

MQSeries<sup>®</sup> Everyplace



# Introduzione

*Versione 1*



MQSeries<sup>®</sup> Everyplace



# Introduzione

*Versione 1*

## Attenzione

Prima di utilizzare queste informazioni e il relativo prodotto, leggere attentamente le informazioni generali in "Appendice. Informazioni particolari" a pagina 61

### Avvertenza sulle licenze

MQSeries Everyplace Versione 1 è un toolkit che consente agli utenti di scrivere applicazioni MQSeries Everyplace e di creare un ambiente in cui eseguirle.

Le condizioni di licenza in cui viene acquistato il toolkit determinano l'ambiente in cui è possibile utilizzarlo:

*Se MQSeries Everyplace viene acquistato come **dispositivo (client)** non potrà essere utilizzato per creare un **programma di gestione canali di MQSeries Everyplace** o un **listener di canali MQSeries Everyplace**.*

*La presenza di un **programma di gestione canali di MQSeries Everyplace** o di un **listener di canali di MQSeries Everyplace** definisce un ambiente a **gateway (server)**, che richiede una licenza per gateway.*

### Prima edizione (giugno 2000)

Questa edizione si riferisce a MQSeries Everyplace Versione 1.0 e a tutti i rilasci e le modifiche successive, salvo indicato diversamente in edizioni future.

Come ultima pagina del manuale è stato predisposto un foglio riservato ai commenti del lettore. Se il foglio è stato rimosso, i commenti possono essere inviati al seguente indirizzo:

Selfin S.p.A  
Translation Assurance  
via F. Giordani, 7  
80122 - NAPOLI

# Indice

<b>Figure</b> . . . . .	v
<b>Tabelle</b> . . . . .	vii
<b>Informazioni su questa pubblicazione</b> . . . . .	ix
A chi si rivolge questa pubblicazione . . . . .	ix
Prerequisiti . . . . .	ix
Termini utilizzati in questa pubblicazione. . . . .	ix
<b>Capitolo 1. Panoramica</b> . . . . .	1
Note per la versione 1.0 . . . . .	1
<b>Capitolo 2. Prerequisiti</b> . . . . .	3
<b>Capitolo 3. Famiglia MQSeries</b> . . . . .	5
<b>Capitolo 4. Requisiti.</b> . . . . .	11
Caratteristiche . . . . .	11
Applicazioni . . . . .	11
Requisiti del cliente . . . . .	12
<b>Capitolo 5. Contenuti del prodotto</b> . . . . .	13
Introduzione. . . . .	13
Oggetti messaggi . . . . .	14
Formato dei dati dump . . . . .	17
Code . . . . .	18
Programmi di gestione code . . . . .	22
Operazioni dei programmi di gestione code . . . . .	24
Amministrazione . . . . .	25
Messaggi di amministrazione . . . . .	26
Amministrazione selettiva . . . . .	28
Programma di controllo e operazioni relative . . . . .	28
Canali dinamici. . . . .	28
Adattatori . . . . .	29
Gestione dei collegamenti dialup . . . . .	30
Traccia . . . . .	30
Registrazione eventi . . . . .	30
Reti MQSeries Everyplace . . . . .	30
Configurazioni e scalabilità . . . . .	30
Consegna asincrona dei messaggi . . . . .	32
Consegna sincrona dei messaggi . . . . .	33
Sicurezza . . . . .	33
Sicurezza locale MQSeries Everyplace . . . . .	34
Sicurezza MQSeries Everyplace basata sulle code . . . . .	35
Sicurezza a livello di messaggio . . . . .	36
Il registro. . . . .	37
Entità autenticabili MQSeries Everyplace . . . . .	38

Registro privato e credenziali . . . . .	38
Autoregistrazione . . . . .	38
Registro pubblico e replica di certificati . . . . .	39
Utilizzo applicazioni dei servizi di registro . . . . .	39
Servizio di creazione mini-certificati predefiniti . . . . .	39
L'interfaccia di sicurezza . . . . .	40
Configurazione e personalizzazione . . . . .	40
Regole . . . . .	40
Stili di collegamento . . . . .	42
Collegamento peer-to-peer . . . . .	42
Collegamento client-server . . . . .	43
Stili di collegamento multipli . . . . .	43
Classi . . . . .	43
Caricamento applicazioni . . . . .	44
<b>Capitolo 6. Reti MQSeries Everyplace e MQSeries . . . . .</b>	<b>45</b>
Interfaccia per MQSeries . . . . .	45
Conversione messaggi . . . . .	52
Funzione . . . . .	54
Compatibilità . . . . .	54
Consegna assicurata . . . . .	55
<b>Capitolo 7. Applicazioni e programmi di utilità . . . . .</b>	<b>57</b>
Postcard . . . . .	57
MQSeries Everyplace Explorer . . . . .	57
<b>Capitolo 8. Interfacce di programmazione . . . . .</b>	<b>59</b>
<b>Appendice. Informazioni particolari . . . . .</b>	<b>61</b>
Marchi . . . . .	62
<b>Glossario . . . . .</b>	<b>63</b>
<b>Bibliografia. . . . .</b>	<b>67</b>
<b>Indice analitico . . . . .</b>	<b>69</b>

---

## Figure

1.	Famiglia MQSeries . . . . .	5
2.	Configurazioni semplici per host e distribuite . . . . .	6
3.	Configurazioni tipiche per workstation . . . . .	7
4.	Configurazioni tipiche per dispositivi . . . . .	8
5.	Reti MQSeries Everyplace semplici . . . . .	31
6.	Una rete MQSeries Everyplace a stella . . . . .	32
7.	Gerarchia oggetti di collegamento MQSeries . . . . .	46



---

## Tabella

1.	Gli ambienti software supportati per la versione 1 . . . . .	3
2.	Elementi di MQSeries Everyplace e MQSeries . . . . .	9
3.	Oggetti campi e proprietà . . . . .	15
4.	Proprietà degli oggetti attributi . . . . .	15
5.	Proprietà degli oggetti messaggi . . . . .	16
6.	Campi di oggetti messaggi forniti . . . . .	16
7.	Proprietà delle code . . . . .	20
8.	Proprietà dei programmi di gestione code locali . . . . .	23
9.	Proprietà di collegamento (programma di gestione code remote) . . . . .	23
10.	Operazioni di messaggistica su code MQSeries Everyplace . . . . .	25
11.	Classi di messaggi di amministrazione . . . . .	26
12.	Struttura generica di un messaggio di amministrazione. . . . .	26
13.	Supporto per autenticazione, codifica e compressione . . . . .	34
14.	Stili di collegamento . . . . .	42
15.	Opzioni di classe . . . . .	43
16.	Proprietà degli oggetti collegamenti . . . . .	46
17.	Proprietà collegamenti . . . . .	48
18.	Proprietà proxy del programma di gestione code MQSeries . . . . .	48
19.	Proprietà del servizio di collegamento al client . . . . .	49
20.	Proprietà listener . . . . .	50
21.	Proprietà delle code remote di MQSeries . . . . .	51



---

## Informazioni su questa pubblicazione

Questa pubblicazione costituisce un'introduzione generale a MQSeries Everyplace e tratta del contenuto del prodotto e delle relazioni con altri prodotti MQSeries.

Per informazioni dettagliate sull'API di MQSeries Everyplace e su come utilizzarla per creare applicazioni MQSeries Everyplace, consultare *Riferimento per la programmazione di MQSeries Everyplace* e *Manuale per la programmazione di MQSeries Everyplace*.

---

## A chi si rivolge questa pubblicazione

Questa pubblicazione è rivolta a tutti gli utenti che siano interessati a sistemi di messaggistica sicura su dispositivi leggeri come sensori, telefoni, PDA (Personal Digital Assistant) e computer laptop.

---

## Prerequisiti

Non è richiesta alcuna preparazione di base per leggere queste informazioni, ma una conoscenza iniziale dei concetti della messaggistica sicura costituirà un valido aiuto.

In caso contrario, si consiglia di leggere le seguenti pubblicazioni di MQSeries:

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

Queste pubblicazioni sono disponibili in linea presso la sezione delle pubblicazioni della libreria in linea di MQSeries. Si accede alla libreria dal sito Web di MQSeries all'indirizzo URL <http://www.ibm.com/software/ts/MQSeries/library/>

---

## Termini utilizzati in questa pubblicazione

I seguenti termini sono utilizzati nell'arco di questa pubblicazione:

### **Famiglia MQSeries**

si riferisce alla raccolta di prodotti MQSeries descritti in "Capitolo 3. Famiglia MQSeries" a pagina 5.

### **Messaggistica MQSeries**

si riferisce ai quattro gruppi di prodotti di messaggistica descritti in "Capitolo 3. Famiglia MQSeries" a pagina 5.

### **MQSeries**

si riferisce ai tre gruppi di prodotti di messaggistica MQSeries:

- Messaggistica distribuita
- Messaggistica host
- Messaggistica per workstation

### **MQSeries Everyplace**

si riferisce al quarto gruppo di messaggistica MQSeries, sulla messaggistica più diffusa.

### **Dispositivo**

Un computer di qualsiasi dimensione che esegue programmi MQSeries Everyplace ma *non dispone di un **programma di gestione canali di MQSeries Everyplace** o di un **listener di canali di MQSeries Everyplace*** installato.

**Nota:** Ai fini della licenza *dispositivo* è sinonimo di *client MQSeries Everyplace*.

### **Gateway**

Un computer di qualsiasi dimensione che esegue programmi MQSeries Everyplace e che *dispone di un **programma di gestione canali di MQSeries Everyplace** o di un **listener di canali di MQSeries Everyplace*** installato.

**Nota:** Ai fini della licenza *gateway* è sinonimo di *server MQSeries Everyplace*.

---

## Capitolo 1. Panoramica

MQSeries Everyplace è un membro della famiglia di prodotti per la messaggistica aziendale MQSeries. E' progettato per soddisfare non solo le esigenze di messaggistica di dispositivi leggeri come sensori, telefoni, PDA (Personal Digital Assistant) e computer laptop, ma anche le richieste di apparecchiature mobili e le esigenze che riguardano l'impiego di reti di comunicazione di tipo non protetto. Il prodotto fornisce la consegna standard MQSeries assicurata per una volta sola e consente di scambiare messaggi con altri membri di prodotti della famiglia. Poiché sono molte le applicazioni MQSeries Everyplace che si eseguono al di fuori della protezione di un firewall su Internet, fornisce anche delle sofisticate funzioni di sicurezza.

I dispositivi leggeri richiedono che il sistema di messaggistica secondario utilizzi in modo accurato le risorse del sistema e, in base a MQSeries Everyplace, migliori l'efficienza dei protocolli e la struttura del sistema. Questo sistema non è dotato di funzioni uguali a quelle degli altri membri di messaggistica della famiglia MQSeries, ma consente di svolgere operazioni incrociate. MQSeries Everyplace consente di utilizzare apparecchi mobili, roaming, accesso ai messaggi locali e remoti, sicurezza e supporto per messaggistica su rete non protette.

MQSeries Everyplace è un membro della famiglia di sistemi informatici globali IBM® ed è progettato per integrarsi perfettamente con gli altri prodotti IBM diffusi e senza fili.

---

### Note per la versione 1.0

- La versione 1.0 di MQSeries Everyplace è un toolkit che consente agli utenti di scrivere applicazioni MQSeries Everyplace e di creare un ambiente in cui eseguirle.
- In questo rilascio, lo sviluppo di MQSeries Everyplace per dispositivi diffusi è l'obiettivo dell'applicazione e del provider di soluzioni.



---

## Capitolo 2. Prerequisiti

Tabella 1 indica i requisiti software per poter eseguire MQSeries Everyplace Versione 1.1

*Tabella 1. Gli ambienti software supportati per la versione 1*

	<b>Sistema operativo</b>
Dispositivo	EPOC
	Sistema operativo Palm
	Windows CE
	Windows® 95 Windows 98 Windows NT v4 Windows 2000
Gateway	Windows NT v4 Windows 2000
<b>Note:</b> <ol style="list-style-type: none"><li>1. La versione 1.0 viene fornita in linguaggio Java per essere utilizzata su tutte le piattaforme che supportano Java.</li><li>2. Un client con funzioni limitate che fornisce solo accesso sincrono alle code remote, è disponibile come codice base C per essere utilizzato solo con il sistema operativo Palm.</li><li>3. Si consiglia Java 1.1, al livello più aggiornato disponibile per la piattaforma. Vedere il sito Web di MQSeries Everyplace (<a href="http://www.ibm.com/software/mqseries/everyp1ace">www.ibm.com/software/mqseries/everyp1ace</a>) per dettagli sui livelli testati di Java.</li></ol>	

---

1. Il codice di MQSeries Everyplace può essere utilizzato su qualsiasi dispositivo che utilizza Java®, ma è stato testato solo con i sistemi operativi riportati in Tabella 1.



## Capitolo 3. Famiglia MQSeries

La famiglia MQSeries comprende una serie di prodotti che offrono un'ampia gamma di funzioni, come illustrato in Figura 1

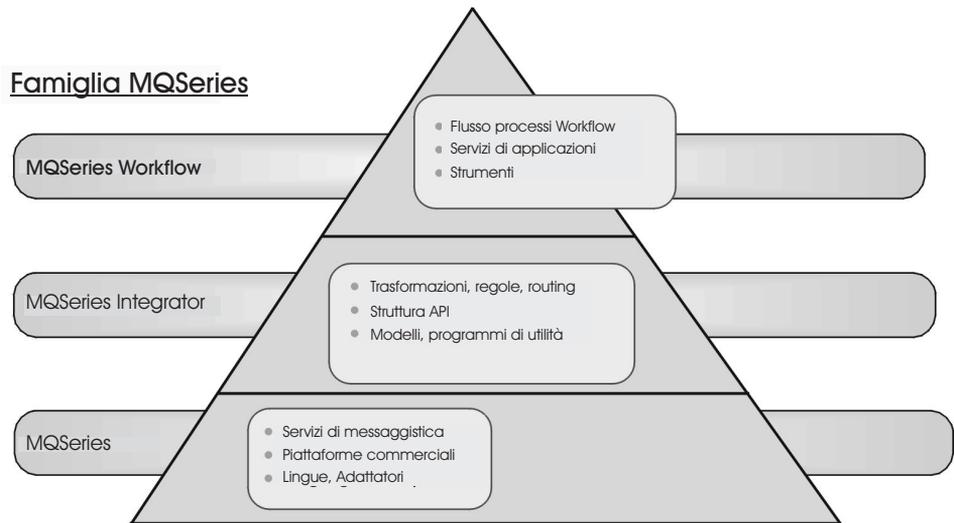


Figura 1. Famiglia MQSeries

- **MQSeries Workflow** semplifica l'integrazione di un'intera azienda con l'automazione dei processi aziendali, che coinvolga il personale e le applicazioni
- **MQSeries Integrator** è un software potente per la messaggistica che consente l'accodamento di messaggi in tempo reale e basato su criteri intelligenti, oltre che la trasformazione e la formattazione del contenuto
- **MQSeries Messaging** consente connessioni any-to-any da desktop a mainframe, attraverso un sistema di messaggistica aziendale con oltre 35 piattaforme supportate

Sia MQSeries Workflow che MQSeries Integrator sono prodotti che sfruttano le connessioni fornite dal sistema di messaggistica di MQSeries.

La messaggistica di MQSeries è fornita dai prodotti MQSeries (MQS) e MQSeries Everyplace; ognuno è progettato per supportare una o più piattaforme hardware server e/o sistemi operativi associati. Data la gran varietà di funzioni di piattaforma, questi singoli prodotti sono organizzati in gruppi e riflettono funzioni e progetti comuni.

Esistono quattro gruppi di prodotti:

- **Messaggistica distribuita:** MQSeries per Windows NT, AIX®, AS/400®, HP-UX, Sun Solaris e altre piattaforme
- **Messaggistica host:** MQSeries per OS/390®
- **Messaggistica per workstation:** MQSeries per Windows 3.1, 95, 98

- **Messaggistica diffusa:** MQSeries Everyplace

a messaggistica si basa su programmi di gestione code, a prescindere dal singolo prodotto o gruppo di prodotti. Questi programmi gestiscono code che a loro volta possono contenere messaggi. Le applicazioni comunicano con un programma di gestione code locale e ricevono o memorizzano i messaggi sulle code. Se un messaggio viene memorizzato su una coda remota, gestita da un relativo programma remoto, il messaggio verrà trasmesso su canali al programma di gestione code remoto. In questo modo i messaggi possono passare attraverso vari programmi di gestione intermedi prima di arrivare a destinazione. L'obiettivo della messaggistica consiste nel separare l'applicazione di invio da quella di ricezione, accodando i messaggi in viaggio se necessario. Tutti i prodotti di messaggistica di MQSeries riguardano gli stessi elementi di base per programmi di gestione code, code, messaggi e canali, nonostante le tante differenze nel dettaglio.

I prodotti MQSeries per host e messaggistica distribuita sono utilizzati per supportare varie configurazioni di rete diverse, tutte riguardanti client e server<sup>2</sup>Alcuni esempi sono illustrati in Figura 2.

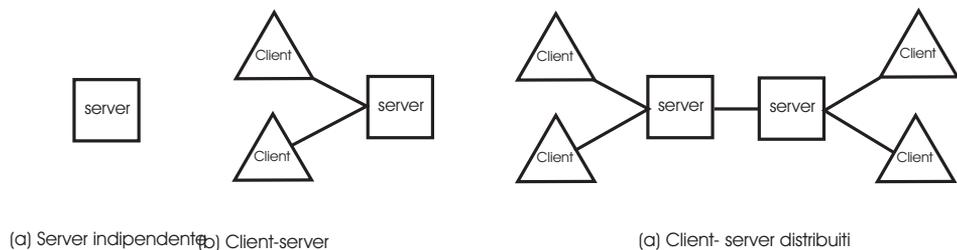


Figura 2. Configurazioni semplici per host e distribuite

Nel caso più semplice un server indipendente viene configurato eseguendo un programma di gestione code. Sul server vengono eseguite una o più applicazioni, scambiando messaggi tramite code. Una configurazione alternativa è quella client-server, in cui il programma di gestione code esiste solo sul server, ma ciascun client ne ha accesso tramite un canale client. Il canale client è un collegamento per comunicazioni bidirezionali che funziona con un protocollo MQSeries unico, implementando qualcosa di simile a un RPC (remote procedure call). E' possibile eseguire le applicazioni sui client e accedere alle code dei server. Un vantaggio della configurazione client-server è dato dal fatto che l'infrastruttura della messaggistica client è leggera, in quanto dipende dal programma di gestione code del server. Uno degli svantaggi invece consiste nel fatto che i client e il relativo server funzionano sincronicamente, per cui il canale del client deve essere sempre disponibile.

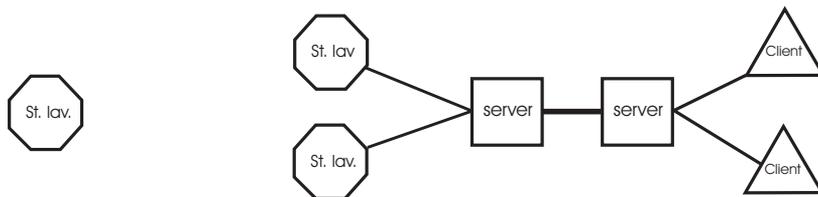
La configurazione distribuita client-server mostra un caso più complesso, riguardante più server. In queste configurazioni i server scambiano messaggi attraverso appositi canali. I canali per i messaggi sono unidirezionali, con un protocollo studiato per uno

2. Questi termini hanno un significato specifico all'interno di MQSeries che non sempre corrisponde al loro utilizzo più diffuso. In questa pubblicazione vengono utilizzati sempre secondo la semantica MQSeries.

scambio sicuro e asincrono dei dati di messaggi. Non è necessario che questi canali di messaggi siano disponibili ai client per continuare a funzionare, ma nessun messaggio può essere trasmesso tra i server quando le comunicazioni non sono disponibili.

I prodotti di messaggistica per workstation MQSeries offrono una serie inferiore di queste opzioni di configurazione. Invece dei server, supportano workstation dotate di un programma di gestione code ma non supportano il collegamento ai client. Le workstation possono essere tuttavia collegate ad altre workstation e anche ai server, tramite canali di messaggi di MQSeries; quindi sono considerate in genere come dei server leggeri e vengono utilizzate in sostituzione ai client quando è richiesta una funzione asincrona.

Due configurazioni tipiche per workstation sono visualizzate in Figura 3. In (b) le applicazioni per workstation possono essere eseguite indipendentemente da server e client:



(A) Staz. di lavoro indipendente

(B) Client/st. lav./server distribuito

Figura 3. Configurazioni tipiche per workstation

Il prodotto MQSeries Everyplace per la messaggistica flessibile supporta le configurazioni attraverso l'impiego di dispositivi e gateway.

