittstelle wird die zugrundeliegende Einheitenunterstützung transparent; diese Unterstützung ist momentan auf das lokale Dateisystem beschränkt, kann jedoch später unter Umständen auf die Zuordnung zu übertragbaren Token erweitert werden.

Automatische Registrierung

MQSeries Everyplace stellt StandardServices für die Unterstützung der automatischen Registrierung zur Verfügung. Diese Services werden bei der Konfiguration einer authentifizierbaren Definitionseinheit automatisch ausgelöst, beispielsweise beim Start eines WS-Managers oder bei der Definition einer neuen Warteschlange. In beiden Fällen werden die Registrierung gestartet und neue Berechtigungen erstellt und in der privaten Registrierungsdatenbank der authentifizierbaren Definitionseinheit abgelegt. Die automatische Registrierung umfasst die Generierung eines neuen RSA-Schlüsselpaares, den Schutz und das Speichern des privaten Schlüssels in der privaten Registrierungsdatenbank sowie die Packung des öffentlichen Schlüssels in die Anforderung für ein neues Zertifikat an den standardmäßigen Mini Certificate Server. Ist der Mini Certificate Server konfiguriert und verfügbar, gibt er das neue Mini Certificate für die authentifizierbare Definitionseinheit zusammen mit dem eigenen Mini Certificate zurück. Diese werden zusammen mit dem geschützten privaten Schlüssel als neue Berechtigungen der authentifizierbaren Definitionseinheit in deren privaten Registrierungsdatenbank gespeichert. Während mit der automatischen Registrierung auf einfache Weise die Berechtigungen für eine authentifizierbare Definitionseinheit eingerichtet werden können, ist für den Schutz auf Nachrichtenebene (**MQeTrustAttribute**, siehe oben) außerdem noch der Zugriff auf den öffentlichen Schlüssel (Mini Certificate) des betreffenden Empfängers erforderlich.

Öffentliche Registrierungsdatenbank und Zertifikatreplikation

MQSeries Everyplace stellt StandardServices zur Verfügung, die MQSeries Everyplace-Komponenten die gemeinsame Benutzung der öffentlichen Berechtigungen (Mini Certificates) authentifizierbarer Definitionseinheiten ermöglicht. Diese öffentlichen Berechtigungen sind die Voraussetzung für die Sicherheit auf Nachrichtenebene auf Basis von **MQeMTrust**. Die öffentliche MQSeries Everyplace-Registrierungsdatenbank stellt ein Repository mit öffentlichen Zugriffsmöglichkeiten für Mini Certificates dar. Dies entspricht einem persönlichen Telefonverzeichnis auf einem mobilen Telefon, nur dass es sich hier nicht um Telefonnummern, sondern um die Mini Certificates der authentifizierbaren Definitionseinheiten handelt, zu denen am häufigsten eine Verbindung hergestellt wird. Bei den Services der öffentlichen Registrierungsdatenbank handelt es sich nicht nur um passive Services. Wird die Datenbank nach einem nicht vorhandenen Mini Certificate durchsucht und ist sie mit einer gültigen Ausgangs-Server-Komponente konfiguriert, ruft die öffentliche Registrierungsdatenbank das angeforderte Mini Certificate automatisch aus der öffentlichen Registrierungsdatenbank des Ausgangs-Servers ab. Diese Services können als intelligenter Service für die automatische Replikation von Mini Certificates eingesetzt werden, die dafür sorgen, dass die richtigen Mini Certificates immer zur Verfügung stehen.

Verwendung der Datenbankservices durch Anwendungen

Der MQSeries Everyplace-WS-Manager ist für die Nutzung der privaten und öffentlichen Registrierungsdatenbanken konzipiert, allerdings ist der Zugriff auf diese Services nicht eingeschränkt. Es sind MQSeries Everyplace-Lösungen denkbar, in denen eigene authentifizierbare Definitionseinheiten, beispielsweise Benutzer, definiert und verwaltet werden können. Die Services der privaten Registrierungsdatenbank können in diesem Fall für die automatische Registrierung und Verwaltung der Berechtigungen der neuen authentifizierbaren Definitionseinheiten verwendet werden, während die Services der öffentlichen Registrierungsdatenbank den Zugriff auf Berechtigungen ermöglichen, wann immer dies erforderlich ist. Alle registrierten authentifizierbaren Definitionseinheiten können als Aufrufer bzw. Empfänger von Services auf Nachrichtenebene, die mit **MQEmTrustAttribute** geschützt werden, verwendet werden.

Standardservices für die Vergabe von Mini Certificates

MQSeries Everyplace stellt Standardservices für die Vergabe von Mini Certificates zur Verfügung, die für die Verarbeitung von Anforderungen für eine automatische Registrierung in privaten Registrierungsdatenbanken konfiguriert werden. Mit den von MQSeries Everyplace zur Verfügung gestellten Tools kann eine Lösung implementiert werden, mit der ein Service zur Vergabe von Mini Certificates an bestimmte Definitionseinheiten eingerichtet und verwaltet werden kann. Dieser Service verfügt über folgende Merkmale:

- Verwaltung einer Gruppe registrierter authentifizierbarer Definitionseinheiten
- Vergabe von Mini Certificates (basierend auf dem WAP WTLS Mini Certificate)
- Verwaltung des Mini Certificate Repository

Mit den in MQSeries Everyplace verfügbaren Tools kann der Administrator der Services zur Vergabe von Mini Certificates die Vergabe von Mini Certificates an bestimmte Definitionseinheiten steuern, indem der Name der jeweiligen Definitionseinheit und deren registrierte Adresse registriert und eine Anforderungs-PIN für die einmalige Verwendung eines Zertifikats definiert wird. Dies erfolgt in der Regel, nachdem die Identität des Anforderers offline überprüft wurde. Die Anforderungs-PIN für das Zertifikat wird an den entsprechenden Benutzer gesendet (z. B. auf ähnliche Weise, wie Bankkarten-PINs bei Ausgabe einer neuen Bankkarte an den Benutzer geschickt werden). Die Benutzer der privaten Registrierungsdatenbank (z. B. die MQSeries Everyplace-Anwendung oder der MQSeries Everyplace-WS-Manager) können anschließend so konfiguriert werden, dass sie beim Start diese Anforderungs-PIN für das Zertifikat zur Verfügung stellen. Wenn von der privaten Registrierungsdatenbank die automatische Registrierung ausgelöst wird, überprüft der Service zur Ausgabe von Mini Certificates die betreffende neue Zertifikatsanforderung (wobei der Name der angegebenen Definitionseinheit und die Anforderungs-PIN für das Zertifikat mit den registrierten Angaben verglichen werden), gibt das neue Mini Certificate aus und setzt die registrierte Anforderungs-PIN für das Zertifikat zurück, damit diese nicht mehr verwendet werden kann. Alle automatisch registrierten Mini Certificate-Anforderungen werden über einen gesicherten Kanal übertragen. Die von einem entsprechenden Service ausgegebenen Mini Certificates werden in der eigenen Registrierungsdatenbank dieses Services gespeichert. Bei der erneuten Ausgabe eines Mini Certificate (z. B. im Falle einer Zeitlimitüberschreitung) wird das abgelaufene Mini Certificate archiviert.

Sicherheitsschnittstelle

Es steht eine optionale Schnittstelle zur Verfügung, die von einem benutzerdefinierten Sicherheitsmanager implementiert werden kann. Die Methoden ermöglichen dem Sicherheitsmanager, folgende Anforderungen zuzulassen bzw. zurückzuweisen:

- Hinzufügen bzw. Entfernen von Klassenaliasnamen
- Adapterdefinitionen
- Zuordnung von Dateideskriptoren
- Verarbeitung von Kanalbefehlen

Konfiguration und Anpassung

Regeln

Bei Regeln handelt es sich um Java-Klassen, über die das Betriebsverhalten von MQSeries Everyplace bei verschiedenen Statusänderungen gesteuert werden kann. Die notwendigen Standardregeln werden zur Verfügung gestellt, sie können jedoch vom Kunden bei Bedarf durch anwendungs- oder installationsspezifische Regeln ersetzt werden. Die unterstützten Regelarten unterscheiden sich nicht durch ihre Funktion, sondern durch die Auslöseart; Regeln enthalten Logikcode und können daher eine Vielzahl von Funktionen ausführen.

Attributsregeln

An diese Regelklasse wird die Steuerung übergeben, sobald eine Statusänderung auftritt, wie beispielsweise eine Änderung der:

- Authentifizierungsfunktion
- Komprimierungsfunktion
- Verschlüsselungsfunktion

Die Regel kann die Änderung akzeptieren oder zurückweisen.

MQSeries-Brückenregeln

Diesen Regelklassen wird die Steuerung übergeben, wenn im Code der Brücke zwischen MQSeries Everyplace und MQSeries eine Änderung auftritt. Für jede der folgenden Situationen steht eine eigene Brückenregelklasse zur Verfügung:

- Vorgehensweise, wenn eine Nachricht von MQSeries nicht vom Empfangsprogramm an MQSeries Everyplace übergeben werden kann (z. B. weil die Nachricht zu lang oder die Warteschlange nicht vorhanden ist).
- Der Status, in dem von der Brücke verwaltete Objekte unmittelbar nach dem Erstellen eines Server-Exemplars starten sollten.
- Vorgehensweise, wenn die Brücke ein Problem in Zusammenhang mit der Sync-Warteschlange (dem permanenten Speicher, der für die Wiederherstellung nach einem Absturz herangezogen wird) auf MQSeries feststellt. Dieses Problem wird von den Standardregeln lediglich angezeigt.

