

MQSeries Everyplace for Multiplatforms



Introduction

version 1.2

MQSeries Everyplace for Multiplatforms



Introduction

version 1.2

Important

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section «Annexe. Remarques» à la page 89.

Troisième édition - mai 2001

Réf. US : GC34-5843-02

LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT". IBM DECLINE TOUTE RESPONSABILITE, EXPRESSE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE QUALITE MARCHANDE OU D'ADAPTATION A VOS BESOINS. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. Les informations qui y sont fournies sont susceptibles d'être modifiées avant que les produits décrits ne deviennent eux-mêmes disponibles. En outre, il peut contenir des informations ou des références concernant certains produits, logiciels ou services non annoncés dans ce pays. Cela ne signifie cependant pas qu'ils y seront annoncés.

Pour plus de détails, pour toute demande d'ordre technique, ou pour obtenir des exemplaires de documents IBM, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial.

Vous pouvez également consulter les serveurs Internet suivants :

- <http://www.fr.ibm.com> (serveur IBM en France)
- <http://www.can.ibm.com> (serveur IBM au Canada)
- <http://www.ibm.com> (serveur IBM aux Etats-Unis)

*Compagnie IBM France
Direction Qualité
Tour Descartes
92066 Paris-La Défense Cedex 50*

© Copyright IBM France 2001. Tous droits réservés.

© Copyright International Business Machines Corporation 2000, 2001. All rights reserved.

Table des matières

Avis aux lecteurs canadiens	v	Surveillance et actions connexes	46
À propos de ce manuel	vii	Canaux dynamiques	46
Lecteurs concernés	vii	Adaptateurs	47
Connaissances requises	vii	Gestion de connexion commutée	48
Terminologie	viii	Trace	48
Résumé des modifications.	xi	Journal des événements	48
Modifications apportées dans cette édition (GC11-1701-02)	xi	Livraison des messages	49
Modifications apportées dans cette édition (GC11-1701-01)	xi	Livraison asynchrone des messages	49
Chapitre 1. Généralités	1	Livraison synchrone des messages	50
Chapitre 2. Environnements logiciels	3	Sécurité	50
Plateformes prises en charge	3	Sécurité locale MQSeries Everyplace	52
Environnement Java	4	Sécurité MQSeries Everyplace au niveau des files d'attente	52
Mémoire requise	4	Sécurité au niveau message	53
Chapitre 3. La famille MQSeries	7	Registre	55
Produits hôte et distribués MQSeries	8	Entités MQSeries Everyplace authentifiables	55
MQSeries Everyplace	9	Registre privé et accréditations	56
Chapitre 4. Caractéristiques fonctionnelles du produit	13	Enregistrement automatique	56
Possibilités	13	Registre public et réplication de certificat	57
Applications	14	Utilisation des services de registre par les applications	57
Exigences des clients	14	Service d'émission de mini-certificats par défaut	57
Chapitre 5. Concepts du produit	17	Interface de sécurité	58
Introduction	17	Personnalisation	59
Objets messages	18	Règles	59
Format des données de vidage	24	Styles de connexions	60
Files d'attente	25	Connexion entre homologues	61
Gestionnaires de files d'attente	32	Connexion client-serveur	61
Configuration d'un gestionnaire de files d'attente	36	Styles de connexions multiples	62
Opérations des gestionnaires de files d'attente	39	Classes	62
Connexions	41	Chargement des applications	63
Administration	42	Chapitre 6. Réseaux MQSeries Everyplace et MQSeries	65
Messages d'administration	43	Interface avec MQSeries	65
Administration sélective	45	Conversion de message	74
		Fonction	76
		Compatibilité	76
		Livraison garantie unique	77
		Chapitre 7. Interfaces de programmation	79

Chapitre 8. Initiation à MQSeries	
Everyplace	81
Utilisation de MQSeries Everyplace	82
Acquisition d'expérience	83
Première utilisation du produit ES02 :	
MQe_Explorer	84
Annexe. Remarques.	89

Marques	91
Glossaire	93
Bibliographie	97
Index	99

Avis aux lecteurs canadiens

Le présent document a été traduit en France. Voici les principales différences et particularités dont vous devez tenir compte.

Illustrations

Les illustrations sont fournies à titre d'exemple. Certaines peuvent contenir des données propres à la France.

Terminologie

La terminologie des titres IBM peut différer d'un pays à l'autre. Reportez-vous au tableau ci-dessous, au besoin.

IBM France	IBM Canada
ingénieur commercial	représentant
agence commerciale	succursale
ingénieur technico-commercial	informaticien
inspecteur	technicien du matériel

Claviers

Les lettres sont disposées différemment : le clavier français est de type AZERTY, et le clavier français-canadien de type QWERTY.