Il dispositivo MQSeries Everyplace è un computer che esegue il codice MQSeries Everyplace *senza un programma di gestione canali*. Quindi un dispositivo è limitato alle comunicazioni con un solo dispositivo o gateway alla volta. I dispositivi MQSeries Everyplace possono essere molto piccoli (per esempio un sensore o tubo per olio), più grandi come un telefono, un PDA (personal data assistant) o un computer laptop, fino a grandi macchinari per desktop e workstation. In genere questi tipi di computer vengono definiti flessibili, anche se questo implica una limitazione di dimensioni e capacità che non è presente in realtà nel prodotto.

Un *gateway* è un computer che esegue il codice MQSeries Everyplace *con un programma di gestione canali di MQSeries Everyplace oppure un listener di canali MQSeries Everyplace configurato*. Esso offre tutte le funzioni di un codice di dispositivo, oltre alla capacità di comunicare con più gateway di dispositivi contemporaneamente.. I gateway possono fornire inoltre il meccanismo per scambiare messaggi tra una rete MQSeries Everyplace e una rete MQSeries.

I dispositivi possono inizialmente combinare molti degli attributi di client e server: possono essere configurati con una funzione completa di accodamento che consente di

svolgere operazioni asincrone. E' possibile inoltre accedere alle code in remoto, una funzione che presenta dei punti in comune con l'accesso dei client alle code dei server. Diversamente dai server, i dispositivi non possono essere collegati ai client. I dispositivi possono comunicare direttamente tra di loro grazie alla funzione di messaggistica peer-to-peer. Possono comunicare inoltre tramite canali, ma questi sono unici in MQSeries Everyplace e vengono definiti *canali dinamici* per poterli distinguere dai canali dei client di MQSeries e dai canali di messaggi di MQSeries. I canali dinamici sono bidirezionali e supportano la gamma completa di funzioni fornite da MQSeries Everyplace, compresa la messaggistica sincrona e asincrona.

I gateway devono necessariamente supportare i canali dinamici di MQSeries Everyplace per poter comunicare con i dispositivi. Possono supportare, facoltativamente, i canali client di MQSeries per poter comunicare con i server. Come per i server, i gateway sono dotati di programmi di gestione code, quindi sono in grado di supportare applicazioni locali di messaggistica.

Alcune configurazioni flessibili tipiche sono indicate in Figura 4

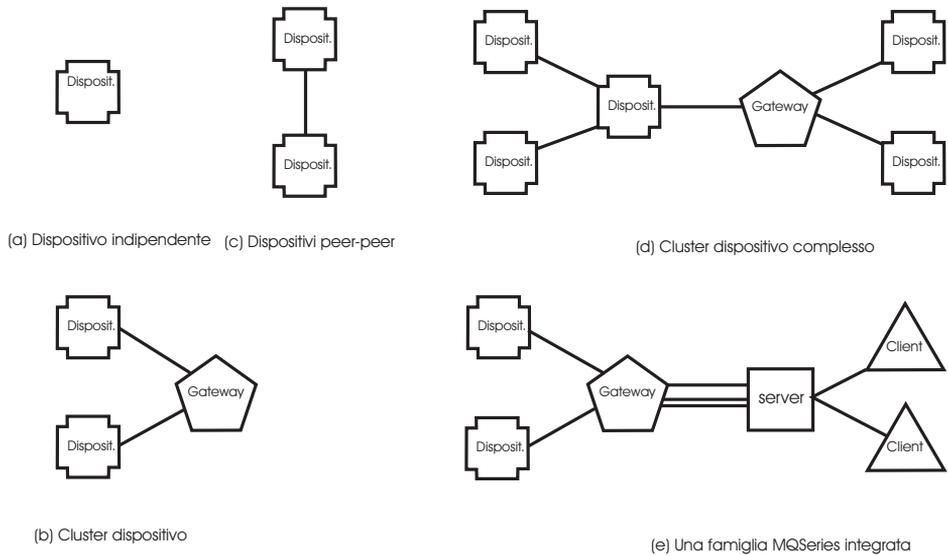


Figura 4. Configurazioni tipiche per dispositivi

In Figura 4 (b) un gateway viene utilizzato per collegare dei dispositivi. Una caratteristica dei gateway consiste nel fatto che sono in grado di gestire più richieste di collegamenti entranti in contemporanea, rispetto ai dispositivi che invece possono gestire una sola richiesta alla volta. Sia i gateway che i dispositivi possono inviare più richieste simultanee in uscita. Se nella configurazione (b) è stato utilizzato un dispositivo al posto di un gateway, i due dispositivi terminali avrebbero dovuto tentare di entrare in contatto con il dispositivo intermedio, anche se questo li avrebbe dovuto contattare simultaneamente. In (d) sono utilizzati al tempo stesso un dispositivo e un gateway per il collegamento di dispositivi. In (e) viene utilizzato un gateway per collegare una rete di dispositivi ad un server MQSeries, una configurazione in cui i

messaggi possono circolare tra tutte le parti costituenti, ovvero dispositivi, gateway, server, workstation e client. Le caratteristiche più importanti di questi componenti sono indicate in Tabella 2.

*Tabella 2. Elementi di MQSeries Everyplace e MQSeries*

<b>Componente</b>	<b>Caratteristiche</b>	<b>Offerto da</b>
Dispositivo (MQSeries Everyplace )	Fornisce messaggistica sicura per applicazioni attraverso canali dinamici <ul style="list-style-type: none"> <li>consente l'accesso sincrono a code locali e remote</li> <li>consente la distribuzione asincrona a code remote</li> <li>limitato alla gestione di una richiesta in entrata alla volta</li> </ul>	Flessibile
Client (MQS)	Fornisce messaggistica sicura per applicazioni locali <ul style="list-style-type: none"> <li>richiede una connessione del canale client sincrono ad un server</li> <li>consente l'accesso sincrono solo alle code sul server collegato</li> <li>consente la distribuzione asincrona a code remote tramite il server collegato</li> </ul>	Host distribuito
Gateway (MQSeries Everyplace )	Fornisce messaggistica sicura per applicazioni attraverso canali dinamici <ul style="list-style-type: none"> <li>consente l'accesso sincrono a code remote di &amp;</li> <li>consente la distribuzione asincrona a code remote</li> <li>può gestire più richieste in entrata alla volta</li> </ul> <p>Supporta il collegamento a più server MQSeries tramite canali client</p>	Flessibile
Server (MQS)	Fornisce messaggistica sicura per applicazioni attraverso canali di messaggi <ul style="list-style-type: none"> <li>consente l'accesso sincrono a code locali</li> <li>consente la distribuzione asincrona a code remote</li> </ul> <p>Supporta il collegamento a più client MQSeries tramite canali client</p>	Host distribuito
Workstation (MQS)	Fornisce messaggistica sicura per applicazioni attraverso canali di messaggi <ul style="list-style-type: none"> <li>consente l'accesso sincrono a code locali</li> <li>consente la distribuzione asincrona a code remote</li> </ul>	Workstation



---

## Capitolo 4. Requisiti

Questo capitolo descrive i requisiti che caratterizzano la progettazione e l'implementazione di MQSeries Everyplace.

---

### Caratteristiche

MQSeries Everyplace estende l'ambito della messaggistica alla famiglia MQSeries:

- Con il supporto di dispositivi low-end, come PDA, telefoni e sensori. In questo modo è possibile partecipare a una rete di messaggistica MQSeries. Supporta i dispositivi di media dimensione, come laptop, workstation e alcune piattaforme distribuite. MQSeries Everyplace offre la stessa qualità di servizi, con unica consegna sicura dei messaggi, e consente lo scambio di messaggi con altri membri della famiglia.
- Fornendo ampie funzioni di sicurezza per proteggere messaggi, code e dati correlati, in memoria o in trasmissione.
- Funzionando in modo efficiente in ambienti di comunicazione ostili in cui le reti sono instabili, o in cui la larghezza di banda è limitata. Dispone di un protocollo di rete efficiente e della correzione automatica di errori di comunicazione.
- Con il supporto dell'utente mobile, per fare in modo che i punti di connessione della rete vengano modificati in roaming dei dispositivi. Consente inoltre di controllare il funzionamento in condizioni in cui le risorse di batteria e le reti sono limitate o riportano errori.
- Operando tramite firewall correttamente configurati
- Con la riduzione delle attività di gestione per l'utente, per poter tenere nascosta la presenza di MQSeries Everyplace su un dispositivo. Ciò rende MQSeries Everyplace una base ideale per creare applicazioni per programmi di utilità.
- Con una facile procedura di personalizzazione e di ampliamento, attraverso l'impiego di regole fornite dalle applicazioni e altre classi che modificano il funzionamento, oppure tramite una suddivisione delle classi di oggetti base, ad esempio, per rappresentare vari tipi di messaggi.

---

### Applicazioni

Non esiste un elenco limitato delle applicazioni disponibili di MQSeries Everyplace in quanto la scelta è ampia e varia, ma una gran parte delle applicazioni può essere personalizzata per gruppi di utenti particolari. L'elenco seguente contiene alcuni esempi delle applicazioni considerate:

- **Applicazioni per il consumatore:** spesa in supermarket da casa utilizzando una PDA, raccolta delle opinioni dei viaggiatori sulle compagnie di voli, transazioni finanziarie da un telefono mobile
- **Applicazioni per il controllo:** raccolta e integrazione di dati da sensori di tubi dell'olio trasmessi via satellite, operazioni remote di attrezzature (per esempio valvole) con sicurezza per garantire la validità dell'operatore
- **Lavoratori in movimento:** agenti rappresentanti (ramo assicurazioni), stampa veloce di ricevute al cliente per società di recapiti, camerieri di fast-food in comunicazione

con la cucina, punteggio in tornei di golf, sistemi mobili protetti, sistemi di messaggistica per la polizia, informazioni per i lavoratori con difficoltà di comunicazione, lettura metrica di superfici.

- **Produttività personale:** replica posta/calendario, replica database, riduzione dimensioni laptop

---

## Requisiti del cliente

I requisiti che hanno influenzato la progettazione di MQSeries Everyplace comprendono:

- **Gestione:** gestione e impostazione minima; supporto per la gestione locale e remota; capacità di ampliare e personalizzare le funzioni amministrative per soddisfare le esigenze di applicazioni particolari; enfasi sulla scoperta e correzione automatica di problemi; elementi di amministrazione indipendenti da poter utilizzare in modo selettivo.
- **Comunicazioni:** un protocollo di rete molto efficiente; intestazioni minime; nessun campo obbligatorio nei messaggi (ad eccezione di un solo identificativo); capacità di modificare la codifica dei dati; supporto per compressione, codifica e autenticazione; negoziazione end-to-end delle funzioni di compressione e sicurezza; capacità di attraversare facilmente i firewall; adattatori di comunicazioni collegabili.
- **Compatibilità:** qualità MQSeries di servizi e interscambio di messaggi; capacità di comunicare con sistemi MQSeries esistenti senza modificare applicazioni; controllo flessibile dell'interscambio di messaggi tra MQSeries e MQSeries Everyplace.
- **Struttura:** al di sotto di 100 KB per un sistema di dispositivi con configurazione minima.
- **Funzioni:** funzioni di messaggistica sincrona e asincrona, accesso ai messaggi memorizzati su code locali o remote; capacità di utilizzare tutti i campi di un messaggio per il recupero selettivo; controllo della funzione di memoria di una coda.
- **Supporto delle regole:** controllo di vari aspetti del funzionamento generale attraverso regole, ad esempio quando inviare messaggi, quando ritentare una comunicazione, cosa fare di un messaggio troppo grande, cosa fare quando una coda di destinazione è piena.
- **Sicurezza:** supporto completo per sicurezza, autenticazione e accettazione; sicurezza per livelli di code e di messaggi; protezione del sistema di messaggistica da interferenze alla sicurezza; sicurezza di collegamento tramite algoritmi standard del settore; capacità di integrazione con le credenziali del sistema operativo dell'utente: capacità di aderire ai requisiti di sicurezza nazionale, in modo che il supporto per la sicurezza venga modificato quando i messaggi passano i confini del paese.

---

## Capitolo 5. Contenuti del prodotto

---

### Introduzione

Gli elementi fondamentali del modello di programmazione di MQSeries Everyplace sono messaggi, code e programmi di gestione code. I messaggi di MQSeries Everyplace sono oggetti contenenti dati definiti dalle applicazioni. Vengono memorizzati in una coda, per poi viaggiare su una rete MQSeries Everyplace. I messaggi sono indirizzati ad una coda di destinazione, specificando il relativo programma di gestione code e la coppia di nomi. Le applicazioni dispongono i messaggi sulle code con un'operazione di inserimento (put) e in genere li recuperano con un'operazione di estrazione (get). Le code possono essere locali o remote e sono gestite dagli appositi programmi di gestione code. Sia i dispositivi che i gateway memorizzano i dati di configurazione in un registro.

Le applicazioni su dispositivi possono utilizzare tutte le API o funzioni disponibili sul dispositivo, senza alcuna limitazione per le interfacce di programmazione di MQSeries Everyplace. Attraverso i canali dinamici è possibile collegare dispositivi MQSeries Everyplace ad altri dispositivi MQSeries Everyplace e/o ad un gateway MQSeries Everyplace.

Anche le applicazioni su gateway possono utilizzare tutte le API o funzioni disponibili sul gateway, non solo le interfacce di programmazione di MQSeries Everyplace. Attraverso i canali dinamici, è possibile collegare un gateway ad altri gateway e/o a dispositivi MQSeries Everyplace. Attraverso canali client di MQSeries, è possibile collegare un gateway a uno o più server MQSeries (ma non ad altri gateway MQSeries Everyplace). Sia MQSeries Everyplace che MQSeries possono coesistere su un solo computer nonostante la presenza di MQSeries sia del tutto facoltativa.

e caratteristiche dei dispositivi con funzioni complete e dei gateway sono le stesse, ad eccezione di:

- I gateway possono gestire varie richieste simultanee in entrata (da altri dispositivi e/o gateway)
- I gateway possono essere connessi contemporaneamente a più server MQSeries

I canali dinamici supportano le seguenti connessioni di rete:

- Connessioni dial-in
- Connessioni permanenti, ad esempio una LAN convenzionale, una linea temporanea, una LAN senza fili o a infrarossi.

I protocolli di comunicazione sono implementati da una serie di adattatori, uno per ciascun protocollo supportato. In questo modo è possibile aggiungere nuovi protocolli, quando richiesto, e la struttura della memoria di un determinato ambiente può essere progettata per una particolare configurazione.

Le code vengono mappate in modo individuale su supporti di memoria tramite un'altra serie di adattatori. A questo punto una coda può essere memorizzata in un file system o nella memoria, in base all'adattatore prescelto.

L'interfaccia di programmazione MQSeries Everyplace è progettata in modo che le applicazioni vengano scritte senza dover dipendere dalla posizione delle code. Un programma creato per accedere alle code locali dovrebbe quindi essere eseguito senza problemi da un programma di gestione code remote (in base all'adesione alle modalità di sicurezza e al fatto che alcune operazioni di MQSeries Everyplace non sono supportate sulle code remote). Questa indipendenza del programma riguarda anche l'impiego delle funzioni amministrative.

---

## Oggetti messaggi

Gli oggetti messaggi di MQSeries Everyplace differiscono molto dai messaggi supportati da MQSeries. In MQSeries, i messaggi sono dei gruppi di byte, divisi in un'intestazione e un corpo del messaggio. MQSeries individua l'intestazione, che comprende informazioni fondamentali come l'identità della risposta alla coda, la risposta al programma di gestione code, l'id di messaggio e l'id di correlazione; il corpo del messaggio invece non viene individuato.

Invece i messaggi in MQSeries Everyplace sono *oggetti messaggi* ereditati da un oggetto di MQSeries Everyplace definito *oggetto di campi*. I messaggi sono oggetti concreti, senza un'intestazione né un corpo del messaggio. La vera natura dell'oggetto messaggio diventa più chiara quando viene individuato l'oggetto campi di base. Questi oggetti campi, utilizzati in MQSeries Everyplace, sono un insieme di campi, in cui un campo è costituito da un nome, un tipo di dati e i dati stessi. I nomi di campi sono stringhe in caratteri ASCII (un numero di caratteri protetti) di lunghezza illimitata.

I campi possono essere di tipo:

- stringa **ASCII** o un gruppo dinamico di stringhe ASCII
- valore **Booleano**
- valore di **Byte**, gruppo fisso o un gruppo dinamico di valori di byte
- Valore a **doppia virgola mobile**, gruppo fisso o un gruppo dinamico di valori a doppia virgola mobile
- oggetto **Campi** o un gruppo dinamico di oggetti campi (nidificazione di oggetti campi supportata)
- Valore a **virgola mobile**, gruppo fisso o un gruppo dinamico di valori a virgola mobile
- **Cifra** (4 byte), gruppo fisso o gruppo dinamico di cifre
- **Cifra lunga** (8 byte), gruppo fisso o gruppo dinamico di cifre lunghe
- **Cifra breve** (2 byte), gruppo fisso o gruppo dinamico di cifre brevi
- stringa **UNICODE** o gruppo dinamico di stringhe UNICODE

Gli oggetti campo sono dotati di un *tipo* in cui il tipo corrisponde al nome classe dell'oggetto di programmazione. Gli oggetti dipendenti di questa classe di oggetti

vengono utilizzati dalle applicazioni come oggetti messaggi e da MQSeries Everyplace per avviare la classe di oggetto quando richiesto, ad esempio dopo l'invio di un messaggio attraverso un canale.

Gli oggetti campi forniscono un numero di metodi, per esempio è possibile numerare i campi o verificarne l'esistenza. E' inoltre possibile confrontare i campi in base alle uguaglianze. E' possibile eseguire il dump e il recupero di queste voci di campi da e verso un gruppo di byte, utilizzato ad esempio per fornire dati per la trasmissione dell'oggetto e per recuperare l'oggetto dopo la trasmissione. Questi metodi di dump e recupero possono essere sovrascritti per disporre in serie gli oggetti campi in modo diverso, per esempio, per interrogare un database sul contenuto dei campi al momento della trasmissione. Tabella 3 elenca le proprietà degli oggetti campi e/o dei campi che li costituiscono.

*Tabella 3. Oggetti campi e proprietà*

Proprietà	Presenza	
	Oggetti campi	Campi
Oggetto di attributo associato	Facoltativo	
Campi costitutivi	sì	
Nascosto		sì
Nome		sì
Tipo	sì	sì
Valore		sì

La proprietà nascosta consente di ignorare questo campo per operazioni di confronto.

Gli *oggetti attributo* sono dotati del meccanismo per eseguire autenticazione, codifica e compressione, e possono essere associati agli oggetti campi.

- **Autenticazione:** accesso ai controlli
- **Compressione:** riduce i requisiti di memoria (per operazioni di trasmissione e/o memorizzazione)
- **Codifica:** protegge i dati quando si esegue il dump dell'oggetto (e ne consente il recupero)

Gli oggetti attributo sono fondamentali per il modello di sicurezza di MQSeries Everyplace e consentono una serie di funzioni, tra cui l'accesso selettivo al contenuto e la protezione dei dati per la memorizzazione. Tabella 4 elenca le proprietà degli oggetti attributo. Il valore delle regole, laddove presente, controlla le operazioni consentite.

*Tabella 4. Proprietà degli oggetti attributi*

Proprietà	Presenza
Programma di codifica	facoltativo (solo in determinate circostanze)
Strumento di autenticazione	facoltativo
Compressore	facoltativo

*Tabella 4. Proprietà degli oggetti attributi (Continua)*

Proprietà	Presenza
Regola	facoltativo
Tipo	facoltativo

Gli oggetti messaggio provengono dagli oggetti campi e comprendono un UID (unique identifier) creato da MQSeries Everyplace. Questo UID identifica in modo unico un oggetto messaggio ed è costituito da:

- **Nome** del programma di gestione code di origine (aggiunto dal programma di gestione code alla ricevuta dell'oggetto). Deve trattarsi di un nome unico.
- **Ora** della creazione dell'oggetto messaggio (aggiunto alla creazione)

Gli oggetti messaggio hanno delle proprietà di base elencate in Tabella 5 oltre a quelle che ereditano come oggetti campi.