- Die Vorgehensweise bei der Umsetzung von MQSeries Everyplace-Nachrichten in MQ-Nachrichten und umgekehrt. Die Umsetzungsprogramme für die Nachrichtenkonvertierung zwischen MQSeries Everyplace und MQ werden nicht von **MQRule**-Klassen abgeleitet, sondern müssen die Schnittstelle **MQRTransformerInterface** implementieren. Davon abgesehen arbeiten Umsetzungsprogramme ähnlich wie Regeln und werden aufgerufen, wenn für eine Nachricht eine Formatumsetzung erforderlich ist.

RAS-Wählprogrammregeln

An diese Regelklasse wird die Steuerung übergeben, sobald eine Statusänderung in Zusammenhang mit dem RAS-Wählprogramm auftritt:

- Wenn zu einer angewählten Nummer keine Verbindung hergestellt werden kann.
- Wenn eine Fehlerschwelle überschritten wird.
- Wenn ein Wählvorgang durchgeführt wird und Verbindungen zu bestimmten Zeiten verwendet werden sollen (z. B. nur zu Zeiten mit wenig Datenverkehr).

Warteschlangenregeln

An diese Regelklasse wird die Steuerung übergeben, wenn eine Änderung am Warteschlangenstatus auftritt, wie beispielsweise in folgenden Fällen:

- Beim Einreihen einer Nachricht in die Warteschlange. In diesem Fall kann beispielsweise festgestellt werden, ob ein Schwellenwert (Anzahl der Nachrichten, Nachrichtenlänge, ungültiges Merkmal) überschritten wurde.
- Es wurden Warteschlangenkenndaten zugeordnet oder geändert.
- Die Warteschlange wird geöffnet bzw. geschlossen.
- Die Warteschlange soll gelöscht werden.

WS-Managerregeln

An diese Regelklasse wird die Steuerung übergeben, wenn eine Änderung am WS-Managerstatus auftritt, wie beispielsweise in folgenden Fällen:

- Der WS-Manager wird geöffnet. Es kann beispielsweise ein Zeitgeber-Thread, der im Hintergrund abläuft, gestartet werden, damit zeitlich angesetzte Vorgänge aktiviert werden.
- Der WS-Manager wird geöffnet. Beispielsweise kann der Zeitgeber-Thread, der im Hintergrund abläuft, beendet werden.
- Es wird eine neue Warteschlange hinzugefügt.

Verbindungsarten

MQSeries Everyplace unterstützt Client/Server¹⁰ und/oder Peer-zu-Peer-Betrieb. Ein *Client* kann die Kommunikation mit einem Server initialisieren, ein *Server* hingegen reagiert lediglich auf die vom Client abgesetzten Anforderungen. In einer *Peer-zu-Peer*-Umgebung ist der Kommunikationsfluss in beide Richtungen möglich. Für diese Verbindungsarten sind jeweils unterschiedliche MQSeries Everyplace-Komponenten erforderlich. Hierbei handelt es sich um folgende Komponenten:

- **Kanalempfangsprogramm:** Ist für eingehende Verbindungsanforderungen empfängsbereit.
- **Kanalmanager:** Unterstützt mehrere gleichzeitige logische Übertragungsleitungen zwischen Endpunkten.
- **WS-Manager:** Unterstützt Anwendungen über die Steuerung der Nachrichtenübertragung und die Steuerung von Warteschlangen.

In Tabelle 14 ist aufgeführt, welche Komponenten für welche Verbindungsart erforderlich sind. Bei der Client/Server-Verbindungsart kann MQSeries Everyplace entweder im Client- oder im Server-Modus betrieben werden. Bei der Servlet-Option wird MQSeries Everyplace als HTTP-Servlet konfiguriert, wobei der HTTP-Server selbst für eingehende Verbindungsanforderungen empfängsbereit sein muss.

Tabelle 14. Verbindungsarten

	WS-Manager	Kanalmanager	Kanalempfangsprogramm
Client	Ja		
Client/Server	Ja	Ja	Ja
Peer	Ja		
Server	Ja	Ja	Ja
Servlet	Ja	Ja	

Aus Lizenzgründen wird eine MQSeries Everyplace-Instanz bei Verwendung eines MQSeries Everyplace-Kanalmanagers oder -Kanalempfangsprogramms als Gateway angesehen.

Welche Verbindungsart von den WS-Managern verwendet wird, hat keine direkten Auswirkungen auf die MQSeries Everyplace-Anwendungen. Die Verbindungsart legt jedoch fest, welche Ressourcen für die Partner zur Verfügung stehen, zwischen welchen WS-Managern eine Verbindung hergestellt werden kann, wie groß der Speicherbedarf für MQSeries Everyplace auf einer Einheit oder einem Gateway ist und welche Verbindungen gleichzeitig bestehen können.

10. In diesem Abschnitt werden die Begriffe "Client" und "Server" in ihrer gängigen Bedeutung verwendet, nicht in ihrer Bedeutung im MQSeries-Kontext.

Peer-zu-Peer-Verbindung

Peer-zu-Peer-Kanäle weisen die Möglichkeiten auf, die einem einzelnen Kanal mit einem Kanalmanager und einem Kanalempfangsprogramm zur Verfügung stehen. Bei der Erstellung eines Peer-zu-Peer-Kanals zwischen zwei WS-Managern übernimmt einer von diesen die Aufgaben des Empfangsprogramms, während der andere WS-Manager Verbindungsanforderungen absetzt. Ein WS-Manager, der für diese Verbindungsart konfiguriert ist, kann mehrere Peer-zu-Peer-Verbindungen mit anderen WS-Managern einleiten, jedoch nur jeweils auf eine eingehende Verbindungsanforderung reagieren; weitere Verbindungsanforderungen können immer erst nach Beendigung des vorherigen Peer-zu-Peer-Kanals bearbeitet werden. Die beiden beteiligten WS-Manager können über jeden beliebigen Peer-zu-Peer-Kanal Vorgänge einleiten; so können beispielsweise Anwendungen auf den WS-Managern jeweils auf die Warteschlangen des anderen WS-Managers zugreifen.

Peer-zu-Peer-Kanäle können unter Umständen nicht bei Vorhandensein einer Firewall verwendet werden, da das Ziel der eingehenden Verbindungsanforderung eventuell von der Firewall zurückgewiesen wird.

Client/Server-Verbindung

Standardkanäle für Client/Server-Verbindungen verfügen nicht über die Möglichkeiten, Verbindungsanforderungen zu erkennen, daher ist ein unabhängiges Empfangsprogramm auf dem Server erforderlich; darüber hinaus muss auf dem Server ein Kanalmanager für die Steuerung mehrerer gleichzeitiger Kanäle vorhanden sein. Der Client setzt eine Verbindungsanforderung ab, auf die der Server reagiert. Ein Server kann in der Regel gleichzeitige Anforderungen mehrerer Clients verarbeiten. Über einen Standardkanal erhält der Client Zugriff auf die Server-Ressourcen. Ist für eine Anwendung auf dem Server der synchrone Zugriff auf die Client-Ressourcen erforderlich, muss ein zweiter Kanal erstellt werden, bei dem die Rollen vertauscht sind. Da Standardkanäle jedoch bidirektional sind, werden die an einen Client gerichteten Nachrichten aus der Übertragungswarteschlange des Servers über den Standardkanal (Client/Server-Kanal) zugestellt, der vom Client eingeleitet wurde.

Ein Client kann mehreren Servern zugeordnet sein. Für diese Konfiguration ist kein Kanalmanager erforderlich, da Kanalmanager mehrere eingehende Kanäle handhaben können.

Client/Server-Verbindungen können im Allgemeinen bei Vorhandensein von Firewalls eingesetzt werden, da das Ziel der eingehenden Verbindung in der Regel von der Firewall zugelassen wird.

Mehrere Verbindungsarten

Ein einzelner WS-Manager kann Peer-zu-Peer- oder Client/Server-Verbindungen einleiten und entweder als Server oder als Peer reagieren. In diesem Fall müssen dem Peer-Kanalempfangsprogramm und dem Standardkanalempfangsprogramm verschiedene Anschlussnummern zugeordnet sein.

Klassen

MQSeries Everyplace stellt für verschiedene Funktionen eine Auswahl von Klassen zur Verfügung, mit denen der MQSeries Everyplace-Betrieb an die spezifischen Anwendungsanforderungen angepasst werden kann. In einigen Fällen werden die Schnittstellen zu Klassen dokumentiert, so dass die Entwicklung zusätzlicher Lösungen ermöglicht wird. Tabelle 15 enthält eine Übersicht über diese Möglichkeiten. Klassen können entweder direkt oder über Aliasnamen identifiziert werden.

Tabelle 15. Klassenoptionen

Klasse	Verfügbare Alternative	Dokumentierte Schnittstellen
Verwaltung	Nein	Ja
Authentifizierungsfunktionen	Ja	Nein
Kommunikationsadapter	Ja	Ja
Übertragungsarten	Ja	Nein
Komprimierungsfunktionen	Ja	Nein
Verschlüsselungsfunktionen	Ja	Nein
Ereignisprotokoll	Beispiel verfügbar	Ja
Nachrichten	Nein	Ja
Warteschlangenspeicher	Ja	Nein
Regeln	Standardklassen verfügbar	Ja
Trace	Beispiele verfügbar	Ja

Anwendungen laden

Bei der Konfiguration von MQSeries Everyplace als Client (oder Peer) ist die Anwendung, von der die Anforderung abgesetzt wird, für das Laden anderer Anwendungen in die JVM verantwortlich. Hierzu können entweder standardmäßige Java-Funktionen verwendet werden, oder das Klassenladeprogramm, das Bestandteil von MQSeries Everyplace ist, muss verfügbar sein. Dies ermöglicht die gleichzeitige Ausführung mehrerer Anwendungen für einen einzelnen WS-Manager auf derselben JVM. Ebenso können mehrere JVMs verwendet werden, allerdings ist für jede ein eigener WS-Manager und ein eindeutiger Name erforderlich.

Bei der Konfiguration von MQSeries Everyplace als Server ist MQSeries Everyplace selbst die Anwendung, von der Verbindungsanforderungen abgesetzt werden. MQSeries Everyplace unterstützt das vorherige Laden einer Klassenliste; die in dieser Liste aufgeführten Klassen werden zuerst geladen, dann erst der WS-Manager selbst.

Kapitel 6. MQSeries Everyplace- und MQSeries-Netze

Ein MQSeries Everyplace-Netz kann zwar als Standalone-Installation, d. h. ohne MQSeries-Server oder -Netz betrieben werden, wird aber häufig als Ergänzung zu einer vorhandenen MQSeries-Installation eingesetzt. Diese kann dadurch um neue Plattformen und Einheiten oder erweiterte Leistungsmerkmale wie beispielsweise warteschlangen- oder nachrichtenbasierte Sicherheit und synchrone Nachrichtenübertragung erweitert werden. Für eine MQSeries Everyplace-Anwendung stellen die MQSeries-Warteschlangen und -WS-Manager lediglich zusätzliche ferne Warteschlangen bzw. WS-Manager dar. Allerdings bestehen hier einige Einschränkungen, da der Zugriff auf diese Warteschlangen nicht direkt über dynamische MQSeries Everyplace-Kanäle und einen MQSeries Everyplace-WS-Manager erfolgen kann, sondern ein MQSeries Everyplace-Gateway erforderlich ist. Der Gateway kann über MQSeries-Client-Kanäle direkt oder indirekt Nachrichten an mehrere MQSeries-WS-Manager senden. Bei einer indirekten Verbindung werden die Nachrichten über MQSeries-Client-Kanäle an zwischengeschaltete MQSeries-WS-Manager gesendet und von dort über MQSeries-Nachrichtenkanäle an den Ziel-WS-Manager übertragen.

Nachrichten von einer MQSeries-Anwendung an MQSeries Everyplace werden wie üblich an den MQSeries Everyplace-WS-Manager und die MQSeries Everyplace-Warteschlange gerichtet. Hierzu wird das MQSeries-Routing (die Definition des fernen WS-Managers) so definiert, dass die MQSeries Everyplace-Nachrichten in besonderen MQSeries-Übertragungswarteschlangen eingereiht werden. MQSeries-Kanäle sind nicht wie sonst für die Übertragungswarteschlangen definiert. Statt dessen ruft der MQSeries Everyplace-Gateway die Nachrichten aus den Warteschlangen ab und sorgt dafür, dass sie dem MQSeries Everyplace-Ziel zugestellt werden. Die Anzahl der Übertragungswarteschlangen (d. h. der Routes) ist konfigurierbar und sollte dem Nachrichtenvolumen entsprechen.

Schnittstellen zu MQSeries

Die MQSeries Everyplace-Architektur unterstützt den Einsatz einer oder mehrerer optionaler Brücken zwischen MQSeries Everyplace und anderen Nachrichtenübertragungssystemen.

In MQSeries Everyplace Version 1.0 wird nur eine Brücke, die *MQSeries-Brücke*, unterstützt, die als Schnittstelle zwischen MQSeries Everyplace- und MQSeries-Netzen fungiert. Diese Brücke stellt über den MQSeries-Java-Client eine Verbindung zu einem oder mehreren MQSeries-WS-Managern her und ermöglicht damit den Austausch von Nachrichten zwischen MQSeries Everyplace und MQSeries. In MQSeries Everyplace Version 1.0 wird pro Gateway nur der Einsatz einer Brücke empfohlen; jede dieser Brücken ist mit mehreren *MQSeries-Proxy-WS-Managern* (den Definitionen von MQSeries-WS-Managern) verbunden. Für jeden MQSeries-WS-Manager, der mit MQSeries Everyplace kommuniziert, ist die Definition eines Proxy-WS-Managers erforderlich. Jeder dieser Definitionen können ein oder mehrere *Client-Verbindungsservices* zugeordnet werden, wobei jeder die Verbindung zu einem einzelnen MQSeries-WS-Manager darstellt.

Diese Client-Verbindungsservices können jeweils unterschiedliche MQSeries-Server-Verbindungen zum WS-Manager sowie unterschiedliche Merkmale wie beispielsweise Benutzer-Exits oder Anschlüsse verwenden.

Ein Gateway-Client-Verbindungsservice kann über verschiedene *Empfangsprogramme* verfügen, die über den Gateway-Service eine Verbindung zum MQSeries-WS-Manager herstellen und Nachrichten von MQSeries an MQSeries Everyplace abrufen. Ein Empfangsprogramm verwendet nur jeweils einen Service, um diese Verbindung herzustellen, und jedes Empfangsprogramm stellt nur eine Verbindung zu jeweils einer Übertragungswarteschlange auf dem MQSeries-WS-Manager her. Jedes Empfangsprogramm überträgt die Nachrichten aus einer MQSeries-Übertragungswarteschlange über den übergeordneten Gateway-WS-Manager an ein beliebiges Ziel im MQSeries Everyplace-Netz. Damit kann ein einzelner Gateway-WS-Manager die Nachrichten aus mehreren MQSeries-Nachrichtenquellen in das MQSeries Everyplace-Netz leiten.

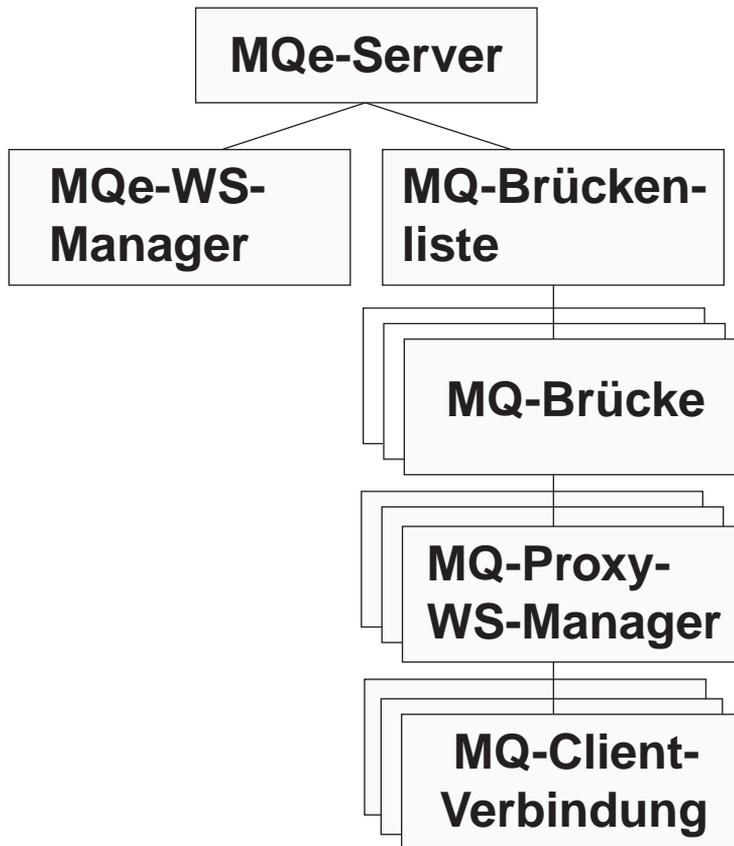


Abbildung 7. MQSeries-Brückenobjekthierarchie

Bei der Nachrichtenübertragung von MQSeries Everyplace an MQSeries konfiguriert der Gateway-WS-Manager ein oder mehrere *Brückenwarteschlangen*objekte. Jedes dieser Objekte kann direkt eine Verbindung zu einem beliebigen WS-Manager herstellen und die Nachrichten an die Zielwarteschlangen senden. Auf diese Weise kann ein Gateway die über einen einzelnen MQSeries Everyplace-WS-Manager weitergeleiteten MQSeries Everyplace-Nachrichten direkt oder indirekt einem beliebigen MQSeries-WS-Manager zustellen. In Tabelle 16 sind die Merkmale des Brückenobjekts aufgeführt.

Tabelle 16. Merkmale des Brückenobjekts

Merkmal	Beschreibung
Brückenname	Eine Liste mit Brücken
Betriebsstatus	Der Status ist aktiv oder gestoppt

Das Brückenobjekt und die anderen Gateway-Objekte können unabhängig vom MQSeries Everyplace-WS-Manager gestartet und gestoppt werden. Wird ein Gateway-Objekt gestartet bzw. gestoppt, werden alle untergeordneten Objekte (d. h. alle Brücken, Proxy-WS-Manager, Client-Verbindungen und Empfangsprogramme der Übertragungswarteschlangen) ebenfalls gestartet bzw. gestoppt. In Tabelle 17 auf Seite 58 sind die Brückenmerkmale aufgeführt.