*Tabella 5. Proprietà degli oggetti messaggi*

Proprietà	Spiegazione
Msg_OriginQMgr	Il nome del programma di gestione code che ha inviato il messaggio
Msg_Time	L'ora in cui l'applicazione ha creato l'oggetto messaggio

Queste due proprietà costituiscono l'UID (unique identifier) dell'oggetto messaggio

Non è richiesta alcuna informazione in un messaggio destinato a un altro programma di gestione code di MQSeries Everyplace, anche se in genere vengono inseriti altri campi per le informazioni del messaggio. Questi messaggi derivano dalla classe oggetto di base dei messaggi e contengono dei campi aggiuntivi. Di questi ultimi, un certo numero sarà comune ad una vasta gamma di applicazioni, per esempio quello per la risposta al programma di gestione code. MQSeries Everyplace fornisce inoltre un relativo supporto ai campi.

Tabella 6 elenca i campi supportati.

*Tabella 6. Campi di oggetti messaggi forniti*

Nome campo	Utilizzo
Msg_CorrelID	Stringa di byte utilizzata per correlare una risposta al messaggio originale
Msg_ExpireTime	Ora in cui il messaggio può essere cancellato (anche se non recapitato)
Msg_LockID	La chiave necessaria per sbloccare un messaggio
Msg_MsgID	Utilizzato dall'applicazione per la correlazione al messaggio originale
Msg_Priority	Priorità del messaggio
Msg_ReplyToQ	Nome della coda a cui inviare la risposta al messaggio

Tabella 6. Campi di oggetti messaggi forniti (Continua)

Nome campo	Utilizzo
Msg_ReplyToQMgr	Nome del programma di gestione code a cui inviare la risposta a un messaggio
Msg_Resend	Indica che si tratta del rinvio di un messaggio precedente
Msg_Style	Distingue i comandi da richieste, risposte e via di seguito

In tutti i casi è disponibile una costante che consente di portare il nome campo a un solo byte. Per alcuni campi si adotta un provvedimento più esteso - per esempio: la priorità (se presente) influisce sull'ordine in cui vengono trasmessi i messaggi; l'id di correlazione provoca l'indicizzazione di una coda per consentire un recupero veloce dei valori dei campi; il tempo di scadenza avvia la scadenza del messaggio e così via.

Gli oggetti messaggi sono costituiti da un numero di metodi definiti, per esempio la capacità di estrarre l'UID del messaggio, il nome originario del programma di gestione code e l'ora di creazione dell'oggetto. Altri metodi utili vengono ereditati dalla classe degli oggetti campi, per esempio vari metodi per inserire ed estrarre valori campi. Un metodo di particolare interesse è il *dump*, utilizzato per eseguire il dump dei dati su una stringa di byte. MQSeries Everyplace richiama questo metodo quando un messaggio viene salvato nella memoria permanente e quando deve essere trasmesso su un canale dinamico. In questo modo l'oggetto messaggio è responsabile nel determinare la rappresentazione esterna del valore dei dati, che può essere eseguita in vari modi. Per esempio, un oggetto può semplicemente eseguire il dump dei valori dei campi oppure interrogare il database. Il metodo complementare di *recupero* offre delle funzioni di controllo simili quando un oggetto viene creato nuovamente dal formato generato dal dump. Se l'oggetto messaggio ha un oggetto attributo collegato, eseguendo il dump viene richiamato il programma di codifica dati dell'attributo e il compressore; con il recupero vengono invece richiamati lo strumento di decodifica e il decompressore.

Quando MQSeries Everyplace trasmette un oggetto messaggio, per ridurre la struttura sulla rete, non trasmette il file di classe associato. Allo stesso modo, la classe di messaggio appropriata deve essere disponibile per tutti i programmi di gestione code in cui l'oggetto messaggio viene avviato.

Il metodo dump per l'oggetto di messaggio predefinito è stato ottimizzato per ridurre la dimensione della stringa di byte generata ed avere così una memoria messaggi e una trasmissione efficienti.

## Formato dei dati dump

Il formato predefinito dei dati dump codifica i campi nel modo seguente:

```
{Lunghezza Identificativo Margine {Dati}} {Lunghezza
Identificativo Margine {Dati}} { ...}
```

dove:

- *Dati*: il valore dei dati. Le cifre sono compresse, con 0 e F rimossi. Gli indicatori booleani non hanno byte di dati associati
- *Margine*: un byte speciale che delimita il confine tra l'identificativo e la voce facoltativa Dati. Il byte indica anche il tipo di voce Dati

- *Identificativo*: contiene il nome campo in una stringa di byte ASCII di lunghezza variabile, che termina con un byte finale
- *Lunghezza*: indica la lunghezza del campo di dati. Si utilizza un numero variabile di byte tra 1 e 4. Il primo byte riserva i primi due bit per indicare la lunghezza del campo per la lunghezza. Sono supportate le lunghezze nell'intervallo tra 0 e 1.073.741.823

Ne deriva un flusso di dati altamente ridotto. E' possibile eseguire altri salvataggi comprimendo i dati. La compressione XOR con un flusso di byte precedente può creare buoni risultati ma, data la natura variabile di tali campi e dal momento che l'ordine dei campi può variare, una semplice compressione XOR non sempre può dare l'effetto desiderato. MQSeries Everyplace comprende un XOR intelligente, che funziona campo per campo e che può migliorare la compressione.

---

## Code

Le code vengono utilizzate in genere per contenere gli oggetti messaggi in attesa di rimozione dai programmi di applicazioni. Come i messaggi, anche le code derivano dagli oggetti campi. Non è consentito l'accesso diretto delle applicazioni nell'oggetto code<sup>3</sup>; invece il programma di gestione code funziona da intermediario tra i programmi di applicazione e le code. Le code si distinguono in base al nome, che può essere una stringa di caratteri ASCII di lunghezza illimitata<sup>4</sup> ma deve essere unica all'interno di un determinato programma di gestione code.

MQSeries Everyplace supporta un numero di tipi diversi di code:

### Code locali

Le code locali sono utilizzate dalle applicazioni per memorizzare i messaggi in modo protetto. Dispongono di una memoria messaggi alla quale si accede tramite una classe di adattatori, in genere la classe di adattatori di dischi. Una classe di adattatori di memoria viene in genere fornita con MQSeries Everyplace, che conserva in memoria i messaggi per un accesso più rapido (con il rischio di perdite di messaggi se il sistema viene danneggiato). Creando l'adattatore corretto è possibile memorizzare i messaggi in qualsiasi posto, su una base "da coda a coda", per esempio in un database relazionale, su un CD riscrivibile e così via. Le code locali possono essere poi utilizzate in linea e non, ovvero collegate o meno a una rete. L'accesso e la sicurezza sono gestiti dalla coda e utilizzati da un programma di gestione code remoto, quando si è collegati a una rete: in questo modo altri utenti possono inviare o ricevere messaggi da e sulla coda. L'accesso alla coda locale è sempre di tipo sincrono.

### Code remote

Le code remote non si trovano nell'ambiente in locale; esiste in locale una definizione che identifica il programma corrente di gestione code e la coda

---

3. L'accesso diretto è possibile all'interno di una regola di coda

4. Per l'interoperabilità si consiglia di rispettare i limiti di denominazione di MQSeries, compreso un nome di lunghezza massima di 48 caratteri. La lunghezza può essere limitata dal file system utilizzato

reale. E' possibile accedere alle code remote in modo sincrono e asincrono. Se esiste una definizione della coda remota memorizzata in locale, la modalità di accesso si baserà su questa definizione. Altrimenti, si verifica la scoperta della coda, per cui vengono scoperte le caratteristiche e il modo di accesso diventa necessariamente sincrono.

Le code *sincrone* sono code a cui è possibile accedere solo quando si è collegati a una rete dotata di un percorso per il programma di gestione code in questione. Se non viene stabilito il collegamento alla rete, operazioni come inserisci, ricevi e sfoglia (vedere Tabella 10 a pagina 25) provocano un'eccezione. La coda corrente controlla i permessi di accesso e i requisiti di sicurezza necessari per l'accesso alla coda. E' responsabilità dell'applicazione gestire eventuali errori o nuovi tentativi per l'invio o la ricezione di messaggi, e in questo caso MQSeries Everyplace non sarà più responsabile per la consegna unica assicurata.

Le code *asincrone* possono contenere messaggi ma non possono recuperarli. Se viene stabilito il collegamento alla rete i messaggi vengono inviati alla coda e al programma di gestione code corrente. Se comunque la rete non è collegata, i messaggi vengono memorizzati in locale fino al momento del collegamento a una rete, quando cioè i messaggi verranno trasmessi. In questo modo le applicazioni possono operare sulla coda quando il dispositivo non è collegato; non è necessario che questi tipi di coda abbiano una memoria di messaggi per poter memorizzare temporaneamente dei messaggi.

### **Code di memorizzazione e inoltramento**

Questo tipo di coda memorizza i messaggi fino al momento in cui è possibile inoltrarli al programma di gestione code successivo (non necessariamente quello corrente). Questo tipo di coda viene definito in genere (ma non necessariamente) su un gateway e il dispositivo dovrebbe raccogliere i messaggi quando è collegato alla rete. Le code di memorizzazione e inoltramento possono contenere messaggi per molti programmi di destinazione (per la gestione) oppure può esistere una coda di memorizzazione e inoltramento per ciascun programma di destinazione. Quando un utente desidera inviare un messaggio a un destinatario che potrebbe non essere collegato, egli indirizzerà comunque il messaggio alla coda o al programma di gestione code del destinatario; il server intermediario rileverà che il destinatario non è collegato e memorizzerà il messaggio nella coda locale dei messaggi. Non occorre effettuare modifiche all'applicazione di invio per inviare un messaggio su una coda di destinazione collegata o non collegata.

### **Code del server home**

Questo tipo di coda si trova di solito su un dispositivo (considerato collegato in modo occasionale) e punta a una coda di memorizzazione e inoltramento di un programma di gestione code noto come server home. La coda del server home preleva e i messaggi dal server home ogni volta che il dispositivo si collega alla rete. Quando la coda preleva un messaggio dal server, la passa al programma di gestione code locali tramite i metodi `putMessage` e `confirmputMessage` (vedere "Operazioni dei programmi di gestione code" a pagina 24). A questo punto è responsabilità del programma di gestione di collocare il messaggio nella coda locale corretta. Questo metodo di estrazione dei messaggi dal server può essere più efficiente in termini di flusso sulla rete

rispetto al server che invia i messaggi: questo dipende dal fatto che la coda del server home utilizza l'avviso di ricezione del primo messaggio come richiesta per il messaggio successivo (se esistente), mentre il server per l'invio manda una richiesta/risposta per inviare il messaggio e la seconda richiesta/risposta per il flusso di conferma. Le code del server home che in genere hanno un intervallo di poll impostato controllano tutti i messaggi in sospenso sul server mentre si è collegati alla rete. L'intervallo di poll è un'opzione per la configurazione dell'amministrazione.

### **Code di amministrazione**

Questo tipo di coda riceve messaggi di amministrazione di MQSeries Everyplace. E' possibile utilizzare la coda di risposta dei messaggi di amministrazione anche per ricevere risposte ai messaggi di amministrazione inviati dal sistema MQSeries Everyplace. Le code di amministrazione non conoscono le operazioni di amministrazione, gestiscono i messaggi che contengono i dettagli sull'argomento.

### **code di collegamento di MQSeries**

Questa è una forma specializzata di coda remota con la definizione di un gateway e la coda di destinazione di un programma di gestione code di MQSeries. Questo tipo di coda fornisce un percorso tra gli ambienti di MQSeries Everyplace e MQSeries. I trasformatori vengono utilizzati per elaborare tutti i dati necessari o per la riformattazione dei messaggi. Un trasformatore di base viene fornito con MQSeries Everyplace; i programmatori devono poi personalizzare questo trasformatore per adeguarlo alle proprie necessità.

MQSeries Everyplace memorizza i dati sulle code in modo sicuro, assicurando che i messaggi vengano scritti fisicamente nei supporti e non semplicemente memorizzati nel buffer dal sistema operativo. Tuttavia MQSeries Everyplace non registra in modo indipendente le modifiche ai messaggi e alle code. Se è richiesto un recupero per un errore dei supporti, sarà necessario adottare delle soluzioni di hardware come l'utilizzo di sistemi di dischi RAID. Altrimenti la coda deve avere un corrispondente nella memoria recuperabile, per esempio alcuni sistemi secondari di database.

MQSeries Everyplace non richiede che il programma di gestione code definisca le code. Viene comunque preso un provvedimento per quattro code di sistemi, se richiesto:

- **AdminQ:** richiesto per la ricezione dei messaggi di amministrazione
- **AdminReplyQ:** utilizzato in modo facoltativo per le risposte ai messaggi di amministrazione
- **DeadLetterQ:** utilizzato per memorizzare i messaggi che non è possibile recapitare in altro modo
- **SYSTEM.DEFAULT.LOCAL.QUEUE:** una coda che condivide un nome uguale con la coda di sistema obbligatoria sui server di MQSeries

Le proprietà delle code sono illustrate in Tabella 7 a pagina 21, tuttavia non tutte le proprietà visualizzate sono valide per tutti i tipi di code:

Tabella 7. Proprietà delle code

Proprietà	Spiegazione
Admin_Class	Classe di code
Admin_Name	Nome coda Ascii
Queue_Active	Indica che la coda è attiva
Queue_AttRule	Classe di regole che controllano le operazioni di sicurezza
Queue_Authenticator	Classe autenticatore
Queue_BridgeName	Nome del collegamento MQ in uso
Queue_ClientConnection	Nome collegamento al client
Queue_CreationDate	La data in cui la coda è stata creata
Queue_Compressor	Classe compressore
Queue_Cryptor	Classe codificatore
Queue_CurrentSize	Numero di messaggi sulla coda
Queue_Description	Descrizione Unicode
Queue_Expiry	Scadenza dei messaggi
Queue_FileDesc	Il luogo in cui viene memorizzata la coda
Queue_MaxMsgSize	Lunghezza massima dei messaggi consentiti sulla coda
Queue_MaxQSize	Num. massimo di messaggi consentiti
Queue_Mode	Sincrono o asincrono
Queue_MQMgr	proxy del programma di gestione code di MQSeries
Queue_Priority	Priorità per i messaggi (se non sovrascritta da un valore di messaggio)
Queue_QAliasNameList	Nomi alternativi per la coda
Queue_QMgrName	Programma di gestione code che gestisce la coda reale
Queue_RemoteQName	Nome campo remoto di MQSeries
Queue_Rule	Classe di regole per le operazioni sulle code
Queue_TargetRegistry	Il tipo di registro di destinazione
Queue_Transporter	Classe trasportatore
Queue_Transformer	Classe trasformatore

Le funzioni amministrative consentono di creare ed eliminare le code oltre che ricercare o modificare le proprietà.

Le code non sono limitate solo all'utilizzo come destinazione per i messaggi. Le code di classe secondaria sono utilizzate negli scenari per le applicazioni di controllo processi, ad esempio l'oggetto coda può controllare direttamente un valore. Un messaggio di tipo corretto può far aprire la valvola, modificare il volume del flusso o via di seguito. Un'applicazione non sposterà i messaggi fuori dalla coda per eseguire l'operazione, l'oggetto coda stesso controlla l'operazione. Altre code possono, ad esempio,

aggiornare fogli elettronici o eseguire delle conversioni da testo in dialogo. I vantaggi di questa tecnica consistono nel fatto che gli aspetti della sicurezza delle code sono sempre operativi, così come viene garantita la messaggistica. In questo modo MQSeries Everyplace assicura la consegna unica dei messaggi, mentre un programma di autenticazione e uno di codifica associati garantiranno che solo il mittente autorizzato del messaggio potrà inviare tali messaggi, con contenuto altamente sicuro nel trasferimento. A nessuna applicazione è consentito l'accesso alla coda, né verrà richiesto.

---

## Programmi di gestione code

Il programma di gestione code di MQSeries Everyplace fornisce l'accesso delle applicazioni a messaggi e code, e controlla tutti i canali. In MQSeries Everyplace Versione 1.0 solo un programma di gestione code può essere attivo su ogni singola macchina virtuale Java in qualsiasi momento. Se esistono più JVM su un computer, può essere presente lo stesso numero di programmi di gestione code delle JVM. I programmi di gestione code vengono identificati per nome e questo deve essere globalmente unico<sup>5</sup> e una stringa di caratteri ASCII che può essere di lunghezza illimitata.<sup>6</sup> I programmi di gestione code possono essere configurati con o senza l'accodamento locale. Tutti i programmi di gestione code supportano operazioni di messaggistica sincrona; un programma di gestione code con accodamento locale supporta anche la consegna asincrona dei messaggi.

Le consegne di messaggi sincrona e asincrona presentano caratteristiche e conseguenze molto diverse:

### Consegna asincrona dei messaggi<sup>7</sup>

Con la consegna asincrona dei messaggi, l'applicazione passa il messaggio a MQSeries Everyplace per recapitarlo ad una coda remota. Al termine di questa operazione si torna immediatamente all'applicazione. MQSeries Everyplace conserva temporaneamente il messaggio in locale fino alla consegna. La consegna può essere a fasi, sotto responsabilità di MQSeries Everyplace. Questo tipo di operazione fornisce una *consegna unica assicurata*. Vedere "Consegna asincrona dei messaggi" a pagina 32 per ulteriori dettagli.

### Consegna sincrona dei messaggi:

La messaggistica sincrona può essere utilizzata per recapitare:

- Code di destinazione su un programma di gestione code di MQSeries Everyplace indirizzato sulla rete di MQSeries Everyplace
- Code di destinazione su un programma di gestione code di MQSeries collegato direttamente ad un gateway MQSeries Everyplace
- Code di destinazione su un server MQSeries non collegato in modo diretto

---

5. Questa limitazione non è applicata da MQSeries Everyplace o MQSeries, ma nomi di programmi di gestione duplicati possono far recapitare messaggi al programma di gestione sbagliato.

6. Per l'interoperabilità si raccomanda di osservare le regole del programma di gestione code di MQSeries, compresa la limitazione della lunghezza massima del nome a 48 caratteri. La lunghezza può essere limitata dal file system utilizzato.

7. MQSeries Everyplace non modi persistenti e non persistenti offerti da MQSeries, supporta solo il modo persistente. Se richiesto, tuttavia, la scelta della memorizzazione sulla coda consente uno scambio tra prestazioni e recupero.

Con la messaggistica sincrona, l'applicazione porta il messaggio su MQSeries Everyplace per la consegna. MQSeries Everyplace contatta in modo sincrono la coda remota di destinazione e posiziona il messaggio. Dopo la consegna MQSeries Everyplace torna all'applicazione.

Il contatto con il programma di gestione code remote può coinvolgere l'instradamento di MQSeries Everyplace attraverso dispositivi intermedi e/o gateway. Vedere "Consegna sincrona dei messaggi" a pagina 33 per ulteriori dettagli.

La consegna asincrona dei messaggi implica che l'applicazione locale consegna il messaggio a MQSeries Everyplace; da questo punto in poi la consegna dal programma di gestione code locale è responsabilità di MQSeries Everyplace. Quindi non è necessario che la rete e/o l'applicazione per la ricezione siano disponibili. L'ora dell'effettiva consegna è sconosciuta all'applicazione per l'invio. Per la consegna sincrona dei messaggi è necessario che la rete sia attiva, ma l'applicazione per l'invio saprà che il messaggio viene recapitato alla coda dell'applicazione per la ricezione. L'applicazione per la ricezione non deve necessariamente essere disponibile, sia nel caso della consegna asincrona che sincrona.

Un programma di gestione code locale è dotato di proprietà che riflettono la gestione locale delle code. E' necessaria inoltre una *definizione di collegamento* per ciascun programma di gestione code remoto con cui deve essere in contatto. Da questo momento in poi ci si riferirà alle definizioni di collegamento come a *definizioni di programmi di gestione code remote*. Queste definizioni possono includere tutte le informazioni necessarie per una comunicazione diretta tra i programmi di gestione code (compreso un indirizzo di rete) oppure possono indicare semplicemente che la comunicazione è indiretta, quindi tramite un programma di gestione code intermedio. Nell'ultimo caso è richiesto il nome del programma di gestione code successivo.

Le proprietà dei programmi di gestione code sono visualizzati in Tabella 8 e Tabella 9.