Tabelle 17. Brückenmerkmale

Merkmal	Beschreibung
Klasse	Brückenklasse
Standardumsetzungsprogramm	Die Standardklasse (Regelklasse) für die Umsetzung von MQSeries Everyplace- in MQSeries-Nachrichten oder umgekehrt; diese Klasse kommt zur Anwendung, wenn der Zielwarteschlange keine andere Umsetzungsprogrammklasse zugeordnet ist.
Intervall der Überwachungssignale	Die Basiszeitgebereinheit für die Ausführung von Aktionen für Brückenobjekte
Name	Der Name des Brückenobjekts
Betriebsstatus	Der Status ist aktiv oder gestoppt
Startregelklasse	Die Regelklasse, die beim Start des Brückenobjekts verwendet wird
Untergeordnete MQSeries-Proxy-WS-Manager	Eine Liste mit allen Proxy-WS-Managern, die der Brücke zugeordnet sind

Für einfache Fälle kann die gesamte Nachrichtenumsetzung von einem Standardumsetzungsprogramm (Regel) ausgeführt werden. Diese Einstellung kann geändert werden, indem zusätzlich ein Umsetzungsprogramm pro Empfangsprogramm (für Nachrichten von MQSeries an MQSeries Everyplace) konfiguriert wird. Für eine erweiterte Steuerung können die Umsetzungsregeln über die Definitionen von Brückenwarteschlangen auf dem Gateway für jede Zielwarteschlange konfiguriert werden; dies gilt für MQSeries Everyplace- und MQSeries-Zielwarteschlangen.

Der MQSeries-Proxy-WS-Manager beschreibt die Merkmale eines bestimmten MQSeries-WS-Managers. In Tabelle 18 sind die Proxy-Merkmale aufgeführt.

Tabelle 18. MQSeries-Proxy-WS-Manager - Merkmale

Merkmal	Beschreibung
Klasse	Die Klasse "MQSeries-Proxy-WS-Manager"
Name des MQSeries-Host	IP-Host-Name, mit dem über die Java-Client-Klassen die Verbindungen zum MQSeries-WS-Manager hergestellt werden. Falls keine Angabe erfolgt, wird davon ausgegangen, dass sich der MQSeries-WS-Manager auf derselben Maschine wie die Brücke und die verwendeten Java-Bindungen befindet.
Name des MQSeries-Proxy-WS-Managers	Der Name des MQSeries-WS-Managers
Name der zugeordneten Brücke	Der Name des Brückenobjekts, dem dieser MQSeries-Proxy-WS-Manager zugeordnet ist
Betriebsstatus	Der Status ist aktiv oder gestoppt
Startregelklasse	Die Regelklasse, die beim Start des MQSeries-WS-Managerobjekts verwendet wird
Untergeordnete Client-Verbindungen	Eine Liste mit allen Client-Verbindungsobjekten, die diesem Proxy zugeordnet sind

Die Definition des Brücken-Client-Verbindungsservices enthält alle Angaben, die für die Herstellung einer Verbindung zu einem MQSeries-WS-Manager erforderlich sind. In Tabelle 19 sind die Verbindungsmerkmale aufgeführt.

Tabelle 19. Client-Verbindungsservice - Merkmale

Merkmal	Beschreibung
Adapterklasse	Die Klasse, die als Gateway-Adapter verwendet wird
CCSID*	Die ganze Zahl, die als MQSeries-CCSID verwendet wird
Klasse	Die Klasse "Brücken-Client-Verbindungsservice"
Maximale Verbindungs-leerlaufzeit	Gibt die maximal zulässige Leerlaufzeit für eine Verbindung an (d. h. die Zeit, in der die Verbindung nicht genutzt wird), bevor sie beendet wird.
MQSeries-Kennwort*	Das vom Java-Client verwendete Kennwort
MQSeries-Anschluss*	IP-Anschlussnummer, die für die Herstellung von Verbindungen zum MQSeries-WS-Manager über die Java-Client-Klassen verwendet wird. Falls keine Angabe erfolgt, wird davon ausgegangen, dass sich der MQSeries-WS-Manager auf derselben Maschine wie die Brücke und die verwendeten Java-Bindungen befindet.
MQSeries-Empfangs-Exit-Klasse*	Wird zum Abgleich mit dem Empfangs-Exit am anderen Ende des Client-Kanals verwendet; dem Exit ist eine Zeichenfolge zugeordnet, die die Übergabe von Daten an den Exit-Code ermöglicht.
MQSeries-Sicherheits-Exit-Klasse*	Wird zum Abgleich mit dem Sicherheits-Exit am anderen Ende des Client-Kanals verwendet; dem Exit ist eine Zeichenfolge zugeordnet, die die Übergabe von Daten an den Exit-Code ermöglicht.
MQSeries-Sende-Exit-Klasse*	Wird zum Abgleich mit dem Send-Exit am anderen Ende des Client-Kanals verwendet; dem Exit ist eine Zeichenfolge zugeordnet, die die Übergabe von Daten an den Exit-Code ermöglicht.
MQSeries-Benutzer-ID*	Die vom Java-Client verwendete Benutzer-ID
Name des Client-Verbindungsservice	Der Name des Server-Verbindungskanals auf der MQSeries-Maschine
Name des zugehörigen Proxy-WS-Managers	Der Name des zugehörigen Proxy-WS-Managers
Startregelklasse	Die Regelklasse, die beim Start des Objekts "Brücken-Client-Verbindungsservice" verwendet wird
Name der Sync-Warteschlange	Der Name der MQSeries-Warteschlange, die von der Brücke für Synchronisationszwecke verwendet wird
Löschregelklasse für Sync-Warteschlange	Die zu verwendende Regelklasse, wenn in der Sync-Warteschlange eine Nachricht vorgefunden wird
Betriebsstatus	Der Status ist aktiv oder gestoppt
Name der Brücke	Der Name des Brückenobjekts, dem diese Client-Verbindung zugeordnet ist

Tabelle 19. Client-Verbindungsservice - Merkmale (Forts.)

Merkm al	Beschreibung
Untergeordnete Empfangsprogramme der MQ-XMIT-Warteschlange	Eine Liste mit allen Empfangsprogrammbjekten, die diese Client-Verbindung verwenden
*Ausführliche Informationen zu diesen Parametern finden Sie im Handbuch <i>MQSeries Using Java</i>	

Die *Adapterklasse* wird zur Nachrichtenübertragung von MQSeries Everyplace an MQSeries, die *Sync-Warteschlange* zur Statusüberwachung des Übertragungsvorgangs verwendet. Der Inhalt dieser Warteschlange wird bei Wiederherstellungen verwendet, um so eine gesicherte Nachrichtenübertragung zu gewährleisten; nach einem normalen Systemabschluss ist diese Warteschlange leer. Diese Klasse kann über mehrere Client-Verbindungen und für mehrere Brückendefinitionen verwendet werden, vorausgesetzt, es handelt sich jeweils um dieselben Empfangs-, Sende und Sicherheits-Exits. Darüber hinaus können in dieser Warteschlange abhängig von den bestehenden Empfangsprogrammmerkmalen Statusinformationen zu Nachrichtenübertragungen von MQSeries an MQSeries Everyplace gespeichert werden. Die *Löschregelklasse für die Sync-Warteschlange* kommt zur Anwendung, wenn in der Sync-Warteschlange eine Nachricht vorgefunden wird. Dies deutet darauf hin, dass eine Nachricht von MQSeries Everyplace nicht bestätigt wurde.

Die maximale Verbindungsleerlaufzeit wird zur Steuerung des Pools mit Java-Client-Verbindungen verwendet, der vom Brücken-Client-Verbindungsservice zum MQSeries-System verwaltet wird. Wird eine MQSeries-Verbindung nicht genutzt, wird ein Zeitgeber gestartet, bei dessen Ablauf die Verbindung beendet wird, falls sie in der Zwischenzeit nicht genutzt wird. Da die Herstellung von MQSeries-Verbindungen äußerst aufwändig ist, stellt dieser Vorgang eine effiziente Wiederverwendung der Verbindungen ohne eine übermäßige Ressourcenbelegung sicher. Der Wert "Null" gibt an, dass kein Verbindungs-Pool verwendet werden soll.

Die Merkmale des Empfangsprogrammbjekts, das Nachrichten von MQSeries an MQSeries Everyplace überträgt, sind in Tabelle 20 aufgeführt.

Tabelle 20. Empfangsprogramm - Merkmale

Merkm al	Beschreibung
Klasse	Die Empfangsprogrammkasse
Name der Warteschlange für nicht zustellbare Nachrichten	Die Warteschlange, in die Nachrichten von MQSeries an MQSeries Everyplace eingereicht werden, die nicht zugestellt werden können
Adapter des Empfangsprogramms für Statusinformationen	Der Klassenname des Adapters, in dem Statusinformationen gespeichert werden
Empfangsprogrammname	Der Name der MQSeries-XMIT-Warteschlange, die Nachrichten zur Verfügung stellt
Name des zugeordneten Client-Verbindungsservices	Der Name des Client-Verbindungsservices

Table 20. Empfangsprogramm - Merkmale (Forts.)

Merkmals	Beschreibung
Betriebsstatus	Der Status ist aktiv oder gestoppt
Startregelklasse	Die Regelklasse, die beim Start des Empfangsprogrammsobjekts verwendet wird
Umsetzungsprogramm-klasse	Die Regelklasse, die bei der Umsetzung von MQSeries- in MQSeries Everyplace-Nachrichten verwendet wird
Regelklasse für nicht zugestellte Nachrichten	Die Regelklasse, die die entsprechende Maßnahme für den Fall, dass Nachrichten von MQSeries an MQSeries Everyplace nicht zugestellt werden können, festlegt
Wartezeit auf Nachrichten in Sekunden	Eine erweiterte Option, mit der die Leistung des Empfangsprogramms in Ausnahmesituationen gesteuert werden kann

Die *Regelklasse für nicht zugestellte Nachrichten* legt die entsprechende Maßnahme für den Fall, dass eine Nachricht von MQSeries an MQSeries Everyplace nicht zugestellt werden kann, fest. In der Regel werden solche Nachrichten in die *Warteschlange für nicht zustellbare Nachrichten* des MQSeries-Systems eingereiht.

Um eine sichere Nachrichtenzustellung zu ermöglichen, verwendet die Empfangsprogrammklasse den *Adapter des Empfangsprogramms für Statusinformationen* zum Speichern von Statusinformationen. Diese werden entweder im MQSeries Everyplace-System oder in der Sync-Warteschlange des MQSeries-Systems gespeichert.

Für eine vollständige Konfiguration der Brücke sind Definitionen ferner WS-Manager und ferner Warteschlangen erforderlich. Definitionen ferner WS-Manager für ferne MQSeries Everyplace-WS-Manager werden wie in MQSeries Everyplace üblich gesetzt; in den Definitionen ferner MQSeries-WS-Manager wird die Kanaldefinition auf Null gesetzt, um anzuzeigen, dass kein normaler dynamischer MQSeries Everyplace-Kanal verwendet wird; statt dessen wird wie oben beschrieben eine Verbindung zum MQSeries-WS-Manager definiert.

Die Definition einer fernen Warteschlange für eine MQSeries Everyplace-Warteschlange wird ebenfalls wie üblich gesetzt; sie unterscheidet sich erheblich von der Definition für eine MQSeries-Warteschlange. In Tabelle 21 auf Seite 62 sind die Merkmale ferner MQSeries-Warteschlangen aufgeführt.

Tabelle 21. Ferne MQSeries-Warteschlangen - Merkmale

Merkmal	Beschreibung
Aliasname	Alternative Namen für die Warteschlange
Authentifizierungsfunktion	Muss auf Null gesetzt werden
Klasse	Die Objektklasse
Client-Verbindung	Der Name des Client-Verbindungsservices, der verwendet werden soll
Komprimierungsfunktion	Muss auf Null gesetzt werden
Verschlüsselungsfunktion	Muss auf Null gesetzt werden
Zeitlimitüberschreitung	Wird an das Umsetzungsprogramm übergeben
Maximal zulässige Nachrichtenlänge	Wird an die Regelklassen übergeben
Modus	Muss auf "synchron" gesetzt werden
MQ-Proxy-WS-Manager	Name des MQSeries-WS-Managers, an den die Nachricht zuerst gesendet werden soll
MQSeries-Brücke	Der Name der Brücke, über die die Nachricht an MQSeries übertragen werden soll
Name	Der Name, unter dem die ferne MQSeries-Warteschlange in MQSeries Everyplace bekannt ist
Zugeordneter WS-Manager	Name des WS-Managers, dem die Definition zugeordnet ist
Priorität	Die Priorität der Nachrichten (wird durch Angabe eines Nachrichtenwertes außer Kraft gesetzt)
Name der fernen MQSeries-Warteschlange	Der Name der fernen MQSeries-Warteschlange
Regel	Die Regelklasse, die für Warteschlangenvorgänge verwendet werden soll
Ziel-WS-Manager	Der MQSeries-WS-Manager, dem die Warteschlange zugeordnet ist
Umsetzungsprogramm	Name der Umsetzungsprogrammklasse, die Nachrichten vom MQSeries Everyplace- in das MQSeries-Format umsetzt
Art	Die MQSeries-Brücken-Warteschlange

Die *Verschlüsselungsfunktions*-, *Authentifizierungsfunktions*- und *Komprimierungsfunktions*klassen definieren eine Reihe von Warteschlangenattributen, die die Sicherheitsstufe für die Nachrichten festlegen, die an diese Warteschlange übertragen werden. Zwischen dem Zeitpunkt, zu dem eine Nachricht in MQSeries Everyplace abgeschickt wird, bis zu ihrer Übergabe an die MQSeries-Brückenwarteschlange gilt für diese Nachricht zumindest die für die Warteschlange definierte Sicherheitsstufe. Diese Sicherheitsstufen gelten *nicht*, wenn die Nachricht von der MQSeries-Brückenwarteschlange an das MQSeries-System gesendet wird; für diese Übertragung werden die Sicherheits-, Sende- und Empfangs-Exits der Client-Verbindung verwendet. Es erfolgt keine Überprüfung, ob die Sicherheitsstufe der Warteschlange aufrecht erhalten bleibt.

MQSeries-Brückenwarteschlangen sind immer synchron; asynchrone Anwendungen müssen daher ihre Nachrichten an diese Warteschlangen über die MQSeries Everyplace-spezifischen Warteschlangen zum Speichern und Weiterleiten (Store and Forward), Ausgangs-Server-Warteschlangen oder über asynchrone Definitionen ferner Warteschlangen senden.

Die Verwaltung des Gateways erfolgt auf dieselbe Weise wie die Verwaltung eines normalen MQSeries Everyplace-WS-Managers, d. h. über Verwaltungsnachrichten. Bei Bedarf werden für das verwaltete Objekt neue Nachrichtenklassen definiert. In Tabelle 11 auf Seite 30 sind die Verwaltungsnachrichtenklassen für den Gateway aufgeführt.

Nachrichtenumsetzung

Die an MQSeries gerichteten MQSeries Everyplace-Nachrichten werden über die Brücke gesendet und mit Hilfe eines Standardumsetzungsprogramms bzw. des Umsetzungsprogramms der Zielwarteschlangen in ein MQSeries-Format umgewandelt. Benutzerdefinierte Umsetzungsprogramme sind sehr flexibel; beispielsweise ist es sinnvoll, eine Unterklasse der MQSeries Everyplace-Nachrichtenobjektklasse für die Darstellung einer bestimmten Nachrichtenart im gesamten MQSeries Everyplace-Netz zu verwenden. Auf dem Gateway kann die Nachricht von einem Umsetzungsprogramm unter Zuordnung der entsprechenden Felder und MQSeries-Werte in ein MQSeries-Format umgesetzt werden, wobei die entsprechenden Daten der Unterklasse hinzugefügt werden.

Das Standardprogramm für die Umsetzung vom MQSeries Everyplace- in das MQSeries-Format kann diese Unterklasseninformationen zwar nicht verwenden, ist jedoch trotzdem vielseitig einsetzbar. Es weist folgende Merkmale auf:

- **Nachrichtenfluss von MQSeries Everyplace an MQS:**

Das Standardumsetzungsprogramm für die Umsetzung vom MQSeries Everyplace- in ein MQSeries-Format arbeitet mit der Klasse **MQeMQMMsgObject** zusammen. Diese Klasse stellt alle Felder dar, die möglicherweise im MQ-Nachrichten-Header vorhanden sind. Mit Hilfe dieser Klasse kann Ihre Anwendung unter Verwendung der Methode **set()** die erforderlichen Werte, wie beispielsweise die Priorität, setzen. Wird ein **MQeMQMMsgObject** (bzw. ein von dieser Klasse abgeleitetes Objekt) an das MQSeries Everyplace-Standardumsetzungsprogramm übergeben, ruft dieses (**MQe-BaseTransformer**) die Werte aus **MQeMSMMsgObject** ab und setzt die entsprechenden Werte in der MQSeries-Nachricht (beispielsweise wird der Wert für die Priorität in die MQSeries-Nachricht kopiert).

Handelt es sich bei der Nachricht weder um ein **MQeMQMMsgObject** noch um ein von **MQeMSMMsgObject** abgeleitetes Objekt, wird die gesamte MQSeries Everyplace-Nachricht in den Hauptteil der MQSeries-Nachricht kopiert (*funneled*). Der im Nachrichtenformatfeld des MQSeries-Nachrichten-Headers gesetzte Wert gibt an, dass die MQSeries-Nachricht eine Nachricht mit diesem speziellen MQSeries Everyplace-Format enthält.

- **Nachrichtenfluss von MQSeries an MQSeries Everyplace:**

Die Umsetzung der an MQSeries Everyplace gerichteten MQSeries-Nachrichten erfolgt in ähnlicher Weise wie die Umsetzung in umgekehrter Richtung. Das Standardumsetzungsprogramm überprüft das Nachrichtenartfeld im MQSeries-Nachrichten-Header und führt die entsprechende Umsetzung durch.

Enthält der MQSeries-Nachrichten-Header den Hinweis, dass es sich um eine MQSeries Everyplace-Nachricht im **funneled**-Format handelt, wird der MQSeries-Nachrichtenhauptteil wieder in die ursprüngliche MQSeries Everyplace-Nachricht umgesetzt und an das MQSeries Everyplace-Netz gesendet.

Handelt es sich nicht um eine MQSeries Everyplace-Nachricht im **funneled**-Format, wird der Inhalt des MQSeries-Nachrichten-Headers extrahiert und in ein **MQeMQMMsgObject** gestellt. Der MQSeries-Nachrichtenhauptteil wird wie ein einfaches Bytefeld gehandhabt und ebenfalls in das Objekt **MQeMQMMsgObject** kopiert. Anschließend wird dieses Objekt an das MQSeries Everyplace-Netz gesendet.