*Tabella 8. Proprietà dei programmi di gestione code locali*

<b>Proprietà</b>	<b>Spiegazione</b>
QMgr_ChnlAttrRules	Regole di attributi di canali
QMgr_ChnlTimeout	Time-out del canale
Admin_Class	Classe del programma di gestione code
QMgr_Description	Descrizione Unicode
Admin_Name	Nome del programma di gestione code
QMgr_Rules	Classe di regole per le operazioni dei programmi di gestione code

*Tabella 9. Proprietà di collegamento (programma di gestione code remote)*

<b>Proprietà</b>	<b>Spiegazione</b>
Con_Adapter	Il programma di descrizione file dell'adattatore
Con_AdapterOptions	Opzioni dell'adattatore (per esempio la cronologia di utilizzo)

Tabella 9. Proprietà di collegamento (programma di gestione code remote) (Continua)

Proprietà	Spiegazione
Con_AdapterParm	Dati ASCII da utilizzare con un adattatore (per esempio un nome di servlet)
Con_Aliases	Nomi alternativi per il programma di gestione code
Con_Channel	Il tipo di canale che il collegamento deve utilizzare
Con_Description	Descrizione Unicode
Queue_QMgrName	Proprietario della definizione
Admin_Name	Nome del programma di gestione code

In una definizione di collegamento sono supportati più adattatori.

## Operazioni dei programmi di gestione code

I programmi di gestione code supportano le operazioni di messaggistica e in modo facoltativo gestiscono le code. Le applicazioni hanno accesso ai messaggi tramite operazioni svolte da un programma di gestione code. Se non viene specificato un filtro, verrà recuperato il primo messaggio disponibile sulla coda. Un filtro è un oggetto campo di cui si trova una corrispondenza per uguaglianza e tutti i campi del messaggio possono essere utilizzati per un recupero selettivo. E' possibile associare all'operazione di ricezione (get), come a tutte le operazioni di invio e di recupero, un oggetto attributo da utilizzare per la codifica e la decodifica di un messaggio.

In MQSeries Everyplace, come per MQSeries, questa operazione di "get" è in genere distruttiva. Se è necessario un tipo di messaggistica sicura tra MQSeries Everyplace e l'applicazione, verrà utilizzata l'operazione di "get" seguita da una sequenza di conferma. Prima viene emesso un get id di conferma (con valore scelto dall'applicazione); questa operazione riceve il messaggio ma lo nasconde sulla coda invece di eliminarlo immediatamente. Una successiva conferma, che specifica l'UID di messaggio originale, indicherà che la ricezione per l'applicazione è avvenuta correttamente; a questo punto il messaggio verrà eliminato. Se la ricezione (get) non avviene correttamente, sarà possibile recuperare il messaggio. Le operazioni di invio (put) si svolgono in modo simile.

Specificando l'UID, è possibile *eliminare* i messaggi da una coda senza recuperarli.

Se è richiesta una lettura non distruttiva, è possibile *ricercare* le code per i messaggi (se si desidera, sotto il controllo di un filtro). Con la ricerca vengono recuperati tutti gli oggetti di messaggi che corrispondono al filtro, ma vengono lasciati sulla coda. E' supportata anche la *ricerca con blocco*. Esiste una funzione aggiuntiva per bloccare i messaggi corrispondenti sulla coda. I messaggi possono essere bloccati singolarmente, oppure in gruppi identificati da un filtro, e l'operazione di blocco darà come risultato un *id blocco*. I messaggi bloccati possono essere ricevuti o eliminati solo se si fornisce l'id blocco. Un'opzione di ricerca consente di recuperare l'intero messaggio o solo l'UID.

Le applicazioni possono *attendere* per un periodo di tempo specificato l'arrivo dei messaggi su una coda. Oppure è possibile utilizzare un filtro per identificare i messaggi

pertinenti e specificare un *id di conferma*. Le applicazioni possono infine attendere gli eventi messaggio di MQSeries Everyplace, in modo facoltativo con un filtro. L'utente in attesa verrà avvisato all'arrivo dei messaggi su una coda.

Le code sono abilitate per operazioni di messaggistica come indicato in Tabella 10

Tabella 10. Operazioni di messaggistica su code MQSeries Everyplace

	Coda locale	Coda remota <sup>1</sup>	
		Sincrono	Asincrono
Sfoggia (±blocco, ±filtro)	Sì	Sì	
Elimina	Sì	Sì	
Get (±filtro)	Sì	Sì	
Attendi (±filtro)	Sì		
Put	Sì	Sì	Sì
Attendi (±filtro)	Sì	Sì	

**Note:**

1. L'operazione di attesa remota sincrona è implementata con un poll della coda remota, quindi il tempo di attesa attuale è un multiplo del tempo di poll
2. <sup>1</sup>Il collegamento MQSeries Everyplace MQSeries fornito con MQSeries Everyplace Versione 1.0 supporta solo l'operazione 'put'.

I programmi di gestione code possono, in modo facoltativo, caricare applicazioni (classi) immediatamente dopo l'inizio; allo stesso modo possono terminare le applicazioni alla chiusura. I programmi di gestione code trasmettono degli eventi per riflettere uno stato o errore; come impostazione predefinita questi vengono visualizzati nella registrazione eventi.

---

## Amministrazione

L'amministrazione fornisce delle funzioni per configurare e gestire risorse di MQSeries Everyplace come code e collegamenti. Le funzioni relative ai messaggi sono considerate responsabilità delle applicazioni. L'amministrazione viene abilitata tramite un'interfaccia che gestisce la creazione e la ricezione di messaggi di amministrazione ed è progettata per gestire amministrazione locale e remota allo stesso modo. Le richieste vengono inviate alla coda di amministrazione del programma di gestione code prescelto e, se richiesto, è possibile ricevere risposte. Tutti i programmi di applicazione locali o remoti di MQSeries Everyplace possono creare ed eseguire messaggi di amministrazione direttamente o indirettamente tramite metodi di aiuto. I messaggi di amministrazione possono essere anche creati indirettamente con MQSeries Everyplace Explorer<sup>8</sup>, uno strumento di gestione che fornisce una GUI per l'amministrazione di sistema.

---

8. MQSeries Everyplace Explorer non è compreso nella Versione 1.0 ma sarà disponibile dal sito di download del software MQSeries su World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

La coda di amministrazione non capisce in che modo svolgere l'amministrazione di risorse individuali: questo metodo viene incapsulato in ciascuna risorsa e nel messaggio di amministrazione corrispondente.

## Messaggi di amministrazione

I messaggi di amministrazione ampliano l'oggetto messaggio base di MQSeries Everyplace. Tabella 11 elenca le classi di messaggio fornite per l'amministrazione di risorse MQSeries Everyplace. Questi messaggi di amministrazione di base possono diventare classi secondarie ed essere utilizzati per l'amministrazione di altri oggetti; ad esempio un tipo diverso di coda può essere gestito utilizzando una classe secondaria di MQQueueAdminMsg. Il collegamento di MQSeries Everyplace a MQSeries utilizza classi secondarie di MQAdminMsg in questo modo.

*Tabella 11. Classi di messaggi di amministrazione*

<b>Classe di messaggi di amministrazione</b>	<b>Utilizzo</b>
MQAdminMsg	Classe astratta utilizzata come base per tutti i messaggi di amministrazione
MQQueueManagerAdminMsg	Amministrazione dei programmi di gestione code
MQQueueAdminMsg	Amministrazione delle code locali
MQRemoteQueueAdminMsg	Amministrazione delle code remote
MQAdminQueueAdminMsg	Amministrazione della coda di amministrazione
MQHomeServerQueueAdminMsg	Amministrazione delle code del server home
MQStoreAndForwardQueueAdminMsg	Amministrazione delle code di memorizzazione e inoltro
MQConnectionAdminMsg	Amministrazione dei collegamenti tra programmi di gestione code
MQClientConnectionAdminMsg	Amministrazione di un oggetto di collegamento al client, utilizzato per la connessione a MQS
MQListenerAdminMsg	Amministrazione di un oggetto di attesa della coda di trasmissione per il collegamento, utilizzato per raccogliere messaggi da MQS
MQBridgeAdminMsg	Amministrazione di un collegamento a MQS
MQMQBridgesAdminMsg	Amministrazione di elenco di collegamenti a MQS
MQMQMgrProxyAdminMsg	Amministrazione di una rappresentazione di collegamento a un programma di gestione code MQSeries
MQMQBridgeQueueAdminMsg	Amministrazione di una coda di collegamento a MQ

La struttura di un messaggio di amministrazione dipende dalla classe di appartenenza, quindi dalla natura della risorsa in gestione e dai dettagli dell'operazione da svolgere su questa risorsa. In genere comunque i messaggi di amministrazioni sono strutturati come indicato in Tabella 12 a pagina 27:

Tabella 12. Struttura generica di un messaggio di amministrazione

Campi livello 1	Campi livello 2 e successivi	Utilizzo
Admin_Action		Creare, eliminare, ricercare e così via
Admin_Errors		Oggetti campi principali
	Campi multipli	Informazioni dettagliate su base di errori
Admin_MaxAttempts		Numero massimo di volte in cui è possibile eseguire un'operazione di amministrazione
Admin_Parameters		Oggetti campi principali
	Risorsa	Nome della risorsa da gestire
	Campi multipli	Dati dettagliati di parametri specifici per classe di messaggio e operazione
Admin_Reason		Messaggio di testo che indica il motivo di un errore
Msg_ReplyToQ		Nome della coda a cui inviare la risposta al messaggio
Msg_ReplyToQMgr		Nome del programma di gestione code a cui inviare la risposta a un messaggio
Admin_RC		Codice numerico di errore che indica il risultato
Msg_Style		Comando o richiesta/risposta
Admin_TargetQMgr		Nome del programma di gestione code che si occupa della risorsa di destinazione

Sono supportati tre stili di messaggi di amministrazione, cioè comandi (datagrammi) che indicano un'operazione di amministrazione che non richiede risposte, richieste che richiedono una risposta, e infine le risposte stesse. La risposta viene creata da una copia del messaggio originale; è possibile poi il mittente aggiunga dei campi supplementari che il ricevente può utilizzare.

Oltre al supporto base per i messaggi di amministrazione, vengono fornite anche le classi degli strumenti di aiuto che comprendono la costruzione del messaggio e l'interpretazione della risposta per e operazioni di amministrazione più comuni. Queste classi possono fornire a richiesta dei dialoghi per l'utente, che rendono più facile la creazione di strumenti semplici di amministrazione.<sup>9</sup>

9. Queste classi non sono comprese nella Versione 1.0 ma saranno disponibili dal sito per lo scaricamento del software di MQSeries sul World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

## Amministrazione selettiva

E' possibile controllare l'accesso all'amministrazione attraverso lo strumento di autenticazione sulla coda di amministrazione. Lo strumento di autenticazione fornito considera le applicazioni locali per rappresentare lo stesso utente locale, quindi abilita o meno l'amministrazione per tutte. Le applicazioni di amministrazione remota sono controllate dal richiamo dello strumento di autenticazione sul canale prima di qualsiasi flusso di messaggi di amministrazione. Sarà possibile così distinguere i vari utenti remoti e abilitarli o meno in modo separato. In tutti i casi, per tutti gli utenti, l'amministrazione viene abilitata o meno in modo completo. Se è richiesto un modello più avanzato di controllo sull'amministrazione, per esempio alcuni utenti di amministrazione hanno accesso a determinate code mentre altri no, sarà richiesta una ulteriore programmazione. Uno strumento di autenticazione più sofisticato può tenere traccia delle autorizzazioni associate alle identità degli utenti, di conseguenza i messaggi di amministrazione possono essere elaborati sulla base di queste autorizzazioni (vedere la sezione sulla sicurezza). Le regole associate alle code vengono poi sfruttate per abilitare o meno le operazioni in modo simile (vedere "Regole" a pagina 40).

## Programma di controllo e operazioni relative

L'amministrazione va spesso oltre la creazione e la modifica di oggetti, per esempio con il controllo dello stato del sistema e con la gestione di situazioni di errore, come informare un operatore quando una coda è quasi piena o quali sono le operazioni da eseguire all'arrivo di un messaggio troppo grande per la relativa coda di destinazione. Questi aspetti sono gestiti in MQSeries Everyplace utilizzando delle regole, ovvero delle classi richiamate ogni volta che gli oggetti modificano in modo sostanziale il loro stato oppure quando si verificano determinate situazioni di errore. Con MQSeries Everyplace viene fornita una serie predefinita di classi di regole, ma in genere vengono sostituite con classi personalizzate (vedere "Regole" a pagina 40).

---

## Canali dinamici

MQSeries Everyplace comunica tra dispositivi e/o programmi di gestione code gateway attraverso collegamenti logici che vengono definiti canali dinamici. Questi supportano flussi bidirezionali e sono stabiliti dal programma di gestione code come richiesto. La messaggistica sincrona e asincrona utilizzano gli stessi canali e un protocollo unico per MQSeries Everyplace. Diversamente, MQSeries utilizza in genere dei canali client per il traffico di tipo sincrono e una coppia di canali di messaggi per la messaggistica asincrona bidirezionale. I *canali di messaggi cluster* di MQSeries hanno caratteristiche simili ai canali dinamici di MQSeries Everyplace, anche se esistono alcune differenze.

Un canale dinamico è un collegamento logico tra due programmi di gestione code, con lo scopo di inviare e ricevere dati. Possono esistere più canali contemporaneamente, anche tra gli stessi elementi, con le seguenti caratteristiche: autenticazione, codifica, compressione e protocollo di trasporto utilizzato. Queste caratteristiche sono collegabili, (è possibile utilizzare diverse versioni su diversi canali) per cui ciascun canale ha una propria qualità di attributi di servizi:

- **Strumento di autenticazione:** un oggetto *strumento di autenticazione* o nullo per eseguire l'autenticazione di un utente o un canale
- **Canale:** la classe che fornisce servizi di trasporto.

- **Compressore:** un oggetto *compressore* o nullo in grado di eseguire la compressione e decompressione dei dati
- **Programma di codifica:** un oggetto *programma di codifica* o nullo in grado di eseguire codifica e decodifica
- **Destinazione:** la destinazione per il canale, ad esempio SERVER.XYZ.COM

Lo strumento di autenticazione si utilizza in genere solo per l'impostazione del canale. I compressori e i programmi di codifica sono invece utilizzati per tutti i flussi.

Il tipo più semplice di programma di codifica è MQeXorCryptor, che codifica i dati da inviare eseguendo un OR esclusivo. Questo tipo di codifica non è sicuro ma rende i dati non visualizzabili. Invece, MQe3DESCryptor implementa DES triplo. Il tipo più semplice di compressore è MQeRleCompressor, che comprime i dati sostituendo i caratteri ripetuti con un conteggio. Vengono forniti anche altri strumenti di autenticazione, compressori e programmi di codifica, vedere Tabella 13 a pagina 34.

Per la scelta dei canali si utilizzano le specifiche dell'adattatore di protocollo per determinare i collegamenti e i protocolli da utilizzare per un determinato canale. Ad ogni nodo intermedio si cercano le definizioni di canale per eseguire l'indirizzo necessario per il collegamento successivo. Quando non ci sono più definizioni, il canale termina e tutti i messaggi in corso vengono spostati al programma di gestione code di quel punto determinato.

I canali non sono direttamente visualizzabili per applicazioni o amministratori e vengono stabiliti dal programma di gestione code come richiesto. I canali collegano tra loro i programmi di gestione code e le relative caratteristiche vengono ripetutamente negoziate da MQSeries Everyplace in base alle informazioni da inviare. I trasportatori sono dei componenti di MQSeries Everyplace che sfruttano i canali per fornire comunicazioni a livello di coda. Anche questi non sono visualizzabili a un programmatore di applicazioni o a un amministratore.

Quando si richiede una messaggistica garantita MQSeries Everyplace consegna i messaggi all'applicazione una, ed una sola, volta. Per fare ciò, assicura che un messaggio venga trasferito correttamente da un programma di gestione code all'altro e che venga confermato, prima di eliminare la copia al termine della trasmissione. Nel caso di un errore di comunicazione, se non c'è stata alcuna conferma, è possibile trasmettere nuovamente un messaggio (la consegna di una volta sola non implica la trasmissione una volta sola) ma non vengono consegnati duplicati.

---

## Adattatori

Gli *adattatori* si utilizzano per creare delle corrispondenze tra MQSeries Everyplace e le interfacce dei dispositivi. I canali sfruttano gli adattatori di protocollo per eseguire HTTP, TCP/IP nativo ed altri protocolli. Allo stesso modo, le code sfruttano gli adattatori di memoria dei campi come interfaccia per un sistema secondario di memoria, come la memoria o il file system. Adapters forniscono un meccanismo per MQSeries Everyplace per ampliare il supporto del dispositivo e per abilitare la conversione.

Un *descrittore di file* è una stringa utilizzata per identificare, caricare e attivare un adattatore.

---

## Gestione dei collegamenti dialup

Il supporto di rete dialup per i dispositivi è gestito dal sistema operativo del dispositivo. Quando MQSeries Everyplace su un dispositivo non collegato tenta di utilizzare la rete, per esempio per inviare un messaggio, se l'accodamento di rete non è attivo, il sistema operativo stesso avvia i RAS (remote access services). In genere viene visualizzato un pannello che propone il profilo per un collegamento dialup. Il sistema operativo esegue un controllo fino a quando viene stabilito il collegamento. Quindi, l'utente del dispositivo deve assicurare che siano disponibili i profili di collegamento dialup adatti al sistema operativo da utilizzare. Non è necessario alcun supporto esplicito per la rete dialup nelle implementazioni dei dispositivi di MQSeries Everyplace.

---

## Traccia

La traccia viene abilitata avviando un programma indipendente che esegue operazioni di traccia. Sono comprese in MQSeries Everyplace le chiamate per tenere traccia di informazioni, situazioni di errore e avvertimenti con varianti di sistemi e utenti. Le applicazioni possono inoltre richiamare direttamente la traccia e aggiungere nuovi messaggi o modificare i messaggi di traccia esistenti. Il programma di traccia di esempio fornito permette di visualizzare i messaggi selezionati, stampati e/o diretti alla registrazione eventi. E' possibile scrivere altri programmi di traccia con ulteriori funzioni oppure progettarli per formattare e consegnare il risultato in modo diverso.

La maggior parte delle eccezioni di MQSeries Everyplace passa alle applicazioni per la gestione, e il programma di gestione eccezioni può instradarle nella traccia.

---

## Registrazione eventi

MQSeries Everyplace fornisce meccanismi e interfacce di registrazione eventi da utilizzare per lo stato di registrazione, ad esempio per il programma di gestione code avviato. E' possibile avviare la registrazione e scriverla in un file, come da impostazione predefinita, ma potrebbe essere intercettata e diretta altrove. La registrazione eventi di MQSeries Everyplace non registra dati di messaggi e non può essere utilizzata per recuperare messaggi o code.

---

## Reti MQSeries Everyplace

Le reti di MQSeries Everyplace sono collegate a dispositivi e gateway. Possono estendersi su più reti fisiche e instradare messaggi tra di esse. In generale, forniscono accesso sincrono e asincrono alle code con un modello di programmazione indipendente dalla posizione della coda.

## Configurazioni e scalabilità

Una selezione di topologie base di rete MQSeries Everyplace è indicata in Figura 5 a pagina 31. A questo scopo si presume che ognuna sia configurata con funzioni di comunicazione sincrone e asincrone.

Il caso più semplice è quando un dispositivo indipendente supporta una comunicazione sincrona tra applicazioni attraverso code locali, come riportato sopra in (a). Il caso (b) risulta più interessante perché mostra una rete peer-to-peer. Questa richiede entrambi i dispositivi per utilizzare lo stesso protocollo di comunicazioni e almeno uno dei dispositivi da configurare con funzioni che consentano di rispondere ad altri tentativi di contatto. In questo primo caso la comunicazione è possibile solo quando entrambi sono disponibili sulla stessa rete. La messaggistica asincrona consente alle applicazioni di funzionare quando i dispositivi non sono collegati, mentre la messaggistica sincrona funziona solo quando i dispositivi sono realmente collegati.