Die Verwendung der Klasse **MQeMQMsgObject** und des Standardumsetzungsprogramms ermöglicht also Folgendes:

- Eine MQSeries Everyplace-Nachricht kann unverändert über ein MQSeries-Netz an ein MQSeries Everyplace-Netz übertragen werden.
- Eine MQSeries-Nachricht kann unverändert von einem MQSeries Everyplace-Netz an ein MQSeries-Netz übertragen werden.
- Eine MQSeries Everyplace-Anwendung kann eine beliebige MQSeries-Anwendung steuern, ohne dass die MQSeries-Anwendung geändert wird.

Funktion

Ferne MQSeries-Warteschlangen ermöglichen das synchrone Einreihen (Put) von MQSeries Everyplace-Nachrichten von einem MQSeries Everyplace-WS-Manager aus; alle anderen Vorgänge im Zusammenhang mit der Nachrichtenübertragung müssen asynchron erfolgen.

MQSeries Everyplace-Verwaltungsnachrichten können nicht an einen MQSeries-WS-Manager gesendet werden, da dieser über keine Verwaltungswarteschlangen verfügt und sich das Format der Verwaltungsnachrichten vom MQSeries-Format unterscheidet.

Kompatibilität

Ein MQSeries Everyplace-Netz kann zwar ohne MQSeries eingesetzt werden, in vielen Fällen sind jedoch für die Anwendungsausführung beide Produkte erforderlich. MQSeries Everyplace kann in ein vorhandenes MQSeries-Netz integriert werden, wobei Kompatibilität in folgenden Punkten gewährleistet sein muss:

- **Adressierung und Benennung:**
 - Identische Adressierungssemantik unter Verwendung einer WS-Manager-/Warteschlangenadresse
 - Einheitliche Verwendung eines ASCII-Zeichenfolgenbereichs
- **Anwendungen:** MQSeries Everyplace unterstützt vorhandene MQSeries-Anwendungen, ohne dass hierzu Änderungen erforderlich sind.
- **Kanäle:** MQSeries Everyplace-Gateways verwenden MQSeries-Client-Kanäle.
- **Nachrichtenaustausch und -inhalt:**
 - Nachrichtenaustausch zwischen MQSeries Everyplace und MQS
 - Transparentes Nachrichtennetz (Nachrichten können ohne Änderungen zwischen MQSeries Everyplace- und MQSeries-Netzen ausgetauscht werden)
 - Gegenseitige Unterstützung gekennzeichnete Felder im MQSeries-Nachrichten-Header
 - Einmalige gesicherte Nachrichtenzustellung
- **Beispielanwendungen:** Interoperabilität der Anwendungen MQSeries PostCard und MQSeries Everyplace PostCard¹¹

11. Diese Anwendung steht in Version 1.0 nicht zur Verfügung, kann jedoch von der MQSeries-Site mit herunterladbarer Software im World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>) heruntergeladen werden.

MQSeries Everyplace Version 1.0 unterstützt nicht alle MQSeries-Funktionen. Einige der wichtigsten Unterschiede (neben umgebungs-, betriebssystem- und übertragungsspezifischen Überlegungen) sind nachfolgend aufgeführt. Es sollte jedoch bedacht werden, dass in MQSeries Everyplace eine Reihe von Möglichkeiten zur Verfügung stehen, mit denen eine Vielzahl von Anwendungsfunktionen ausgeführt werden können - beispielsweise durch die Verwendung von MQSeries Everyplace-Merkmalen, die Nutzung der Unterklassenfunktion, die Verwendung eigener an Stelle der gelieferten Klassen oder durch Nutzung der Regeln, Schnittstellen und anderen Anpassungsmerkmalen, die in diesem Produkt verfügbar sind.

- Keine Cluster-Unterstützung
- Keine Unterstützung von Verteilerlisten
- Keine Unterstützung von Nachrichtensegmentierungen bzw. Anordnung von Nachrichten in Gruppen
- Keine Möglichkeiten zum Lastausgleich bzw. zum Einsatz eines Ersatzsystems (Hot Standby)
- Keine Referenznachrichten
- Keine Berichtsoptionen
- Keine Unterstützung für gemeinsame Warteschlangen
- Keine Auslösemechanismen
- Keine UOW-Unterstützung (UOW = Unit of Work; Arbeitseinheit), keine XA-Koordinierung

Es bestehen außerdem Unterschiede in der Skalierbarkeit und den Leistungsmerkmalen.

Gesicherte Zustellung

Sowohl MQSeries Everyplace als auch MQSeries ermöglichen eine gesicherte Nachrichtenzustellung, allerdings bestehen Unterschiede in der Sicherheitsstufe. Die gesicherte Übertragung einer Nachricht von MQSeries Everyplace an MQSeries kann nur bei der gemeinsamen Verwendung von **putMessage** und **confirmPutMessage** gewährleistet werden (siehe „WS-Managervorgänge“ auf Seite 28). Die Nachrichtenübertragung von MQSeries an MQSeries Everyplace ist nur gesichert, wenn die MQSeries-Nachricht als permanent definiert ist.

Kapitel 7. Anwendungen und Dienstprogramme

Anmerkung: Die folgenden Anwendungen und Dienstprogramme stehen nicht in Version 1.0 zur Verfügung, können jedoch von der MQSeries-Site mit herunterladbarer Software im World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>) heruntergeladen werden.

PostCard

Bei PostCard handelt es sich um eine MQSeries Everyplace-Anwendung, mit der der Betrieb eines Standalone-MQSeries Everyplace-Netzes bzw. die Interaktion zwischen MQSeries Everyplace- und MQSeries-Netzen überprüft werden kann. PostCard ist eine Java-Anwendung, mit der Textnachrichten an einen Benutzer auf einem fernen WS-Manager gesendet werden können. Wird eine PostCard-Nachricht an ein MQSeries-Ziel gesendet, empfängt dieses die Nachricht in Form einer Postkarte (und umgekehrt).

Es ist außerdem eine C-Version von MQSeries Everyplace PostCard erhältlich, die unter PalmOS und mit der Java-Version zusammen eingesetzt werden kann.

MQSeries Everyplace Explorer

Bei MQSeries Everyplace Explorer handelt es sich um ein in Java erstelltes Verwaltungs-Tool, das die Konfiguration und den Zugriff auf lokale und ferne WS-Manager, Warteschlangen und Nachrichten ermöglicht. Es stellt mit Hilfe der Microsoft Foundation Classes eine standardmäßige Windows-GUI zur Verfügung, kann deswegen allerdings nur unter Windows eingesetzt werden. Dieses Dienstprogramm kann jedoch zur Verwaltung aller MQSeries Everyplace-WS-Manager eingesetzt werden, da es ausschließlich auf dem Senden und Empfangen von Verwaltungsnachrichten beruht. Der MQSeries Everyplace Explorer gibt in zwei Teilfenstern eine Sicht des MQSeries Everyplace-Netzes wieder, wobei im linken Teilfenster eine Objektbaumstruktur und im rechten Teilfenster genauere Angaben zu den Objekten angezeigt werden.

MQSeries Everyplace Explorer ermöglicht Folgendes:

- Anzeigen bzw. Ändern der WS-Managermerkmale
- Erstellen, Löschen bzw. Ändern von Verbindungen und die Anzeige ihrer Merkmale
- Erstellen, Löschen bzw. Ändern von Warteschlangen und die Anzeige ihrer Merkmale bzw. ihres Inhalts
- Suchen nach bzw. Löschen von Nachrichten sowie die Anzeige der Nachrichtenmerkmale und Überprüfen der Nachrichtenfelder
- Senden von Testnachrichten
- Konfigurieren der MQSeries Everyplace-Brücke zu MQSeries

MQSeries Everyplace Explorer verwendet in der Regel einen bereits vorhandenen und konfigurierten WS-Manager und kann andere Klassen für die Ausführung laden. Ist kein WS-Manager vorhanden, wird er von MQSeries Everyplace Explorer mit den vom Benutzer vorgegebenen Einstellungen erstellt.

Es können mehrere Kopien dieses Tools auf einer einzelnen Maschine eingesetzt werden, wobei jede Kopie auf einer eigenen JVM zur Ausführung kommt. Dies ermöglicht die Simulation eines MQSeries Everyplace-Netztes und damit die Überprüfung und Demonstration des MQSeries Everyplace-Netzbetriebs.

Kapitel 8. Programmierschnittstellen

Bei der *MQSeries Everyplace-Systemprogrammierschnittstelle* handelt es sich um die Programmierschnittstelle zu MQSeries Everyplace. Es sind je nach Betriebssystem zwei Implementierungen möglich. Die Java-Version ermöglicht den Zugriff auf alle MQSeries Everyplace-Funktionen; die C-Schnittstelle in MQSeries Everyplace Version 1.0 ermöglicht lediglich den Zugriff auf einen Teil dieser Funktionen. Ausführliche Informationen zu Klassen, Methoden und Prozeduren finden Sie im Handbuch *MQSeries Everyplace Programming Reference*, MQSeries Everyplace-Programmierbeispiele im Handbuch *MQSeries Everyplace Programming Guide*.

Anhang. Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Dienstleistungen von IBM verwendet werden können. An Stelle der Produkte, Programme oder Dienstleistungen können auch andere ihnen äquivalente Produkte, Programme oder Dienstleistungen verwendet werden, solange diese keine gewerblichen Schutzrechte der IBM verletzen. Die Verantwortung für den Betrieb von Fremdprodukten, Fremdprogrammen und Fremdservices liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieser Veröffentlichung ist keine Lizenzierung dieser Patente verbunden. Lizenzanfragen sind schriftlich an IBM Europe, Director of Licensing, 92066 Paris La Defense Cedex, France zu richten. Anfragen an obige Adresse müssen auf englisch formuliert werden.

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die Angaben in diesem Handbuch werden in regelmäßigen Zeitabständen aktualisiert. Die Änderungen werden in Überarbeitungen oder in Technical News Letters (TNLs) bekanntgegeben. IBM kann jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Web-Sites anderer Anbieter dienen lediglich als Benutzerinformationen und stellen keinerlei Billigung des Inhalts dieser Web-Sites dar. Das über diese Web-Sites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Web-Sites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängigen, erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

IBM United Kingdom Laboratories,
Mail Point 151,
Hursley Park,
Winchester,
Hampshire
England
SO21 2JN

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des im Handbuch aufgeführten Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt im Rahmen der Allgemeinen Geschäftsbedingungen der IBM, der Internationalen Nutzungsbedingungen der IBM für Programmpakete oder einer äquivalenten Vereinbarung.

Marken

Folgende Namen sind in gewissen Ländern Marken der IBM Corporation:

AIX AS/400 IBM MQSeries OS/390

Microsoft, Windows, Windows NT und das Windows-Logo sind in gewissen Ländern Marken der Microsoft Corporation.

Java sowie alle auf Java basierenden Marken und Logos sind in gewissen Ländern Marken bzw. eingetragene Marken der Sun Microsystems Inc.

Andere Namen von Unternehmen, Produkten oder Dienstleistungen können Marken anderer Unternehmen sein.

Glossar

In diesem Glossar werden die im vorliegenden Handbuch verwendeten Begriffe erläutert sowie Begriffe, die hier in einer anderen als der sonst üblichen Bedeutung verwendet werden. In einigen Fällen gibt es für einen Begriff mehrere Definitionen; es wird jedoch nur jeweils die für dieses Handbuch zutreffende Definition angegeben.

Falls ein gesuchter Begriff nicht im Glossar enthalten ist, sehen Sie im Index oder im *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994, nach.

Anwendungsprogrammierschnittstelle (API).

Eine Anwendungsprogrammierschnittstelle (API; Application Programming Interface) setzt sich aus den Funktionen und Variablen zusammen, die von Programmierern in den von ihnen entwickelten Anwendungen verwendet werden können.

Asynchrone Nachrichtenübertragung. Ein Verfahren zur Kommunikation zwischen Programmen, bei dem Nachrichten in Nachrichtenwarteschlangen gestellt werden. Bei der asynchronen Nachrichtenübertragung setzen die Programme die Verarbeitung fort, ohne auf eine Antwort auf eine Nachricht zu warten. Vgl. *Synchrone Nachrichtenübertragung*.

Authentifizierungsfunktion. Ein Programm, das Sender und Empfänger von Nachrichten überprüft.

Brücke. Ein MQSeries Everyplace-Objekt, das den Austausch von Nachrichten zwischen MQSeries Everyplace und anderen Nachrichtenübertragungssystemen, wie beispielsweise MQSeries, ermöglicht.

Client. In MQSeries eine Laufzeitkomponente, die lokalen Benutzeranwendungen Zugriff auf Services zur Steuerung von Warteschlangen auf einem Server ermöglicht.

Dynamischer Kanal. Ein dynamischer Kanal verbindet MQSeries Everyplace-Einheiten miteinander und ermöglicht die bidirektionale Übertragung synchroner und asynchroner Nachrichten bzw. Antworten.

Gateway. Bei einem MQSeries Everyplace-Gateway oder Server handelt es sich um einen Computer, auf dem neben dem MQSeries Everyplace-Code auch ein Kanalmanager installiert ist.

Geschützt. Ein geschütztes Feld ist nur innerhalb der eigenen Klasse, Unterklasse bzw. in Paketen, die die Klasse enthalten, sichtbar.

Hypertext Markup Language (HTML). Eine Sprache zur Definition von Informationen, die im World Wide Web angezeigt werden.

Instanz. Ein Objekt. Es besteht die Möglichkeit, ein Objekt für eine Klasse anzulegen, das als Instanz dieser Klasse bezeichnet wird.

Internet. Ein öffentliches Netz für die gemeinsame Nutzung von Informationen. Physisch greift das Internet auf einen Teil der Ressourcen zu, die über die derzeit vorhandenen öffentlichen Telekommunikationsnetze zur Verfügung stehen. In technischer Hinsicht zeichnet sich das Internet als gemeinschaftliches öffentliches Netz durch die Verwendung des TCP/IP-Protokolls (Transport Control Protocol/Internet Protocol) aus.

Java Developer's Kit (JDK). Ein von Sun Microsystems für Java-Entwickler vertriebenes Softwarepaket. Es umfasst den Java-Interpreter sowie Java-Entwicklungs-Tools: Compiler, Debugger, Disassembler, Appletviewer, Stub File Generator und Dokumentations-Tool.

Java Naming and Directory Service (JNDI). Eine mit Java erstellte API. Sie stellt Namens- und Verzeichnisfunktionen für Java-Anwendungen zur Verfügung.

Kanal. Siehe *Dynamischer Kanal* und *MQI-Kanal*.

Kanalmanager. Ein MQSeries Everyplace-Objekt, das eine größere Anzahl gleichzeitiger logischer Übertragungsleitungen zwischen Endpunkten unterstützt.

Kapselung. Hierbei handelt es sich um eine objektorientierte Programmiertechnik, bei der Daten ausschließlich einem bestimmten Objekt zugeordnet werden und auf die nur über Methodenaufrufe zugegriffen werden kann.

Klasse. Eine gekapselte Gruppe von Daten und Methoden, die zur Datenverarbeitung verwendet wird. Es besteht die Möglichkeit zur Erstellung von Klasseninstanzen, bei denen es sich um Objekte handelt.

Komprimierungsfunktion. Ein Programm zur Komprimierung von Nachrichten, das das zu übertragende Datenvolumen reduziert.

Lightweight Directory Access Protocol (LDAP). Ein Client/Server-Protokoll für den Zugriff auf Verzeichnisservices.

Methode. Bezeichnet in der objektorientierten Programmierung eine Funktion oder Prozedur.

MQI-Kanal. Verbindet einen MQSeries-Client mit einem WS-Manager auf einem Server und ermöglicht die bidirektionale Übertragung von MQI-Aufrufen und -Antworten.

MQSeries. Eine Gruppe von IBM Lizenzprogrammen, die Services für die Steuerung von Nachrichtenwarteschlangen zur Verfügung stellen.

Nachricht. In Anwendungen für die Steuerung von Nachrichtenwarteschlangen der Datenaustausch zwischen Programmen.

Nachrichtenwarteschlange. Siehe *Warteschlange*.

Objekt. (1) In Java eine Instanz einer Klasse. Eine Klasse ist das Abbild einer Gruppe von Elementen, ein Objekt das Abbild eines bestimmten Elements dieser Gruppe. (2) In MQSeries ein WS-Manager, eine Warteschlange oder ein Kanal.

Öffentlich. Eine öffentliche Klasse bzw. Schnittstelle ist allgemein sichtbar. Eine öffentliche Methode bzw. Variable wiederum ist überall dort sichtbar, wo auch die zugehörige Klasse sichtbar ist.

Paket. Über Pakete wird in Java Teilen des Java-Codes der Zugriff auf bestimmte Klassen-Gruppen ermöglicht. Damit erhält der Code-Teil in einem bestimmten Paket Zugriff auf alle Klassen in diesem Paket sowie auf alle nicht privaten Methoden und Felder dieser Klassen.

Personal Digital Assistant (PDA). Ein Taschen-PC.

Privat. Ein privates Feld ist nur innerhalb der zugehörigen Klasse sichtbar.

Schnittstelle. Eine Klasse, die nur abstrakte Methoden, jedoch keine Exemplarvariablen enthält. Eine Schnittstelle stellt eine einheitliche Gruppe von Methoden zur Verfügung, die in Form von Unterklassen verschiedener Klassen implementiert werden können.

Server. (1) Bei einem MQSeries Everyplace-Server handelt es sich um eine Einheit, auf der ein MQSeries Everyplace-Kanalmanager konfiguriert ist. (2) Bei einem MQSeries-Server handelt es sich um einen WS-Manager, der Client-Anwendungen auf fernen Workstations Services zur Steuerung von Nachrichtenwarteschlangen zur Verfügung stellt. (3) Allgemein ein Programm, das in Client/Server-Modellen auf Datenanforderungen reagiert. (4) Der Computer, auf dem ein Server-Programm zur Ausführung kommt.

Servlet. Ein Java-Programm, das nur auf einem Web-Server ausgeführt wird.

Steuerung von Nachrichtenwarteschlangen. Eine Programmiertechnik, bei der ein Programm einer Anwendung mit den anderen Programmen kommuniziert, indem Nachrichten in Warteschlangen eingereiht werden.

Superklasse. Eine Klasse, die durch eine andere Klasse erweitert wurde. Alle öffentlichen und geschützten Methoden bzw. Variablen der Superklasse stehen auch der Unterklasse zur Verfügung.

Synchrone Nachrichtenübertragung. Ein Verfahren zur Kommunikation zwischen Programmen, bei dem Nachrichten in Nachrichtenwarteschlangen gestellt werden. Bei der synchronen Nachrichtenübertragung setzt das Programm, von

dem eine Nachricht abgesetzt wurde, die Verarbeitung erst nach Empfang einer Antwort fort. Vgl. *Asynchrone Nachrichtenübertragung*.