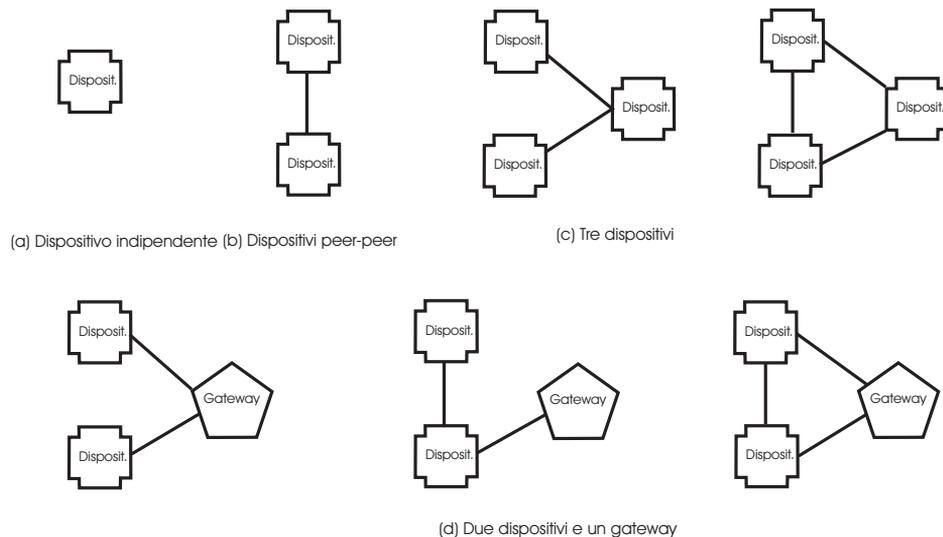


Figura 5. Reti MQSeries Everyplace semplici

Con tre dispositivi le possibilità sono di gran lunga maggiori. Una di esse può svolgere il ruolo di dispositivo di collegamento oppure possono essere tutte e tre collegate tra loro. Se sono configurate per scambiare messaggi tra di esse su una base peer-to-peer, saranno valide le considerazioni appena fatte. Se invece comunicano tramite un terzo dispositivo si avrà una maggiore flessibilità. I due dispositivi di comunicazione non devono necessariamente comunicare con lo stesso protocollo, ognuno deve utilizzare il protocollo condiviso dal dispositivo di collegamento. Inoltre i collegamenti asincroni consentono ai messaggi di viaggiare se il mittente e il destinatario non sono mai in rete contemporaneamente (purché siano in rete nel momento in cui il dispositivo di collegamento è disponibile). La comunicazione sincrona richiede che tutti e tre i dispositivi siano disponibili contemporaneamente. Per i dispositivi non collegati contemporaneamente, una configurazione che utilizza un terzo nodo è più adatta, perché l'intermediario sia prevalentemente disponibile.

I dispositivi nella pratica sono in genere collegati da un gateway di cui tre esempi sono riportati in Figura 5(d). La preferenza di un gateway come nodo di collegamento si basa sul fatto che i gateway supportano più richieste contemporanee di collegamenti in entrata. La prima configurazione indicata è la più probabile, anche se la seconda e la

terza sono fattibili, seppure poco utilizzate. Nella terza configurazione è possibile configurare solo un percorso per un determinato programma di gestione code remote quindi, sebbene ci siano due percorsi, è necessario sceglierne uno.

Per reti più ampie è possibile utilizzare un certo numero di gateway, ciascuno dei quali supporta un certo numero di dispositivi. E' possibile collegare tra loro i gateway in qualsiasi modo, ma se viene stabilito un collegamento completo, nessun percorso tra dispositivi coinvolge più di due gateway. Figura 6 indica un esempio di rete più ampia.

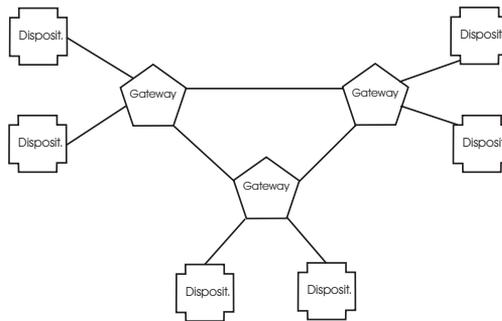


Figura 6. Una rete MQSeries Everyplace a stella

## Consegna asincrona dei messaggi

Quando un messaggio viene spostato su una coda remota in modo asincrono, l'oggetto messaggio viene spostato logicamente nella memoria di riserva associata alla definizione locale della coda, insieme al programma di gestione code di destinazione e ai nomi di code, e con le funzioni di compressore, strumento di autenticazione e programma di codifica che corrispondono alla destinazione prescelta per il messaggio. Il metodo dump dell'oggetto viene richiamato perché l'oggetto viene salvato nella memoria permanente in un formato protetto, come definito dalla coda di destinazione. Il programma di gestione code controlla la spedizione dei messaggi: identifica (o stabilisce) un canale con le funzioni adatte al programma di gestione code per il passaggio successivo, quindi crea (o riutilizza) un trasportatore per la coda di destinazione. Il trasportatore esegue il dump dell'oggetto e trasmette la stringa di byte di risultato. Il programma di gestione code di destinazione e il nome della coda non fanno parte di quel flusso di messaggio.

Se necessario, il messaggio viene codificato e compresso sul canale. Una volta raggiunto il programma di gestione code di destinazione, viene decodificato e decompresso. Viene creato quindi un nuovo oggetto messaggio, con il metodo di ripristino di quella classe di oggetto, e il risultato verrà posizionato nella coda di destinazione. Se il messaggio non arriva al programma di gestione code di destinazione, viene decodificato e aperto, quindi posizionato su una coda di recupero e inoltrato con caratteristiche adatte alla trasmissione. In entrambi i casi resta sulla coda in un formato protetto, come definito dalla coda di destinazione.

Una caratteristica della consegna sincrona dei messaggi sta nel fatto che i messaggi vengono spostati al programma di gestione con dei passaggi intermedi, essendo accodati per la trasmissione in avanti. I messaggi vengono poi spostati dalle code intermedie prima in ordine di priorità, quindi per sequenza data/ora.

## Consegna sincrona dei messaggi

La consegna sincrona dei messaggi è simile a quella di tipo asincrono descritta sopra, ma il coinvolgimento del programma di gestione code a passaggi intermedi ha luogo ad un livello più inferiore, coinvolgendo anche il trasportatore e i canali. Un canale viene stabilito end-to-end, utilizzando gli adattatori definiti nelle specifiche di protocollo ad ogni nodo intermedio, per identificare il collegamento successivo. Al termine dell'ultimo collegamento, dove non esistono altri descrittori file rilevanti, il messaggio passa ai livelli superiori per essere elaborato dal programma di gestione code. Il nodo di invio quindi non accoda il messaggio ma lo invia lungo il canale tramite passaggi intermedi, per poi passarlo al programma di gestione code prescelto per inviarlo alla coda di destinazione.

Il collegamento a MQSeries utilizza una coda di collegamento al gateway, che trasforma il messaggio in un formato MQSeries. Questo meccanismo implica che la messaggistica MQSeries Everyplace di tipo sincrono da un dispositivo è consentita per MQSeries, con il canale dinamico che termina con il gateway. Il messaggio viene recapitato in tempo reale dal gateway, tramite un canale client, ad un server MQSeries. Da qui potrebbe essere necessario instradarlo in modo asincrono su canali di messaggi MQSeries

Allo stesso modo, un dispositivo per sola messaggistica sincrona può inviare messaggi a una coda MQSeries Everyplace asincrona, sempre che sia disponibile un intermediario adatto.

---

## Sicurezza

MQSeries Everyplace fornisce una serie integrata di funzioni di sicurezza che consente di proteggere i dati dei messaggi sia in locale che nel trasferimento.

Le funzioni di sicurezza di MQSeries Everyplace forniscono protezione in tre categorie diverse:

- Sicurezza locale - protezione locale dei dati di messaggio (e altro)
- Sicurezza basata sulle code - protezione dei messaggi tra il programma di gestione code di avvio e la coda di destinazione
- Sicurezza a livello dei messaggi - protezione dei messaggi tra il mittente e il destinatario

La sicurezza locale e a livello di messaggi di MQSeries Everyplace viene utilizzata internamente in MQSeries Everyplace, ma è disponibile anche per altre applicazioni MQSeries Everyplace. La sicurezza MQSeries Everyplace basata sulle code è un servizio interno.

Le funzioni di sicurezza MQSeries Everyplace di tutte e tre le categorie proteggono i dati dei messaggi utilizzando un attributo (MQAttribute o attributo relativo). A seconda della categoria, l'attributo si applica in modo esplicito o implicito.

Ogni attributo può contenere i seguenti oggetti:

- Strumento di autenticazione
- Programma di codifica
- Compressore
- Chiave
- Nome entità di destinazione

Questi oggetti vengono utilizzati in modo diverso, a seconda della categoria di sicurezza MQSeries, ma in tutti i casi la protezione delle funzioni di sicurezza MQSeries Everyplace è attiva quando viene richiamato l'attributo collegato a un oggetto messaggio. Ciò avviene quando viene richiamato un metodo 'dump' di messaggio di MQSeries Everyplace (quando si utilizza il metodo 'encodeData' dell'attributo, per esempio per codificare e comprimere i dati di messaggi). Le funzioni di sicurezza di MQSeries Everyplace vengono disabilitate quando si richiama il metodo 'restore' dei messaggi di MQSeries Everyplace (quando si utilizza il metodo 'decodeData' dell'attributo, ad esempio per decomprimere e decodificare dati di messaggi).

Gli algoritmi supportati da MQSeries Everyplace Versione 1.0 per l'autenticazione, la codifica e la compressione sono descritti in dettaglio in Tabella 13.

*Tabella 13. Supporto per autenticazione, codifica e compressione*

Funzione	Algoritmo
Autenticazione	Basato su Mini Certificato, (deriva da WAP forum WTLS Mini Certificate)
	Convalida identità Windows NT/2000
Compressione	LZW
	RLE
Codifica	DES triplo
	DES
	MARS
	RC4
	RC6
	XOR

## Sicurezza locale MQSeries Everyplace

La sicurezza locale protegge i dati di messaggi di MQSeries Everyplace (o MQFields o derivanti da MQFields) in locale. Si ottiene creando un attributo con programma di codifica simmetrica e compressore adatti, creando e impostando una 'chiave' adatta (fornendo una password o passphrase) e collegando esplicitamente la chiave all'attributo, infine collegando l'attributo al messaggio di MQSeries Everyplace.

MQSeries Everyplace fornisce la classe MQeLocalSecure per l'impostazione della sicurezza locale, ma in tutti i casi è responsabilità dell'utente della sicurezza locale (interno a MQSeries Everyplace oppure applicazione esterna MQSeries Everyplace) impostare un attributo adatto e gestire la chiave per password o passphrase.

## **Sicurezza MQSeries Everyplace basata sulle code**

La sicurezza basata sulle code si applica ai messaggi sincroni e asincroni.

### **Sicurezza basata sulle code sincrone**

L'utilizzo della sicurezza basata sulle code sincrone consente a un'applicazione di lasciare tutte le considerazioni sulla sicurezza messaggi a MQSeries Everyplace. Le code dispongono di funzioni di autenticazione, codifica e compressione, che si utilizzano per determinare il livello di sicurezza necessario per proteggere i flussi di messaggi (anche per la memoria permanente).

Quando occorre inviare un messaggio, le funzioni di sicurezza della coda di destinazione vengono richiamate dal registro locale. Se non esistono, il programma di gestione code cerca di scoprire le caratteristiche del programma di destinazione e le richiama per un utilizzo successivo. Se esiste un canale per questo programma di gestione code, verrà utilizzato; altrimenti si creerà un nuovo canale. Vengono recuperati gli attributi delle code di destinazione.

In base alla qualità del servizio richiesto, gli attributi del canale per il programma di gestione code prescelto vengono modificati in modo dinamico. Il programma è soggetto a tutte le regole stabilite. In genere una regola consente un aggiornamento al livello di sicurezza, per esempio dal livello senza protezione a quello di bassa protezione, o da una protezione bassa ad una elevata. Se non è possibile aggiornare il canale, o il livello di sicurezza è giudicato eccessivo (per esempio non è richiesta alcuna protezione e il canale disponibile implementa una protezione forte) verrà creato un nuovo canale. Esiste una serie di canali, riutilizzati quando è possibile, con caratteristiche che si modificano dinamicamente, in base alle domande del traffico. I canali vengono distrutti automaticamente quando non sono richiesti. I messaggi vengono sempre posizionati sulle code al livello di sicurezza definito dalle caratteristiche della coda di destinazione.

L'autenticazione avviene a livello del canale, tenendo il livello minimo per messaggio. La sicurezza basata sulle code sincrone viene in genere utilizzata anche con i programmi di codifica simmetrica poiché consente una codifica/decodifica rapida. Nei casi simmetrici, MQSeries Everyplace utilizza all'inizio la codifica asimmetrica RSA per proteggere i flussi necessari per stabilire una chiave condivisa per il mittente e il destinatario. In seguito la codifica simmetrica viene utilizzata per proteggere la segretezza dei dati trasmessi. MQSeries Everyplace rende più impenetrabile la codifica di questi dati in quanto modifica in modo dinamico la chiave a ciascun flusso del canale. MQSeries Everyplace assicura inoltre l'integrità dei dati trasmessi creando e collegando un riassunto ai dati prima dell'invio e convalidandolo alla ricezione.

### **Sicurezza basata sulle code asincrone**

La messaggistica asincrona differisce da quella sincrona sopra descritta in quanto non fornisce una garanzia che la coda di destinazione sia accessibile nel momento in cui viene eseguita la funzione putMessage. In tal caso il programma di gestione code non

potrà inviare immediatamente il messaggio e lo posizionerà sulla coda di trasmissione; tuttavia verrà codificato in base alle caratteristiche della coda prescelta. Nel momento in cui è possibile trasmetterlo, viene decodificato e poi inviato ad un canale con le caratteristiche adatte. I messaggi quindi sono sempre protetti, anche in attesa della trasmissione. La messaggistica asincrona richiede la definizione della coda remota; in caso contrario, non è possibile determinare le caratteristiche della coda di destinazione.

Per il tipo asincrono, l'autenticazione non è possibile tra l'elemento di origine e di destinazione. Laddove l'autenticazione è importante, per esempio quando il destinatario deve determinare il mittente del messaggio (per stabilire se accettare o meno) o quando il mittente deve assicurare che il messaggio giunga solo al destinatario prescelto, è necessario utilizzare la sicurezza a livello di messaggio.

La sicurezza basata sulle code si utilizza insieme alla sicurezza a livello di messaggio, ma non è necessaria in quanto i dati di messaggio sono già protetti.

## Sicurezza a livello di messaggio

La sicurezza a livello di messaggio assicura la protezione dei dati di messaggio tra l'applicazione MQSeries Everyplace del mittente e quella del destinatario.

La sicurezza a livello di messaggio è un servizio per applicazioni che richiede che l'applicazione iniziale MQSeries Everyplace imposti un attributo a livello di messaggio e lo fornisca quando si utilizza la funzione `putMessage` per posizionare il messaggio su una coda di destinazione. L'applicazione per la ricezione deve impostare e poi passare un attributo corrispondente a livello di messaggio al programma di gestione code che lo riceve, in modo che l'attributo sia disponibile quando l'applicazione richiama `getMessage` per ricevere il messaggio dalla coda di destinazione.

Come la sicurezza locale, la sicurezza a livello di messaggio sfrutta l'applicazione dell'attributo in un oggetto messaggio. Il programma di gestione code dell'applicazione di invio gestisce la funzione `putMessage` con il metodo `'dump'`, che utilizza il metodo dell'attributo `'encodeData'` per proteggere i dati dei messaggi. Il programma di gestione code dell'applicazione per la ricezione gestisce la funzione dell'applicazione `getMessage` con il metodo `'restore'` che utilizza a sua volta il metodo dell'attributo `'decodeData'` per recuperare i dati del messaggio originale.

MQSeries Everyplace fornisce due attributi alternativi per la sicurezza a livello di messaggio:

### **MQeMAttribute**

Si utilizza per comunicazioni tra aziende in cui la fiducia reciproca viene gestita a livello delle applicazioni e non è richiesta una terza parte in qualità di garante della sicurezza. E' possibile utilizzare tutte le opzioni disponibili per il programma di codifica simmetrica e il compressore di MQSeries Everyplace. Come per la sicurezza locale, la chiave dell'attributo deve essere preimpostata prima di fornirla per le funzioni `putMessage` o `getMessage`. MQeAttribute fornisce un metodo semplice ed efficace per la protezione a livello di messaggi, consentendo l'utilizzo della codifica per proteggere la riservatezza dei messaggi, senza alcuna PKI (public key infrastructure) superiore.

### **MQeMTrustAttribute**

Questo attributo fornisce una soluzione molto più avanzata, che utilizza firme digitali e sfrutta l'infrastruttura della chiave pubblica predefinita. Utilizza infatti la firma digitale e la convalida ISO9796 per abilitare l'applicazione per la ricezione a provare che il messaggio proviene effettivamente dal mittente dichiarato. Il programma di codifica dell'attributo fornito si utilizza per proteggere la riservatezza del messaggio. Il riassunto SHA1 garantisce l'integrità dei messaggi e la codifica/decodifica RSA assicura che il messaggio può essere recuperato solo dal destinatario. Come per MQeAttribute, è possibile utilizzare tutte le opzioni disponibili per il programma di codifica simmetrica e il compressore di MQSeries Everyplace. Scelti per un'ottimizzazione delle dimensioni, i certificati utilizzati sono dei mini-certificati basati sul certificato WTLS proposto da WAP forum WTLS Specification. La struttura predefinita di MQSeries Everyplace fornisce la reciproca disponibilità delle informazioni necessarie per autenticare (convalidare firme) e codificare/decodificare.

Un messaggio tipico protetto da MQeMTrustAttribute avrà il formato:

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

dove:

<b>RSA-enc:</b>	RSA codificata con la chiave pubblica del destinatario dichiarato
<b>SymKey</b>	chiave simmetrica creata in modo casuale
<b>SymKey-enc</b>	codificata simmetricamente con SymKey
<b>Data</b>	dati del messaggio
<b>DataDigest</b>	riassunto dati del messaggio
<b>DigSignature</b>	firma digitale del mittente del messaggio

La sicurezza a livello di messaggio è indipendente dalla sicurezza a livello di coda.

## **Il registro**

Il registro è il primo magazzino di informazioni relative al programma di gestione code; ne esiste uno per ogni programma. Ogni programma utilizza il registro per memorizzare:

- Dati di configurazione dei programmi di gestione code
- Definizioni di code
- Definizioni di code remote
- Definizioni di programmi di gestione code remote
- Dati utente (comprese informazioni di sicurezza in base alla configurazione)

L'accesso al registro è in genere limitato all'utente legittimato del relativo programma e il PIN è protetto, ma con un'opzione configurabile è possibile dare l'accesso ad utenti specializzati in dimensioni piuttosto che in sicurezza.

## Entità autenticabili MQSeries Everyplace

La sicurezza basata sulle code, che utilizza una reciproca autenticazione con mini-certificati, e la protezione a livello di messaggi, che utilizza la firma digitale, hanno introdotto il concetto di 'entità autenticabile'. Nel caso dell'autenticazione reciproca, è normale pensare all'autenticazione tra due utenti (persone), ma in generale la messaggistica non si riferisce a un utente. Il concetto è invece gestito a livello delle applicazioni, quindi dall'utente di servizi di messaggistica. MQSeries Everyplace astrae deliberatamente il concetto di 'destinazione di autenticazione' dall'utente per 'entità autenticabile'. Non si esclude la possibilità che le entità autenticabili siano delle persone, tuttavia sarebbe una corrispondenza tra applicazioni. All'interno, MQSeries Everyplace definisce tutti i programmi di gestione code che possono creare o essere la destinazione di servizi dipendenti da mini-certificati come entità autenticabile. Inoltre MQSeries Everyplace definisce le code stabilite per utilizzare gli strumenti di autenticazione basati su mini-certificati come entità autenticabili. Quindi i programmi di gestione code che supportano questi servizi possono avere un'entità autenticabile, un programma di gestione code, oppure una serie di entità autenticabili, un programma di gestione code e tutte le code che utilizzano un programma di autenticazione basato su certificato.

## Registro privato e credenziali

Ogni entità autenticabile, per poter essere utilizzata, ha bisogno delle proprie credenziali. Ciò implica due obiettivi. Prima di tutto, come seguire la registrazione per ottenere le credenziali, e poi dove gestire le credenziali in modo sicuro. In genere questi obiettivi sono più difficili da attuare rispetto alle tecniche di codifica. MQSeries Everyplace fornisce dei servizi predefiniti, da utilizzare per consentire alle entità autenticabili di eseguire l'autoregistrazione, al registro privato (un elemento secondario del registro di base) di gestire in modo sicuro e credenziali private di un'entità autenticabile, e al registro pubblico (altro elemento secondario del registro di base) di gestire le credenziali pubbliche. Il registro privato fornisce al registro di base vari tipi di simboli sicuri o codificati, per esempio può essere un luogo sicuro per conservare oggetti pubblici come mini-certificati e oggetti privati come chiavi private. Fornisce inoltre un meccanismo che consente unicamente all'utente autorizzato di accedere agli oggetti privati e un supporto per servizi come la firma digitale e la codifica RSA, per cui gli oggetti privati non possano mai essere spostati dal registro privato. Grazie a un'interfaccia comune, nasconde il supporto del dispositivo, che è attualmente limitato al sistema di file locale, ma può essere esteso per delle corrispondenze future.