Transmission Control Protocol/Internet Protocol (TCP/IP). Eine Gruppe von Übertragungsprotokollen, die Peer-zu-Peer-Konnektivitätsfunktionen für lokale und Weitverkehrsnetze (LANs und WANs) unterstützen.

Unterklasse. Eine Klasse, die zur Erweiterung einer anderen Klasse dient. Die Unterklasse übernimmt alle öffentlichen und geschützten Methoden und Variablen der zugehörigen Superklasse.

Verschlüsselungsfunktion. Ein Programm, das Nachrichten verschlüsselt und damit eine sichere Übertragung gewährleistet.

Warteschlange. Ein MQSeries-Objekt. Anwendungen für die Steuerung von Warteschlangen können Nachrichten in Warteschlangen stellen und aus diesen abrufen.

Web. Siehe *World Wide Web*.

Web-Browser. Ein Programm zur Formatierung und Anzeige von Informationen im World Wide Web.

World Wide Web (Web). Ein Internet-Service, der auf einer einheitlichen Protokollgruppe basiert und bestimmten, hierfür konfigurierten Servern die einheitliche Verteilung von Dokumenten im Internet ermöglicht.

WS-Manager. Ein Systemprogramm, das Anwendungen Services für die Steuerung von Nachrichtenwarteschlangen zur Verfügung stellt.

Literaturverzeichnis

Referenzliteratur

- *MQSeries Everyplace Wichtige Hinweise*, GC12-2861-00
- *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 Einstieg*, GC12-2642-00

Index

A

Adapter, MQSeries Everyplace 35
Anforderungen, kundenspezifische 13
Anpassung 49
Anwendung, PostCard 67
Anwendungen 67
Anwendungen, laden 53
Anwendungen, MQSeries Everyplace 12
Asynchrone Nachrichtenübertragung 26, 39
Attributsregeln 49
Ausgabedaten, Format 20
Ausgangs-Server, MQSeries Everyplace 23
Ausgangs-Server-Warteschlangen 23
Authentifizierbare Definitionseinheiten 46
Automatische Registrierung 47

B

Begriffe x
Beschreibung 1
Betriebssysteme, unterstützte 3
Brücke, MQS 55
Brückenobjekt 57

C

Client, MQSeries 6
Client-Kanäle 7, 15
Client/Server-Verbindung 52

D

Definitionseinheiten, authentifizierbar 46
Dienstprogramm, MQSeries Everyplace Explorer 67
Dienstprogramme 67
Dynamische Kanäle 7, 15, 33

E

Einheiten, MQSeries Everyplace 7, 15
Empfangsprogrammobjekt 60, 62
Ereignisprotokoll 36
Erforderliche Betriebssysteme 3

F

Ferne Warteschlangen 22
Format der Ausgabedaten 20

G

Gateways, MQSeries Everyplace 7, 15
Gesicherte Nachrichtenzustellung 66

H

Hinweise, rechtliche 71
Host-Nachrichtenübertragung 6

K

Kanäle, Client 7, 15
Kanäle, dynamisch 7, 15, 33

Kanalempfangsprogramm 51
Kanalmanager 51
Klassen, MQSeries Everyplace 53
Kompatibilität mit MQSeries 65
Komprimierung 40
Konfiguration 49
Konfigurationen, MQSeries Everyplace 36
Konzepte, Produkt 15
Kundenanforderungen 13

L

Laden, Anwendungen 53
Leistungsspektrum 11
Lokale Sicherheit 42
Lokale Warteschlangen 21

M

Marken 72
Mehrere Verbindungsarten 52
Mini Certificate, Service für die Vergabe 48
Mini Certificates 46
MQeAttribute 44
MQeMTrustAttribute 44
MQSeries, Kompatibilität mit 65
MQSeries, Schnittstelle zu 55
MQSeries-Brücke 55
MQSeries-Brückenregeln 49
MQSeries-Brückenwarteschlangen 23
MQSeries-Client 6
MQSeries Everyplace, Registrierungsdatenbank 15
MQSeries Everyplace-Adapter 35
MQSeries Everyplace-Anwendungen 12, 67
MQSeries Everyplace-Dienstprogramme 67
MQSeries Everyplace-Einheiten 7, 15
MQSeries Everyplace Explorer, Dienstprogramm 67
MQSeries Everyplace-Gateways 7, 15
MQSeries Everyplace-Klassen 53
MQSeries Everyplace-Konfigurationen 36
MQSeries Everyplace-Netze 36, 55
MQSeries Everyplace-Objekte 16
MQSeries Everyplace-Regeln 49
MQSeries Everyplace-Registrierungsdatenbank 45
MQSeries Everyplace-Sicherheit 40
MQSeries Everyplace-Verwaltung 30
MQSeries Everyplace-Verwaltungsdienstprogramm 67
MQSeries Everyplace-Warteschlangen 21
MQSeries Everyplace-WS-Manager 26
MQSeries Integrator 5
MQSeries-Nachrichtenübertragung 5
MQSeries-Netze 55
MQSeries-Produktfamilie 5
MQSeries-Server 6
MQSeries Workflow 5

N

Nachrichten, Verwaltung 30
Nachrichtenbasierte Sicherheit 44
Nachrichtenobjekte 16
Nachrichtenübertragung, asynchron 26, 39
Nachrichtenübertragung, MQSeries 5
Nachrichtenübertragung, synchron 26, 40
Nachrichtenumsetzung 64
Nachrichtenzustellung, gesichert 66
Netze, MQS 55
Netze, MQSeries Everyplace 36, 55

O

Objekte, MQSeries Everyplace 16
Objekte, Nachrichten 16
Öffentliche Registrierungsdatenbank 47

P

Peer-zu-Peer-Verbindung 52
PostCard, Anwendung 67
Private Registrierungsdatenbank 46
Produktkonzepte 15
Programmierschnittstellen 69
Proxy-WS-Managerobjekt 58

R

RAS-Wählprogrammregeln 50
Rechtliche Hinweise 71
Regeln, MQSeries Everyplace 49
Registrierungsdatenbank 45
Registrierungsdatenbank, MQSeries Everyplace 15
Registrierungsdatenbank, öffentlich 47
Registrierungsdatenbank, privat 46
Replikation von Zertifikaten 47

S

Schnittstelle, Sicherheit 49
Schnittstellen, Programmierung 69
Schnittstellen zu MQSeries 55
Server, MQSeries 6
Sicherheit, auf Nachrichtenebene 44
Sicherheit, auf Warteschlangenebene 42
Sicherheit, lokal 42
Sicherheit, MQSeries Everyplace 40
Sicherheitsschnittstelle 49
Skalierbarkeit 36
Speichern und Weiterleiten, Warteschlangen 22
Synchrone Nachrichtenübertragung 26, 40
Systemprogrammierschnittstelle 69

T

Trace, MQSeries Everyplace 35

U

Übersicht 1
Übertragung 51

Überwachung 33
Umsetzungsprogramme 64
Universelle (Pervasive) Nachrichtenübertragung 6
Unterstützte Betriebssysteme 3

V

Verbindung, Client/Server 52
Verbindung, Peer-zu-Peer 52
Verbindungsarten 51
Verbindungsarten, mehrere 52
Vergabeservice für Mini Certificates 48
Verschlüsselung 40
Verteilte Nachrichtenübertragung 6
Verwaltung mit MQSeries Everyplace 30
Verwaltung von Einwahlverbindungen 35
Verwaltungsdienstprogramm 67
Verwaltungsnachrichten 30
Verwaltungswarteschlangen 23
Voraussetzungen 3
Vorgänge, WS-Manager 28
Vorkenntnisse ix

W

Warteschlangen, ferne 22
Warteschlangen, lokale 21
Warteschlangen, MQSeries-Brücke 23
Warteschlangen, MQSeries Everyplace 21
Warteschlangen, Speichern und Weiterleiten 22
Warteschlangen, Verwaltung 23
Warteschlangenbasierte Sicherheit 42
Warteschlangenregeln 50
Workstation-Nachrichtenübertragung 6
WS-Manager 6, 51
WS-Manager, MQSeries Everyplace 26
WS-Managerregeln 50
WS-Managervorgänge 28

Z

Zertifikatreplikation 47
Zielgruppe ix
Zu diesem Handbuch ix

Antwort

**MQSeries Everyplace
Einführung
Version 1**

IBM Form GC12-2860-00

Anregungen zur Verbesserung und Ergänzung dieser Veröffentlichung nehmen wir gerne entgegen. Bitte informieren Sie uns über Fehler, ungenaue Darstellungen oder andere Mängel.

Zur Klärung technischer Fragen sowie zu Liefermöglichkeiten und Preisen wenden Sie sich bitte entweder an Ihre IBM Geschäftsstelle, Ihren IBM Geschäftspartner oder Ihren Händler.

Unsere Telefonauskunft "HALLO IBM" (Telefonnr.: 01803/31 32 33) steht Ihnen ebenfalls zur Klärung allgemeiner Fragen zur Verfügung.

Kommentare:

Danke für Ihre Bemühungen.

Sie können ihre Kommentare betr. dieser Veröffentlichung wie folgt senden:

- Als Brief an die Postanschrift auf der Rückseite dieses Formulars
- Als E-Mail an die folgende Adresse: comment@tvm.vnet.ibm.com

Name

Adresse

Firma oder Organisation

Rufnummer

E-Mail-Adresse

Antwort
GC12-2860-00



IBM Deutschland Informationssysteme GmbH
SW NLS Center

70548 Stuttgart



Printed in Denmark

GC12-2860-00