## Autoregistrazione

MQSeries Everyplace fornisce servizi predefiniti che supportano l'autoregistrazione. Questi servizi vengono avviati automaticamente quando si configura un'entità autenticabile, per esempio quando si avvia un programma di gestione code o si definisce una nuova coda. In entrambi i casi viene avviata la registrazione, vengono create nuove credenziali e memorizzate nel registro privato dell'entità autenticabile. Le fasi di autoregistrazione includono la creazione di una nuova coppia di chiavi RSA, la protezione e il salvataggio della chiave privata nel registro privato, e la sistemazione della chiave pubblica in una richiesta di 'nuovo certificato' al server predefinito dei mini-certificati. Supponendo che il server di mini-certificati sia configurato e disponibile, restituisce il nuovo mini-certificato dell'entità autenticabile insieme al proprio mini-certificato e questi, insieme alla chiave privata protetta, vengono memorizzati nel

registro privato dell'entità autenticabile come nuove credenziali. Mentre l'autoregistrazione fornisce un meccanismo semplice per stabilire le credenziali dell'entità autenticabile, per la protezione a livello di messaggio (MQeMTrustAttribute, vedere sopra) è necessario anche l'accesso alla chiave pubblica del destinatario (mini-certificato).

## **Registro pubblico e replica di certificati**

MQSeries Everyplace fornisce dei servizi predefiniti per la condivisione delle credenziali pubbliche di entità autenticabili (mini-certificati) tra i componenti di MQSeries. Questi costituiscono un requisito essenziale per la sicurezza a livello di messaggio basata su MQeMTrust. Il registro pubblico di MQSeries Everyplace fornisce un magazzino accessibile a tutti per i mini-certificati. Costituisce una funzione analoga a quella della rubrica telefonica personale su un telefono cellulare: la differenza sta nel fatto che, invece dei numeri di telefono, si tratta di una serie di mini-certificati delle entità autenticabili che vengono contattate più frequentemente. Il registro pubblico non è del tutto passivo per i servizi. Se si accede per fornire un mini-certificato che non è presente, e il registro è configurato con un componente del server home valido, il registro pubblico tenta automaticamente di recuperare il mini-certificato richiesto dal registro pubblico del server home. E' possibile utilizzare questi servizi per fornire un servizio di replica automatizzata intelligente di mini-certificati, che consente di avere a disposizione il mini-certificato giusto al tempo giusto.

## **Utilizzo applicazioni dei servizi di registro**

Il programma di gestione code di MQSeries Everyplace è progettato per utilizzare al meglio i servizi di registro privato e pubblico, e l'accesso a questi servizi non è limitato. Le soluzioni MQSeries Everyplace potrebbero definire e gestire le proprie entità autenticabili, per esempio gli utenti. E' possibile inoltre utilizzare i servizi di registro privato per autoregistrare e gestire le credenziali delle nuove entità autenticabili, e i servizi di registro pubblico per rendere disponibili in qualsiasi momento le credenziali pubbliche. Tutte le entità autenticabili registrate possono essere utilizzate come servizi a livello di messaggio del mittente o del destinatario, protetti da MQeMTrustAttribute

## **Servizio di creazione mini-certificati predefiniti**

MQSeries Everyplace fornisce un servizio di creazione mini-certificati predefiniti da configurare per poter soddisfare richieste di autoregistrazione registro privato. Con gli strumenti forniti da MQSeries Everyplace, una soluzione è in grado di impostare e gestire un servizio per creare mini-certificati per una serie ben controllata di nomi di entità. Le caratteristiche di questo servizio sono:

- Gestione dell'insieme di entità autenticabili registrate
- Creazione di mini-certificati (basati su mini-certificati WAP WTLS)
- Gestione magazzino dei mini-certificati

Gli strumenti forniti con MQSeries Everyplace permettono all'amministratore dei servizi di creazione mini-certificati di autorizzare di creare un mini-certificato per una determinata entità, registrando il nome dell'entità e l'indirizzo, e definendo un PIN di richiesta certificati valido per una sola operazione. L'operazione viene eseguita dopo che il controllo 'offline' abbia convalidato l'autenticità del richiedente. Il PIN di richiesta certificati viene inviato all'utente definito (in modo simile all'invio del PIN di una nuova

carta di credito). E' possibile quindi configurare l'utente del registro privato (per esempio il programma di gestione applicazioni MQSeries Everyplace o code MQSeries Everyplace) per fornire questo PIN di richiesta certificati al momento dell'avvio. Quando il registro privato avvia l'autoregistrazione, il servizio di creazione mini-certificati convalida le nuove richieste di certificati (basate sulla corrispondenza tra il nome dell'entità fornito e il PIN di richiesta certificati con i corrispondenti valori preregistrati), crea il nuovo mini-certificato e reimposta il PIN registrato in modo che non possa essere più utilizzato. Tutte le richieste di auto-registrazione di nuovi mini-certificati vengono eseguite su un canale sicuro.

La serie di mini-certificati emessi da un apposito servizio di creazione viene conservata nel relativo registro. Quando un mini-certificato viene riemesso (ad esempio dopo una scadenza), il mini-certificato scaduto viene archiviato.

## L'interfaccia di sicurezza

Viene fornita un'interfaccia facoltativa da implementare in un programma di gestione sicurezza personalizzata. I metodi consentono al programma di gestione sicurezza di autorizzare o respingere le richieste associate a:

- Aggiunta o rimozione di alias di classi
- Definizione di adattatori
- Corrispondenza di descrittori file
- Esecuzione di comandi di canali

---

## Configurazione e personalizzazione

### Regole

Le regole sono classi Java utilizzate per personalizzare l'utilizzo di MQSeries Everyplace in caso di modifiche di stato. Vengono fornite delle regole predefinite laddove necessario, ma possono essere sostituite da regole di applicazioni, o specifiche per installazione, per soddisfare i requisiti del cliente. I tipi di regole supportati differiscono nel modo in cui vengono avviati, ma non nelle prestazioni; le regole contengono una logica per cui sono in grado di svolgere un'ampia gamma di funzioni.

#### Regole di attributo

Questa classe di regole viene controllata ogni volta che si tenta di apportare una modifica di stato, per esempio a:

- Strumento di autenticazione
- Compressore
- Programma di codifica

La regola in genere abilita o meno la modifica.

#### Regole di collegamento MQSeries

Queste classi di regole vengono controllate quando MQSeries Everyplace per il codice di collegamento MQSeries modifica lo stato. Esiste una classe separata per le regole di collegamento per determinare quanto segue:

- Cosa fare di un messaggio quando non è possibile spostarlo in MQSeries Everyplace, se proviene da MQSeries. Ad esempio perché il messaggio è troppo grande o la coda non esiste.
- Gli oggetti per il collegamento dello stato dovrebbero essere avviati quando il server viene istanziato
- Cosa fare quando il collegamento rileva un problema con Sync Q su MQSeries (la memoria permanente utilizzata per il recupero errori). La regola predefinita visualizza solo il problema.
- Come convertire un messaggio MQSeries Everyplace in un messaggio MQ e viceversa. I trasformatori per eseguire la conversione di messaggi tra MQSeries Everyplace e MQ non derivano da classi MQeRule, ma devono implementare l'interfaccia MQeTransformerInterface. Inoltre, i trasformatori funzionano come le regole e vengono richiamati quando un messaggio richiede la conversione del formato.

### **Regole compositore RAS**

Questa classe di regole viene controllata ogni volta che un compositore RAS deve apportare una modifica di stato, per esempio:

- Cosa fare quando il numero da chiamare non è raggiungibile
- Cosa fare in caso di errore
- Nel tentativo di composizione, è possibile utilizzare sono alcuni tipi di collegamenti, a seconda dell'orario. Per esempio, utilizzare il telefono al di fuori degli orari di punta

### **Regole di coda**

Questa classe di regole viene controllata ogni volta che si verifica una modifica di stato della coda associata, per esempio:

- Aggiunta di un messaggio su una coda. Per esempio, per visualizzare se la soglia di errore viene superata (numero di messaggi, dimensione del messaggio, priorità non valida)
- Caratteristiche della coda assegnata o modificata
- La coda viene aperta o chiusa
- La coda deve essere eliminata

### **Regole del programma di gestione code**

Questa classe di regole viene controllata ogni volta che si verifica una modifica di stato del programma di gestione code, per esempio:

- Il programma di gestione code viene aperto. Per esempio, avviare un thread del timer per abilitare operazioni a tempo
- Il programma di gestione code viene chiuso. Per esempio, chiudere il thread del timer
- Viene aggiunta una nuova coda

---

## Stili di collegamento

MQSeries Everyplace può supportare client-server<sup>10</sup> e operazioni peer-to-peer. Un *client* è in grado di avviare a comunicazione con un server; un *server* è solo in grado di rispondere alle richieste da parte di un client. In un'operazione *peer-to-peer*, i due pari (peer) possono avviare flussi in entrambe le direzioni. Questi stili di collegamento hanno bisogno di diversi componenti di MQSeries Everyplace attivi. I componenti coinvolti sono:

- **Listener di canali:** in attesa di richieste di collegamento in entrata.
- **Programma di gestione canali:** che supporta più cavi di comunicazione logica in contemporanea tra le estremità.
- **Programma di gestione code:** che supporta le applicazioni grazie a funzioni di messaggistica e di accodamento.

Tabella 14 mostra la relazione tra questi componenti e lo stile di collegamento. Lo stile di collegamento client/server descrive la situazione in cui MQSeries Everyplace può funzionare in modo client o server. L'opzione servlet descrive il caso in cui MQSeries Everyplace è configurato come servlet HTTP con il server stesso responsabile per le richieste di collegamento in entrata.

*Tabella 14. Stili di collegamento*

	Programma di gestione code	Programma di gestione canali	Listener di canali
Client	Sì		
Client/server	Sì	Sì	Sì
Pari	Sì		
Server	Sì	Sì	Sì
Servlet	Sì	Sì	

***L'utilizzo di un programma di gestione canali di MQSeries Everyplace o di un listener di canali MQSeries Everyplace determina, a scopo di licenza, che un'istanza di MQSeries Everyplace è u gateway.***

Le applicazioni MQSeries Everyplace non sono direttamente responsabili dello stile di collegamento adottato dai programmi di gestione code. Lo stile è però importante perché influenza le risorse disponibili per le parti, che i programmi di gestione code possono collegare ad altri programmi, la struttura di MQSeries Everyplace per dispositivo o gateway, e i collegamenti che possono esistere contemporaneamente.

## Collegamento peer-to-peer

Un canale peer-to-peer comprende le funzioni di un programma di gestione canali e un listener per un singolo canale. Quando si crea un canale peer-to-peer tra due programmi di gestione code, uno di questi deve funzionare come listener e l'altro come punto di avvio del collegamento. Un programma di gestione code collegato peer-to-peer

---

10. In questa sezione i termini 'client' e 'server' riguardano l'utilizzo generale e non la semantica di MQSeries.

può avviare più collegamenti peer-to-peer con altri programmi di gestione code, ma può rispondere solo a una richiesta di collegamento in entrata e deve attendere la chiusura del canale peer-to-peer prima di poter rispondere a un'altra richiesta simile. Oltre un canale peer-to-peer, i due programmi di gestione code attivi possono avviare delle operazioni, quindi per esempio le applicazioni di ciascun programma possono avere accesso alle code di un altro programma.

I canali peer-to-peer non sono utilizzabili con i firewall perché la destinazione della richiesta di collegamento in entrata non viene accettata.

## Collegamento client-server

I canali standard, utilizzati per lo stile di collegamento client-server, non hanno funzioni di attesa ma dipendono da un listener indipendente al server, e il server richiede un programma di gestione canali per gestire più canali contemporanei. Il client avvia la richiesta di collegamento e il server risponde. Un canale server può in genere gestire più richieste in entrata dai client. Oltre un canale standard il client ha accesso alle risorse sul server. Se un'applicazione sul server ha bisogno dell'accesso sincrono alle risorse sul client, è necessario un secondo canale quando le regole vengono invertite. Tuttavia, essendo i canali standard di tipo bidirezionale, i messaggi destinati a un client dalla coda di trasmissione del server, vengono trasmessi sul canale standard (client-server) avviato.

Un client deve essere abilitato per più server contemporanei. Non è richiesto un programma di gestione canali per supportare questa configurazione poiché questi programmi gestiscono più canali in entrata.

Lo stile di collegamento client-server è in genere adatto per l'utilizzo con firewall, in quanto la destinazione del collegamento in entrata viene normalmente identificata come accettabile per il firewall.

## Stili di collegamento multipli

Un programma di gestione code singolo può avviare collegamenti peer-to-peer o client/server, e può rispondere a un server o a un pari (peer). In questo caso, il listener di canali di pari e il listener di canali standard devono avere numeri di porta diversi.

---

## Classi

MQSeries Everyplace fornisce una scelta di classi per determinate funzioni, per poter personalizzare MQSeries Everyplace aderendo ai requisiti specifici delle applicazioni. In alcuni casi le interfacce delle classi sono documentate per poter sviluppare ulteriori alternative. Tabella 15 riassume le possibilità. Le classi possono essere identificate in modo esplicito oppure utilizzando degli alias.

*Tabella 15. Opzioni di classe*

Classe	Alternative fornite	Interfacce documentate
amministrazione	no	sì
Strumenti di autenticazione	sì	no
Adattatore di comunicazioni	sì	sì

Tabella 15. Opzioni di classe (Continua)

Classe	Alternative fornite	Interfacce documentate
Stili di comunicazione	sì	no
Compressori	sì	no
Programmi di codifica	sì	no
Registrazione eventi	campione fornito	sì
Messaggi	no	sì
Memoria delle code	sì	no
Regole	classi predefinite fornite	sì
Traccia	campioni forniti	sì

## Caricamento applicazioni

Quando MQSeries Everyplace è configurato per funzionare come un client (o peer) l'applicazione di avvio è responsabile del caricamento di tutte le altre applicazioni in JVM. Le funzioni Java standard possono essere utilizzate a questo scopo, altrimenti è disponibile lo strumento di caricamento incluso come parte di MQSeries Everyplace. Quindi è possibile eseguire più applicazioni con un singolo programma di gestione code nello stesso JVM. Altrimenti è possibile utilizzare più JVM ma ciascuno richiede il proprio programma di gestione code, ciascuno dei quali deve avere un nome unico.

Quando MQSeries Everyplace è configurato come server, funziona anche come applicazione di avvio. MQSeries Everyplace supporta un elenco prestabilito di classi, che vengono caricate a turno, prima di caricare il programma stesso di gestione code.

---

## Capitolo 6. Reti MQSeries Everyplace e MQSeries

Sebbene possa esistere una rete MQSeries Everyplace indipendente, senza la necessità di un server o di una rete MQSeries, in pratica MQSeries Everyplace viene spesso utilizzato per completare un'installazione esistente di MQSeries, ampliando il raggio a nuove piattaforme e dispositivi, oppure fornendo funzioni avanzate come la sicurezza basata su code o messaggi, o la messaggistica sincrona. Da una prospettiva di applicazione di MQSeries Everyplace, le code e i relativi programmi di gestione di MQSeries possono essere considerati semplicemente come code remote o programmi di gestione code aggiuntivi. Tuttavia, esistono alcune limitazioni funzionali perché non è possibile accedere a queste code direttamente tramite canali dinamici di MQSeries Everyplace e un programma di gestione code di MQSeries Everyplace, ma è necessario un gateway MQSeries Everyplace. Il gateway può inviare messaggi a più programmi di gestione code di MQSeries in modo diretto o indiretto, attraverso canali client di MQSeries. In caso di collegamento indiretto, i messaggi passano attraverso canali client di MQSeries verso un programma di gestione code MQSeries intermedio, quindi proseguono attraverso canali di messaggi MQSeries verso il programma di destinazione.

I messaggi da un'applicazione di MQSeries destinata a MQSeries Everyplace vengono indirizzati al programma di gestione code e alla coda di MQSeries Everyplace in modo normale, con l'instradamento definito di MQSeries (definizioni di programma di gestione code remote), in modo che i messaggi di MQSeries Everyplace arrivino su code di trasmissione specifiche di MQSeries. I canali di MQSeries non sono definiti per le code di trasmissione, come vorrebbe la prassi normale; il gateway di MQSeries Everyplace infatti estrae i messaggi da queste code e ne garantisce la consegna alla destinazione MQSeries Everyplace. Il numero di code di trasmissione da utilizzare (numero di percorsi) è configurabile e dovrebbe essere impostato per riflettere il volume di messaggi da recapitare.

---

### Interfaccia per MQSeries

La struttura di MQSeries Everyplace supporta il concetto di uno o più collegamenti facoltativi tra MQSeries Everyplace e altri sistemi di messaggistica.

In MQSeries Everyplace Versione 1.0 è supportato un solo collegamento, il *collegamento MQSeries* che funziona da interfaccia tra le reti MQSeries Everyplace e MQSeries. Questo collegamento utilizza il client Java MQSeries come interfaccia per uno o più programmi di gestione code di MQSeries, consentendo ai messaggi di passare da MQSeries Everyplace a MQSeries e viceversa. In MQSeries Everyplace Versione 1 si consiglia un solo collegamento del genere per gateway, ciascuno associato a più *proxy di programmi di gestione code MQSeries* (definizioni di programmi di gestione code di MQSeries). E' richiesta una definizione di proxy di programma di gestione code per ciascun programma MQSeries che comunica con MQSeries Everyplace. Ciascuna definizione può avere uno o più *servizi di collegamento client* associati, di cui ognuno rappresenta un collegamento a un singolo programma di

gestione code di MQSeries. Ciascuno di questi può utilizzare un diverso collegamento del server MQSeries al programma di gestione code, e inoltre una serie diversa di proprietà come le uscite o porte utenti.

Un servizio di collegamento client per gateway può avere un numero di *listener* che utilizzano questo servizio gateway per il collegamento al programma di gestione code di MQSeries e per il recupero di messaggi da MQSeries a MQSeries Everyplace . Un listener utilizza un solo servizio per stabilire il collegamento, e ciascun listener si collega ad una sola coda di trasmissione sul relativo programma MQSeries. Ogni listener posta i messaggi da una singola coda di trasmissione di MQSeries a qualsiasi altro punto della rete MQSeries Everyplace tramite il programma principale di gestione code del gateway. In questo modo un singolo programma di gestione code del gateway può incanalare i messaggi MQSeries nella rete MQSeries Everyplace.

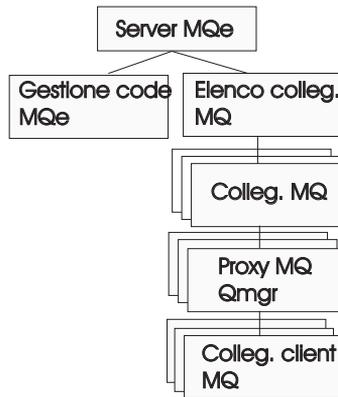


Figura 7. Gerarchia oggetti di collegamento MQSeries

Nello spostamento dei messaggi in un'altra direzione, da MQSeries Everyplace a MQSeries, il programma di gestione code del gateway configura uno o più oggetti *code di collegamento*. Ciascun oggetto coda di collegamento è in grado di collegarsi direttamente a un programma di gestione code e di inviare messaggi alla coda di destinazione. In questo modo, un gateway può distribuire messaggi di MQSeries Everyplace instradati attraverso un singolo programma di code MQSeries Everyplace ad un qualsiasi altro programma simile, in modo diretto o indiretto. L'oggetto collegamenti contiene le proprietà indicate in Tabella 16.

Tabella 16. Proprietà degli oggetti collegamenti

Proprietà	Spiegazione
Nomecollegamento	Elenco collegamenti
Stato esecuzione	Stato: in esecuzione o arrestato

L'oggetto collegamenti e altri oggetti del gateway possono essere avviati e arrestati indipendentemente dal programma di gestione code di MQSeries Everyplace. Se viene

avviato (o arrestato) un oggetto gateway l'operazione fornisce anche tutti gli elementi secondari (collegamenti, proxy del programma di gestione, collegamenti al client e listener delle code di trasmissione). L'oggetto collegamento contiene le proprietà indicate in Tabella 17 a pagina 48.

Tabella 17. Proprietà collegamenti

Proprietà	Spiegazione
Classe	Classe collegamenti
Trasformatore predefinito	La classe predefinita (classe di regole) da utilizzare per trasformare un messaggio da MQSeries Everyplace a MQSeries (o viceversa) se nessun'altra classe di trasformatore è associata alla coda di destinazione
Intervallo impulsi	L'unità di tempo base da utilizzare per operazioni con oggetti collegamento
Nome	Nome dell'oggetto collegamento
Stato esecuzione	Stato: in esecuzione o arrestato
Classe regole di avvio	Classe di regole utilizzate quando si avvia un oggetto collegamento
Elementi secondari proxy del programma di gestione code di MQSeries	Elenco di tutti i proxy dei programmi di gestione code in questo collegamento

In alcuni casi è possibile utilizzare un trasformatore predefinito (regola) per gestire tutte le conversioni dei messaggi. E' inoltre possibile impostare un trasformatore su una base listener (per messaggi da MQSeries a MQSeries Everyplace ) che ricopre quello predefinito. Per un controllo più specifico, è possibile impostare le regole di trasformazione sulla base di una coda di destinazione utilizzando definizioni di code di collegamento sul gateway; ciò è valido per le code di destinazione di MQSeries Everyplace e di MQSeries.

Il proxy del programma di gestione code MQSeries è dotato delle proprietà specifiche per un singolo programma di gestione code di MQSeries. Le proprietà del proxy sono indicate in Tabella 18.

Tabella 18. Proprietà proxy del programma di gestione code MQSeries

Proprietà	Spiegazione
Classe	classe proxy del programma di gestione code di MQSeries
nome host MQSeries	Il nome host IP utilizzato per stabilire collegamenti al programma di code di MQSeries tramite le classi client Java. Se non viene specificato, si assume che il programma di gestione code di MQSeries si trovi sullo stesso computer del collegamento e dei bind Java
nome proxy del programma di gestione code di MQSeries	Il nome del programma di gestione code di MQSeries
Nome del collegamento in corso	Nome dell'oggetto collegamento che gestisce questo proxy MQSeries
Stato esecuzione	Stato: in esecuzione o arrestato
Classe regole di avvio	Classe di regole utilizzate quando si avvia un oggetto programma di gestione code di MQSeries

*Tabella 18. Proprietà proxy del programma di gestione code MQSeries (Continua)*

<b>Proprietà</b>	<b>Spiegazione</b>
Elementi secondari di collegamento al client	Elenco di tutti gli oggetti di collegamento al client gestiti da questo proxy

Il servizio di collegamento al client contiene le informazioni dettagliate necessari per stabilire un collegamento a un programma di gestione code di MQSeries. Le proprietà del collegamento sono indicate in Tabella 19.

*Tabella 19. Proprietà del servizio di collegamento al client*

<b>Proprietà</b>	<b>Spiegazione</b>
Classe di adattatori	Classe da utilizzare come adattatore gateway
CCSID*	Il valore CCSID MQSeries da utilizzare
Classe	Classe di servizio di collegamento al client
Tempo massimo di collegamento inattivo	Il tempo massimo in cui un collegamento può restare inattivo prima di essere chiuso
password di MQSeries*	Password da utilizzare per il client Java
porta di MQSeries*	Il numero di porta IP utilizzato per stabilire collegamenti al programma di gestione code di MQSeries tramite le classi client Java. Se non viene specificato, si assume che il programma di gestione code di MQSeries si trovi sullo stesso computer del collegamento e dei bind Java
classe uscita di ricezione MQSeries *	Utilizzata per associare l'uscita di ricezione con l'altra estremità del canale client; l'uscita ha una stringa associata per consentire il passaggio dei dati al codice di uscita
classe uscita di sicurezza MQSeries *	Utilizzata per associare l'uscita di sicurezza con l'altra estremità del canale client; l'uscita ha una stringa associata per consentire il passaggio dei dati al codice di uscita
classe uscita di invio MQSeries *	Utilizzata per associare l'uscita di invio con l'altra estremità del canale client; l'uscita ha una stringa associata per consentire il passaggio dei dati al codice di uscita
id utente MQSeries *	Id utente per il client Java
Nome servizio di collegamento al client	Nome del canale di collegamento al server per il computer MQSeries
Nome del proxy del programma di gestione code corrente	Il nome proxy del programma di gestione code corrente
Classe regole di avvio	Classe di regole utilizzate quando si avvia un oggetto servizio di collegamento al client
Nome coda Sync	Il nome della coda di MQSeries utilizzata dal collegamento per scopi di sincronizzazione
Classe di regole per coda di sincr.	La classe di regole da utilizzare quando un messaggio si trova su una coda di sincr.
Stato esecuzione	Stato: in esecuzione o arrestato

Tabella 19. Proprietà del servizio di collegamento al client (Continua)

Proprietà	Spiegazione
Nome del collegamento in corso	Il nome dell'oggetto collegamento che gestisce questo collegamento al client
Elementi secondari del listener MQ XmitQ	Elenco di tutti gli oggetti del listener che utilizzano questo collegamento al client
*I dettagli su questi parametri sono reperibili nella documentazione <i>MQSeries Using Java</i>	

La *classe di adattatore* viene utilizzata per inviare messaggi da MQSeries Everyplace a MQSeries, mentre la *coda di sincr.* serve per tenere traccia dello stato dell'operazione. Il contenuto si utilizza in situazioni di recupero della messaggistica assicurata; dopo una normale chiusura la coda risulterà vuota. Può essere condivisa tra più collegamenti al client e tra più definizioni di collegamenti, purché le uscite di ricezione, invio e sicurezza siano le stesse. Questa coda si utilizza anche per memorizzare lo stato dei messaggi che vengono spostati da MQSeries a MQSeries Everyplace, a seconda delle proprietà del listener utilizzato. La *classe di regole per coda di sincr.* si utilizza quando un messaggio si trova sulla coda di sincr. e indica un errore di MQSeries Everyplace per confermare un messaggio.

Il tempo massimo di collegamento inattivo viene utilizzato per controllare l'insieme di collegamenti al client Java gestiti dal servizio di collegamenti al client per il sistema MQSeries. Quando un collegamento a MQSeries diventa inattivo, per mancanza di utilizzo, viene avviato un timer: il collegamento inattivo si chiuderà se il timer scatta prima che il collegamento venga riutilizzato. La creazione di collegamenti a MQSeries è un'operazione costosa, quindi questa procedura assicura che i collegamenti vengano riutilizzati senza consumare troppe risorse. Un valore zero indica che l'insieme di collegamenti non dovrebbe essere utilizzato.

L'oggetto listener, che passa i messaggi da MQSeries a MQSeries Everyplace, è dotato delle proprietà indicate in Tabella 20.

Tabella 20. Proprietà listener

Proprietà	Spiegazione
Classe	Classe listener
Nome coda delle lettere scadute	Coda utilizzata per conservare i messaggi da MQSeries a MQSeries Everyplace che non possono essere recapitati
Adattatore di memoria stato del listener	Nome classe dell'adattatore utilizzato per memorizzare informazioni sullo stato
Nome listener	Nome dei messaggi nella coda XMIT di MQSeries
Nome servizio corrente di collegamento al client	Nome servizio di collegamento al client
Stato esecuzione	Stato: in esecuzione o arrestato
Classe regole di avvio	Classe di regole utilizzate quando si avvia un oggetto listener
Classe trasformatore	Classe di regole utilizzate per determinare la conversione di un messaggio MQSeries ad un messaggio MQSeries Everyplace

Tabella 20. Proprietà listener (Continua)

Proprietà	Spiegazione
Classe regole di messaggi non recapitati	Classe di regole utilizzata per stabilire cosa fare quando non è possibile recapitare messaggi da MQSeries a MQSeries Everyplace
Secondi di attesa per il messaggio	Un'opzione avanzata per controllare le prestazioni del listener in casi particolari

La classe di regole messaggi non recapitati determina cosa fare quando non è possibile recapitare un messaggio da MQSeries a MQSeries Everyplace. In genere viene posizionata nella coda lettere scadute nel sistema MQSeries.

Per garantire la consegna dei messaggi, la classe del listener utilizza un adattatore di memoria stato del listener per memorizzare informazioni sullo stato, in un sistema MQSeries Everyplace o in una coda di sincr. del sistema MQSeries.

Per poter completare la configurazione del collegamento, sono necessarie le definizioni di coda remota e del relativo programma di gestione. Le definizioni di programma di gestione code remote per i programmi remoti di MQSeries Everyplace seguono le pratiche standard di MQSeries Everyplace; le definizioni dei programmi di gestione code remote per i programmi remoti di MQSeries hanno il canale impostato su un valore nullo, per indicare che non viene utilizzato un normale canale dinamico di MQSeries Everyplace, - invece viene definito un collegamento al programma di gestione code di MQSeries come indicato sopra.

Anche la definizione di coda remota per una coda MQSeries Everyplace segue la procedura standard, seppure profondamente modificata per la coda di MQSeries rispetto alle code di MQSeries Everyplace. Tabella 21 indica le proprietà per le code remote di MQSeries.

Tabella 21. Proprietà delle code remote di MQSeries

Proprietà	Spiegazione
Nomi alternativi (alias)	Nomi alternativi per la coda
Strumento di autenticazione	Deve essere nullo
Classe	Classe oggetto
Collegamento al client	Nome del servizio di collegamento al server da utilizzare
Compressore	Deve essere nullo
Programma di codifica	Deve essere nullo
Scadenza	Passato al trasformatore
Dimensione massima dei messaggi	Passato alla classe di regole
Modo	Deve essere sincrono
Proxy del programma di gestione code MQ	Nome del programma di gestione code di MQSeries a cui inviare per primo il messaggio
collegamento MQSeries	Nome del collegamento per inviare il messaggio a MQSeries

Tabella 21. Proprietà delle code remote di MQSeries (Continua)

Proprietà	Spiegazione
Nome	Nome con cui MQSeries Everyplace identifica la coda remota MQSeries
Programma di gestione code corrente	Programma di gestione code che gestisce la definizione
Priorità	Priorità per i messaggi (se non sovrascritta da un valore di messaggio)
Nome coda remota di MQSeries	Nome della coda remota di MQSeries
Regola	Classe di regole utilizzata per le operazioni sulle code
Destinazione programma di gestione code	Programma di gestione code di MQSeries che contiene la coda
Trasformatore	Nome della classe del trasformatore che converte il messaggio dal formato MQSeries Everyplace al formato MQSeries
Tipo	Coda di collegamento MQSeries

Le classi del *programma di codifica*, dello *strumento di autenticazione* e del *compressore* definiscono un insieme di attributi di code che stabiliscono il livello di sicurezza per tutti i messaggi che arrivano sulla coda. Dal momento in cui il messaggio viene inviato fino a quando viene spostato alla coda di collegamento di MQSeries, il messaggio è protetto almeno al livello di sicurezza delle code. Questi livelli di sicurezza *non* sono validi quando la coda di collegamento MQSeries trasferisce il messaggio al sistema MQSeries; durante il trasferimento vengono utilizzate le uscite di sicurezza, invio e ricezione nel collegamento al client. Non vengono eseguiti controlli per verificare che venga mantenuto il livello di sicurezza delle code.

Le code di collegamento di MQSeries sono solo sincrone; le applicazioni devono inviare messaggi a queste code asincrone tramite le code server home di memorizzazione e inoltro di MQSeries Everyplace o le definizioni asincrone di code remote.

L'amministrazione del gateway viene svolta nello stesso modo di quella per un normale programma di gestione code di MQSeries Everyplace, con messaggi di tipo amministrativo. Vengono definite nuove classi di messaggi come adatte all'oggetto gestito. Tabella 11 a pagina 26 indica le classi di messaggi di amministrazione per il gateway.

---

## Conversione messaggi

I messaggi di MQSeries Everyplace destinati a MQSeries passano attraverso il collegamento per essere convertiti in formato MQSeries, utilizzando un trasformatore predefinito o uno specifico per la coda di destinazione. Un trasformatore personalizzato offre una maggiore flessibilità, ad esempio è consigliabile utilizzare una classe secondaria della classe di messaggi di MQSeries Everyplace per rappresentare i messaggi di tipo particolare sulla rete di MQSeries Everyplace. Nel gateway un trasformatore può convertire il messaggio in un formato MQSeries utilizzando qualsiasi

corrispondenza adatta tra i campi e i valori di MQSeries, così come può aggiungere dati specifici per rappresentare il significato della classe secondaria.

Il trasformatore predefinito da MQSeries Everyplace a MQSeries non è in grado di sfruttare le informazioni della classe secondaria ma è stato progettato per una vasta gamma di situazioni. Presenta le caratteristiche seguenti:

- **Flusso di messaggi da MQSeries Everyplace a MQS:**

Il trasformatore predefinito da MQSeries Everyplace a MQSeries funziona insieme alla classe MQeMQMsgObject. Questa classe è una rappresentazione di tutti i campi che è possibile trovare nell'intestazione di un messaggio MQ. Utilizzando MQeMQMsgObject, l'applicazione può impostare i valori (come la priorità) utilizzando i metodi stabiliti. Quindi, quando MQeMQMsgObject (o un oggetto che deriva dalla classe MQeMQMsgObject) passa attraverso il trasformatore predefinito di MQSeries Everyplace, il trasformatore predefinito di MQSeries (MQeBaseTransformer) riceve i valori da MQeMSMsgObject e imposta i valori corrispondenti nel messaggio di MQSeries (per esempio, il valore di priorità viene copiato nel messaggio di MQSeries).

Se il messaggio trasferito non è un MQeMQMsgObject, e non deriva dalla classe MQeMQMsgObject, l'intero messaggio di MQSeries Everyplace viene copiato nel corpo del messaggio di MQSeries (*incanalato*). Il campo del formato messaggio nell'intestazione del messaggio di MQSeries viene impostato per indicare che il messaggio MQSeries contiene un messaggio in formato "incanalato" MQSeries Everyplace.

- **Flusso di messaggi da MQSeries a MQSeries Everyplace:**

I messaggi di MQSeries per MQSeries Everyplace sono gestiti in modo simile a quelli che si spostano nella direzione opposta. Il trasformatore predefinito esamina il campo del tipo di messaggio dell'intestazione di MQSeries e agisce di conseguenza.

Se l'intestazione MQSeries indica un messaggio "incanalato" MQSeries Everyplace, il corpo del messaggio di MQSeries viene ricostruito come il messaggio MQSeries Everyplace originale e viene inviato poi sulla rete MQSeries Everyplace.

Se invece non si tratta di un messaggio "incanalato" MQSeries Everyplace, si estrae il contenuto l'intestazione del messaggio MQSeries e si memorizza nell'oggetto MQeMQMsgObject. Il corpo del messaggio MQSeries viene considerato come campo di byte e inserito nell'oggetto MQeMQMsgObject. MQeMQMsgObject viene quindi inviato alla rete MQSeries Everyplace.

Il funzionamento di questa classe MQeMQMsgObject e del trasformatore predefinito indicano:

- Un messaggio MQSeries Everyplace può essere inviato da una rete MQSeries a una rete MQSeries Everyplace senza modifiche.
- Un messaggio MQSeries può essere inviato da una rete MQSeries Everyplace a una rete MQSeries senza modifiche.
- Un'applicazione MQSeries Everyplace può guidare qualsiasi altra applicazione MQSeries esistente senza alcuna modifica all'applicazione MQSeries.

---

## Funzione

Le code remote MQSeries sono abilitate per operazioni di messaggistica sincrona di MQSeries Everyplace, da un programma di gestione code MQSeries Everyplace; tutte le altre operazioni simili devono essere asincrone.

Non è possibile inviare messaggi di amministrazione MQSeries Everyplace ad un programma di gestione code di MQSeries. AdminQ non esiste e il formato del messaggio di amministrazione è diverso da quello utilizzato da MQSeries.

---

## Compatibilità

Una rete MQSeries Everyplace può esistere indipendentemente da MQSeries, ma in molte situazioni sono necessari entrambi i prodotti per rispondere ai requisiti dell'applicazione. MQSeries Everyplace può essere integrato in una rete MQSeries esistente con compatibilità, compresi gli aspetti elencati sotto:

- **Indirizzamento e denominazione:**
  - semantica di indirizzamento uguale utilizzando un indirizzo di coda o un programma di gestione code
  - Utilizzo comune di uno spazio nome ASCII.
- **Applicazioni:**MQSeries Everyplace è in grado di supportare le applicazioni MQSeries esistenti senza apportarvi modifiche.
- **Canali:**i gateway di MQSeries Everyplace utilizzano canali client di MQSeries.
- **Interscambio e contenuto messaggi:**
  - interscambio di messaggi tra MQSeries Everyplace e MQS
  - rete messaggi invisibile: i messaggi da MQSeries Everyplace o MQSeries possono viaggiare sull'altra rete senza alcuna modifica.
  - supporto reciproco per campi identificati nell'intestazione del messaggio MQSeries
  - consegna di messaggi assicurata per una sola volta
- **Applicazioni di esempio:**Interoperabilità tra MQSeries Postcard e MQSeries Everyplace Postcard <sup>11</sup>applicazioni

---

11. Questa applicazione non è compresa nella Versione 1.0 ma è disponibile da sito per lo scaricamento software di MQSeries sul World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

MQSeries Everyplace Versione 1 non supporta tutte le funzioni di MQSeries. A parte le considerazioni sull'ambiente, il sistema operativo e le comunicazioni, alcune delle differenze più significative sono ampiamente descritte di seguito. Tuttavia all'interno di MQSeries Everyplace è possibile svolgere molte delle attività di applicazioni con strumenti alternativi, utilizzando le funzioni di MQSeries Everyplace, oppure sfruttando le classi secondarie, sostituendo le classi fornite oppure utilizzando regole, interfacce e altre funzioni di personalizzazione all'interno del prodotto.

- Nessun supporto cluster
- Nessun supporto elenco di distribuzione.
- Nessun messaggio raggruppato o segmentato.
- Nessuna funzione di attesa o di bilanciamento carico del lavoro
- Nessun messaggio di riferimento.
- Nessuna opzione di relazione.
- Nessun supporto per code condivise
- Nessun avvio (trigger).
- Nessun supporto per unità di lavoro, nessuna coordinazione XA

Le caratteristiche di scalabilità e prestazioni sono diverse.

---

## **Consegna assicurata**

Anche se MQSeries Everyplace e MQSeries offrono entrambi la consegna assicurata, ciascuno fornisce diversi livelli di assicurazione. Quando un messaggio viene inviato da MQSeries Everyplace a MQSeries, il trasferimento è assicurato solo se si utilizza la combinazione di funzioni `putMessage` e `confirmPutMessage` (vedere "Operazioni dei programmi di gestione code" a pagina 24). Quando un messaggio viene inviato da MQSeries a MQSeries Everyplace, il trasferimento è assicurato solo se il messaggio MQSeries è definito permanente.



---

## Capitolo 7. Applicazioni e programmi di utilità

**Nota:** Queste applicazioni e programmi di utilità sono compresi nella Versione 1.0 ma non saranno disponibili dal sito per lo scaricamento del software MQSeries sul World Wide Web (<http://www-4.ibm.com/software/ts/MQSeries/>).

---

### Postcard

Postcard è un'applicazione di MQSeries Everyplace da utilizzare per convalidare il funzionamento di una rete MQSeries Everyplace indipendente o la collaborazione incrociata di reti MQSeries Everyplace e MQSeries. Postcard è un'applicazione Java che consente di inviare messaggi di testo a un utente su un programma di gestione code remote. Funziona in modo incrociato con Postcard, in modo che un messaggio Postcard inviato ad una destinazione MQSeries risulti come una cartolina ricevuta e viceversa.

Esiste anche una versione C dell'applicazione MQSeries Everyplace Postcard che si esegue su PalmOS e che può interagire con la versione Java.

---

### MQSeries Everyplace Explorer

MQSeries Everyplace Explorer è uno strumento di gestione, creato in Java, che consente la configurazione e l'utilizzo di programmi di gestione code locali e remote, code e messaggi. Si utilizzano le classi di creazione Microsoft® per presentare un'interfaccia grafica utente standard di Windows, di conseguenza possono essere utilizzate solo su piattaforme Windows. E' possibile comunque utilizzarle per gestire tutti i programmi di gestione code di MQSeries Everyplace in quanto funzionano esclusivamente con l'invio e la ricezione di messaggi di amministrazione. Si tratta di una vista a due pannelli di una rete MQSeries Everyplace: una struttura ad albero di oggetti nel pannello di sinistra e una vista a elenco dei dettagli oggetti nel pannello di destra.

MQSeries Everyplace Explorer dispone delle seguenti funzioni:

- Visualizzare o modificare le proprietà dei programmi di gestione code
- Creare, eliminare o modificare i collegamenti e visualizzarne le proprietà
- Creare, eliminare o modificare le code e visualizzarne le proprietà e/o il contenuto
- Cercare o eliminare messaggi, visualizzarne le proprietà e analizzare i campi
- Inviare messaggi di testo
- Configurare il collegamento tra MQSeries Everyplace e MQSeries

MQSeries Everyplace Explorer utilizza in genere un programma di gestione code già configurato e può caricare altre classi da eseguire. Se non esiste un programma di gestione code, ne crea uno con caratteristiche scelte dall'utente.

E' possibile eseguire più copie dello strumento su un solo computer, ciascuna per eseguire il proprio JVM. Questa soluzione consente di simulare una rete MQSeries Everyplace e può essere utilizzata per cercare e dimostrare reti e operazioni di MQSeries Everyplace.

---

## Capitolo 8. Interfacce di programmazione

*MQSeries Everyplace SPI (Systems Programming Interface)* è l'interfaccia di programmazione per MQSeries Everyplace. Sono disponibili due implementazioni, a seconda del sistema operativo. La versione Java fornisce accesso a tutte le funzioni di MQSeries Everyplace; l'interfaccia C in MQSeries Everyplace Versione 1.0 fornisce solo l'accesso a una serie secondaria. Le classi, i metodi e le procedure predefinite sono descritti in dettaglio in *Riferimento per la programmazione di MQSeries Everyplace*; vengono forniti esempi di programmazione di MQSeries Everyplace in *Manuale per la programmazione di MQSeries Everyplace*.



---

## Appendice. Informazioni particolari

Queste informazioni sono state sviluppate per prodotti e servizi offerti negli Stati Uniti E' possibile che negli altri paesi l'IBM non offra i prodotti, i servizi o le funzioni illustrati in questo documento. Consultare il rappresentante IBM locale per informazioni sui prodotti e sui servizi disponibili nel proprio paese. Ogni riferimento relativo a prodotti, programmi o servizi IBM non implica che solo quei prodotti, programmi o servizi IBM possano essere utilizzati. In sostituzione a quelli forniti dall'IBM, possono essere usati prodotti, programmi o servizi funzionalmente equivalenti che non comportino la violazione dei diritti di proprietà intellettuale o di altri diritti dell'IBM. E' responsabilità dell'utente valutare e verificare la possibilità di utilizzare altri programmi e/o prodotti, fatta eccezione per quelli espressamente indicati dall'IBM.

L'IBM può avere brevetti o domande di brevetto in corso relativi a quanto trattato nella presente pubblicazione. La fornitura di questa pubblicazione non implica la concessione di alcuna licenza su di essi. Chi desiderasse ricevere informazioni relative alle licenze può rivolgersi per iscritto a:

Director of Commercial Relations IBM Europe  
Schoenaicher Str. 220  
D-7030 Boeblingen  
Deutschland

**Il seguente paragrafo non è valido per il Regno Unito o per tutti i paesi le cui leggi nazionali siano in contrasto con le disposizioni in esso contenute:**

L'INTERNATIONAL BUSINESS MACHINES CORPORATION FORNISCE QUESTA PUBBLICAZIONE "NELLO STATO IN CUI SI TROVA", SENZA ALCUNA GARANZIA, ESPLICITA O IMPLICITA, IVI INCLUSE EVENTUALI GARANZIE DI COMMERCIALIZZABILITA' ED IDONEITA' AD UNO SCOPO PARTICOLARE. Alcune stati non consentono la rinuncia a garanzie esplicite o implicite in determinate transazioni; quindi la presente dichiarazione potrebbe essere non essere a voi applicabile.

Questa pubblicazione potrebbe contenere imprecisioni tecniche o errori tipografici. Le informazioni incluse in questo documento vengono modificate su base periodica; tali modifiche verranno incorporate nelle nuove edizioni della pubblicazione. L'IBM si riserva il diritto di apportare miglioramenti e/o modifiche al prodotto o al programma descritto nel manuale in qualsiasi momento e senza preavviso.

Tutti i riferimenti a siti Web non dell'IBM contenuti in questo documento sono forniti solo per consultazione. I materiali disponibile presso i siti Web non fanno parte di questo prodotto e l'utilizzo di questi è a discrezione dell'utente.

Tutti i commenti e i suggerimenti inviati potranno essere utilizzati liberamente dall'IBM e dalla Selfin e diventeranno esclusiva delle stesse.

Coloro che detengono la licenza su questo programma e desiderano avere informazioni su di esso allo scopo di consentire (i) uno scambio di informazioni tra programmi indipendenti ed altri (compreso questo) e (ii) l'uso reciproco di tali informazioni, dovrebbero rivolgersi a:

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

Queste informazioni possono essere rese disponibili secondo condizioni contrattuali appropriate, compreso, in alcuni casi, il pagamento di un addebito.

Il programma su licenza descritto in questo manuale e tutto il materiale su licenza ad esso relativo sono forniti dall'IBM nel rispetto delle condizioni previste dalla licenza d'uso.

---

## Marchi

I seguenti termini sono marchi della International Business machines Corporation.

AIX    AS/400    IBM    MQSeries    OS/390

Microsoft, Windows, Windows NT e i logo Windows sono marchi della Microsoft Corporation.

Java e tutti i marchi basati su Java sono logo e marchi della Sun Microsystems, Inc..

Altri nomi di società, prodotti e servizi possono essere marchi di altri.

---

## Glossario

Questo glossario descrive i termini utilizzati in questa pubblicazione e le parole che hanno un significato diverso da quello colloquiale. In alcuni casi una definizione può non essere l'unica riferita ad un termine, ma rende il significato particolare con cui questa parola viene utilizzata in questa pubblicazione.

Se non si riesce a trovare un termine, consultare l'indice oppure *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.

**accodamento messaggi.** Una tecnica di programmazione con cui ciascun programma di un'applicazione comunica con gli altri programmi posizionando i messaggi su delle code (accodamento).

**API (Application Programming Interface).** Un'API è costituita da funzioni e variabili che i programmatori possono utilizzare per le loro applicazioni.

**autenticatore.** Un programma che controlla gli utenti che inviano e ricevono i messaggi.

**bridge.** Un oggetto di MQSeries Everyplace che consente di inviare messaggi tra MQSeries Everyplace e altri sistemi di messaggistica, compreso MQSeries.

**browser Web.** Un programma che elabora e visualizza informazioni distribuite sul World Wide Web.

**canale.** Vedere *canale dinamico* e *canale MQI*.

**canale dinamico.** Un canale dinamico connette i dispositivi di MQSeries Everyplace, trasferisce i messaggi sincroni e asincroni e risponde in modo bidirezionale.

**canale MQI.** Un canale MQI collega un client MQSeries a un programma di gestione code su un server, trasferisce chiamate MQI e risponde in modo bidirezionale.

**classe.** Una classe è una raccolta generale di dati e metodi per operare sui dati. Una classe può

essere istanziata per produrre un oggetto che costituisce l'istanza della classe.

**classe principale.** Una classe principale è una classe ampliata in altre classi. I metodi pubblico e protetto e le variabili di una classe principale sono disponibili per la classe secondaria.

**classe secondaria.** Una classe secondaria è una classe che costituisce un ampliamento della classe principale. La classe secondaria eredita i metodi pubblico e protetto, oltre alle variabili della classe principale di appartenenza.

**client.** In MQSeries, un client è un componente runtime che consente di accedere ai servizi di accodamento su un server per applicazioni per utenti in locale.

**coda.** Una coda è un oggetto di MQSeries. Le applicazioni di accodamento messaggi possono mettere o ricevere messaggi su una coda

**coda di messaggi.** Vedere *coda*

**compressore.** Un programma che comprime un messaggio per ridurre il volume di dati da trasmettere.

**gateway.** Un gateway MQSeries Everyplace gateway (o server) è un computer che esegue il codice MQSeries Everyplace e comprende un programma di gestione canali.

**HTML (Hypertext Markup Language).** Un linguaggio utilizzato per definire informazioni da visualizzare sul World Wide Web.

**incapsulamento.** L'incapsulamento è una tecnica di programmazione orientata all'oggetto che protegge i dati di un oggetto e consente ai programmatori di accedere e manipolare i dati solo con richiami al metodo.

**interfaccia.** Un'interfaccia è una classe contenente solo metodi astratti, senza variabili di istanza. Un'interfaccia fornisce una serie di metodi diffusi da implementare con classi secondarie di alcune classi diverse.

**Internet.** Internet è una rete pubblica cooperativa di informazioni condivise. Fisicamente Internet utilizza una serie secondaria delle risorse totali di tutte le reti di telecomunicazioni pubbliche attualmente esistenti. Tecnicamente, ciò che caratterizza Internet come rete pubblica cooperativa è l'utilizzo di una serie di protocolli definiti TCP/IP (Transport Control Protocol/Internet Protocol).

**istanza.** Un'istanza è un oggetto. Quando una classe viene istanziata per produrre un oggetto, si dice che l'oggetto è un'istanza della classe.

**JDK (Java Developers Kit).** Un pacchetto software distribuito dalla Sun Microsystems per sviluppatori Java, che comprende il programma di interpretazione Java, classi e strumenti di sviluppo Java: compilatore, debugger, disassembler, appletviewer, generatore di file stub e generatore di documentazione.

**JNDI (Java Naming and Directory Service).** Un'API specificata nel linguaggio di programmazione Java. Fornisce funzioni di denominazione e directory per applicazioni scritte con linguaggio di programmazione Java.

**LDAP (Lightweight Directory Access Protocol).** LDAP è un protocollo client-server per avere accesso a un servizio di directory.

**messaggio.** Nelle applicazioni di accodamento messaggi, un messaggio è una comunicazione spedita tra programmi.

**messaggistica asincrona.** Un metodo di comunicazione tra programmi in cui i programmi stessi inseriscono i messaggi nelle code di messaggi. Con la messaggistica asincrona, il programma di invio funziona senza dover aspettare una risposta al messaggio. E' opposto alla *messaggistica sincrona*.

**messaggistica sincrona.** Un metodo di comunicazione tra programmi in cui i programmi stessi inseriscono i messaggi nelle code di messaggi. Con la messaggistica sincrona, il programma di invio deve aspettare una risposta al messaggio prima di riprendere il funzionamento. E' opposto alla *messaggistica asincrona*.

**metodo.** Il metodo è un termine di programmazione orientato all'oggetto per una funzione o una procedura.

**MQSeries.** MQSeries è una famiglia di programmi con licenza IBM che forniscono servizi di accodamento messaggi.

**oggetto.** (1) In Java, un oggetto è un'istanza di una classe. Una classe modella un gruppo di oggetti; un oggetto modella un determinato membro del gruppo. (2) In MQSeries, un oggetto è un programma di gestione code, una coda o un canale.

**pacchetto.** Un pacchetto in Java consente a una parte di codice Java di accedere a una serie specifica di classi. Il codice Java che costituisce un determinato pacchetto ha accesso a tutte le classi del pacchetto e a tutti i metodi e i campi non protetti delle classi.

**PDA (personal digital assistant).** Un personal computer di piccole dimensioni.

**privato.** Un campo privato non può essere visualizzato al di fuori della propria classe.

**programma di codifica.** Un programma che codifica un messaggio per garantire sicurezza durante la trasmissione.

**programma di gestione canali.** Un oggetto di MQSeries Everyplace che supporta i canali di comunicazione logica a più utenti tra gli utenti finali.

**programma di gestione code.** Un programma di gestione code fornisce alle applicazioni dei servizi di accodamento messaggi.

**protetto.** Un campo protetto può essere visualizzato solo all'interno della propria classe, in una classe secondaria o in pacchetti di cui la classe fa parte

**pubblico.** Una classe o un'interfaccia pubblica può essere visualizzata sempre. Un metodo o una variabile pubblica è visualizzabile in tutti i luoghi in cui può essere visualizzata la relativa classe

**server.** (1) Un MQSeries Everyplace server è un dispositivo dotato di un programma di gestione canali di MQSeries Everyplace. (2) Un server MQSeries è un programma di gestione code che fornisce servizi di accodamento messaggi alle applicazioni client in esecuzione su una stazione di lavoro remota. (3) Più in generale, un server è un programma che risponde alle richieste di informazione nel modello particolare per il flusso di informazioni a due programmi per client/server. (3) Il computer su cui viene eseguito un programma per server.

**servlet.** Un programma Java concepito per eseguire unicamente un server Web.

**TCP/IP (Transmission Control Protocol/Internet Protocol).** Una serie di protocolli di comunicazione che supporta funzioni di connessione peer-to-peer per reti di aree locali ed estese.

**Web.** Vedere World Wide Web.

**Web (World Wide Web).** Il World Wide Web è un servizio Internet, basato su una serie di protocolli comuni, che consente ad un server configurato in modo particolare di distribuire documenti su Internet secondo una procedura standard.



---

## Bibliografia

Pubblicazioni correlate:

- *MQSeries Everyplace - Istruzioni per l'installazione*, GC34-5862-00
- *Riferimento per la programmazione di MQSeries Everyplace*, SC34-5846-00
- *Manuale per la programmazione di MQSeries Everyplace*, SC34-5845-00
- *MQSeries An Introduction to Messaging and Queuing*, GC33-0805-01
- *MQSeries for Windows NT V5R1 Quick Beginnings*, GC34-5389-00



---

## Indice analitico

### A

a chi si rivolge questa pubblicazione ix  
adattatori, MQSeries Everyplace 29  
adattatori MQSeries Everyplace 29  
amministrazione con MQSeries Everyplace 25  
amministrazione MQSeries Everyplace 25  
applicazione, Postcard 57  
applicazione di Postcard 57  
applicazioni 57  
applicazioni, caricamento 44  
applicazioni, MQSeries Everyplace 11  
applicazioni di MQSeries Everyplace 11, 57  
asincrona, messaggistica 22, 32  
autoregistrazione 38

### C

canale, dinamico 7, 13, 28  
canali, client 7, 13  
canali client 7, 13  
canali dinamici 7, 13, 28  
caratteristiche 11  
caricamento applicazioni 44  
Cassi di MQSeries Everyplace 43  
classi, MQSeries Everyplace 43  
client, MQSeries 6  
client MQSeries 6  
cliente, requisiti 12  
code, collegamento MQSeries 20  
code, memorizzazione e inoltro 19  
code, MQSeries Everyplace 18  
code del server home 19  
code di collegamento di MQSeries 20  
code di memorizzazione e inoltro 19  
code di MQSeries Everyplace 18  
code locali 18  
code remote 18  
codifica 33  
collegamento, client-server 43  
collegamento, MQS 45  
collegamento, peer-to-peer 42  
collegamento client-server 43  
collegamento MQSeries 45  
collegamento peer-to-peer 42  
compatibilità con MQSeries 54  
compressione 33  
comunicazioni 42  
configurazione 40  
configurazioni, MQSeries Everyplace 30  
configurazioni di MQSeries Everyplace 30  
consegna di messaggi assicurata 55  
consegna messaggi, assicurata 55  
contenuti del prodotto 13

conversione messaggi 52

### D

descrizione 1  
destinatari ix  
Dispositivi, MQSeries Everyplace 7, 13  
dispositivi MQSeries Everyplace 7, 13

### E

entità, autenticabili 38  
entità autenticabili 38

### F

famiglia MQSeries 5  
formato dei dati dump 17

### G

gateway, MQSeries Everyplace 7, 13  
gateway MQSeries Everyplace 7, 13  
gestione dei collegamenti dialup 30

### I

informazioni, legali 61  
informazioni legali 61  
informazioni su questa pubblicazione ix  
interfacce, programmazione 59  
interfacce di programmazione 59  
interfaccia, sicurezza 40  
interfaccia di sicurezza 40  
interfaccia per MQSeries 45

### L

lettori ix  
listener di canali 42  
locali, code 18

### M

marchi 62  
messaggi, amministrazione 26  
messaggi, oggetti 14  
messaggi di amministrazione 26  
messaggistica, MQSeries 5  
messaggistica asincrona 22, 32  
messaggistica di MQSeries 5  
messaggistica diffusa 6  
messaggistica distribuita 5  
messaggistica host 5  
messaggistica per workstation 5  
messaggistica sincrona 22, 33  
mini-certificati 38  
MQeAttribute 36  
MQeMTrustAttribute 37  
MQSeries, compatibilità con 54

MQSeries, interfaccia per 45  
MQSeries Integrator 5  
MQSeries Workflow 5

## O

oggetti, MQSeries Everyplace 14  
oggetti di MQSeries Everyplace 14  
oggetti messaggi 14  
oggetto collegamenti 46  
oggetto collegamento 47  
oggetto listener 50, 51  
oggetto proxy del programma di gestione code 48  
operazioni, programma di gestione code 24  
operazioni dei programmi di gestione code 24

## P

panoramica 1  
personalizzazione 40  
prerequisiti ix, 3  
prodotto, contenuti 13  
programma di controllo 28  
programma di gestione canali 42  
programma di gestione code 42  
programma di utilità, MQSeries Everyplace Explorer 57  
programma di utilità di MQSeries Everyplace Explorer 57  
programma di utilità per amministrazione 57  
programma di utilità per amministrazione di MQSeries Everyplace 57  
programmi di gestione code 6  
programmi di gestione code, MQSeries Everyplace 22  
programmi di gestione code di MQSeries Everyplace 22  
programmi di utilità 57  
programmi di utilità di MQSeries Everyplace 57

## R

registrazioni eventi 30  
registro 37  
registro, MQSeries Everyplace 13  
registro, privato 38  
registro, pubblico 39  
registro di MQSeries Everyplace 37  
registro MQSeries Everyplace 13  
registro privato 38  
registro pubblico 39  
regole, MQSeries Everyplace 40  
regole compositore RAS 41  
regole del programma di gestione code 41  
regole di attributo 40  
regole di coda 41  
regole di collegamento MQSeries 40  
regole di MQSeries Everyplace 40  
remote, code 18  
replica certificati 39  
replica di certificati 39

requisiti del cliente 12  
reti, MQS 45  
reti, MQSeries Everyplace 30, 45  
Reti MQSeries 45  
Reti MQSeries Everyplace 30, 45

## S

scalabilità 30  
server, MQSeries 6  
server home, MQSeries Everyplace 19  
server MQSeries 6  
servizio di creazione mini-certificati 39  
servizio di creazione mini-certificati predefiniti 39  
sicurezza, a livello di messaggio 36  
sicurezza, basata sulle code 35  
sicurezza, locale 34  
sicurezza, MQSeries Everyplace 33  
sicurezza a livello di messaggio 36  
sicurezza basata sulle code 35  
sicurezza di MQSeries Everyplace 33  
sicurezza locale 34  
sincrona, messaggistica 22, 33  
sistemi operativi, supportati 3  
sistemi operativi richiesti 3  
sistemi operativi supportati 3  
SPI 59  
stili di collegamento 42  
stili di collegamento, multipli 43  
stili di collegamento multipli 43

## T

termini ix  
traccia di MQSeries Everyplace 30  
trasformatori 52

---

# Riservato ai commenti del lettore

**MQSeries® Everyplace**  
**Introduzione**  
**Versione 1**

**Pubblicazione N. GC13-2908-00**

Commenti relativi alla pubblicazione in oggetto potranno contribuire a migliorarla. Sono graditi commenti pertinenti alle informazioni contenute in questo manuale ed al modo in cui esse sono presentate. Si invita il lettore ad usare lo spazio sottostante citando, ove possibile, i riferimenti alla pagina ed al paragrafo.

Si prega di non utilizzare questo foglio per richiedere informazioni tecniche su sistemi, programmi o pubblicazioni e/o per richiedere informazioni di carattere generale.

Per tali esigenze si consiglia di rivolgersi al punto di vendita autorizzato o alla filiale IBM della propria zona oppure di chiamare il "Supporto Clienti" IBM al numero verde 167-017001.

I suggerimenti ed i commenti inviati potranno essere usati liberamente dall'IBM e dalla Selfin e diventeranno proprietà esclusiva delle stesse.

Commenti:

Si ringrazia per la collaborazione.

Per inviare i commenti è possibile utilizzare uno dei seguenti modi.

- Spedire questo modulo all'indirizzo indicato sul retro.
- Inviare un fax al numero: +39-081-660236
- Spedire una nota via email a: [translationassurance@selfin.it](mailto:translationassurance@selfin.it)

Se è gradita una risposta dalla Selfin, si prega di fornire le informazioni che seguono:

---

Nome

---

Indirizzo

---

Società

---

Numero di telefono

---

Indirizzo e-mail

Indicandoci i Suoi dati, Lei avrà l'opportunità di ottenere dal responsabile del Servizio di Translation Assurance della Selfin S.p.A. le risposte ai quesiti o alle richieste di informazioni che vorrà sottoporci. I Suoi dati saranno trattati nel rispetto di quanto stabilito dalla legge 31 dicembre 1996, n.675 sulla "Tutela delle persone e di altri soggetti rispetto al trattamento di dati personali". I Suoi dati non saranno oggetto di comunicazione o di diffusione a terzi; essi saranno utilizzati "una tantum" e saranno conservati per il tempo strettamente necessario al loro utilizzo.



Selfin S.p.A.  
Translation Assurance

Via F. Giordani, 7

80122 NAPOLI





Stampato in Italia

GC13-2908-00