OS/2 et Windows - Paramètres canadiens

Au Canada, on utilise :

- les pages de codes 850 (multilingue) et 863 (français-canadien),
- le code pays 002,
- le code clavier CF.

Nomenclature

Les touches présentées dans le tableau d'équivalence suivant sont libellées différemment selon qu'il s'agit du clavier de la France, du clavier du Canada ou du clavier des États-Unis. Reportez-vous à ce tableau pour faire

correspondre les touches françaises figurant dans le présent document aux touches de votre clavier.

France	Canada	Etats-Unis
 (Pos1)		Home
Fin	Fin	End
 (PgAr)		PgUp
 (PgAv)		PgDn
Inser	Inser	Ins
Suppr	Suppr	Del
Echap	Echap	Esc
Attn	Intrp	Break
Impr écran	ImpEc	PrtSc
Verr num	Num	Num Lock
Arrêt défil	Défil	Scroll Lock
 (Verr maj)	FixMaj	Caps Lock
AltGr	AltCar	Alt (à droite)

Brevets

Il est possible qu'IBM détienne des brevets ou qu'elle ait déposé des demandes de brevets portant sur certains sujets abordés dans ce document. Le fait qu'IBM vous fournisse le présent document ne signifie pas qu'elle vous accorde un permis d'utilisation de ces brevets. Vous pouvez envoyer, par écrit, vos demandes de renseignements relatives aux permis d'utilisation au directeur général des relations commerciales d'IBM, 3600 Steeles Avenue East, Markham, Ontario, L3R 9Z7.

Assistance téléphonique

Si vous avez besoin d'assistance ou si vous voulez commander du matériel, des logiciels et des publications IBM, contactez IBM direct au 1 800 465-1234.

À propos de ce manuel

Ce manuel est une présentation générale de MQSeries Everyplace for Multiplatforms (aussi appelé MQSeries Everyplace). Il traite des concepts du produit et de ses relations avec d'autres produits MQSeries.

Pour plus d'informations sur l'API MQSeries Everyplace et son utilisation pour créer des applications MQSeries Everyplace, reportez-vous aux manuels *MQSeries Everyplace for Multiplatform Programming Reference* et *MQSeries Everyplace for Multiplatforms Programming Guide*.

Pour toute information concernant l'utilisation d'autres langages de programmation avec MQSeries Everyplace for Multiplatforms, reportez-vous au manuel *MQSeries Everyplace for Multiplatforms Native Client Information*

Pour plus d'informations sur les procédures d'installation de MQSeries Everyplace for Multiplatforms, reportez-vous au document *MQSeries Everyplace for Multiplatforms - Lisez-moi d'abord*

Ce document est mis à jour périodiquement. Pour obtenir la dernière édition, consultez la page Web relative à la bibliothèque MQSeries, à l'adresse Internet suivante : <http://www.ibm.com/software/ts/mqseries/library/>.

Lecteurs concernés

Ce manuel est destiné aux personnes qui s'intéressent à l'utilisation d'applications de messagerie sécurisées sur des unités légères, telles que capteurs, téléphones, assistants numériques personnels (PDA) et ordinateurs portables, ou qui ont besoin d'étendre la portée d'un réseau de messagerie MQSeries Everyplace.

Connaissances requises

Aucune connaissance préalable n'est requise, mais une compréhension des concepts de messagerie sécurisée constitue un atout.

Si vous ignorez tout de ces concepts, nous vous conseillons de consulter le manuel MQSeries suivant :

- *Introduction à MQSeries*

Ce manuel est disponible sous forme électronique dans la section Manuels de la bibliothèque MQSeries en ligne. Cette bibliothèque est accessible via le site

Terminologie

Voici la définition de quelques termes utilisés tout au long de ce manuel :

Famille MQSeries

Désigne les produits MQSeries suivants :

- **MQSeries Workflow** simplifie l'intégration à l'échelle de l'entreprise en automatisant les processus métier qui impliquent personnes et applications.
- **MQSeries Integrator** est un puissant logiciel de routage de messages, qui assure l'acheminement intelligent, basé sur des règles, en temps réel, des messages, ainsi que la transformation et le formatage des contenus.
- **MQSeries Messaging** fournit une connectivité indifférenciée, du bureau à l'ordinateur principal, grâce à une messagerie professionnelle, avec plus de 35 plateformes prises en charge.

MQSeries Messaging

Désigne les groupes de produits de messagerie suivants :

- **Messagerie distribuée** : MQSeries pour Windows NT, AIX, AS/400, HP-UX, Sun Solaris et autres plateformes.
- **Messagerie hôte** : MQSeries pour OS/390
- **Messagerie diffuse** : MQSeries Everyplace

MQSeries

Désigne les trois groupes de produits de messagerie MQSeries suivants :

- Messagerie distribuée
- Messagerie hôte
- Messagerie de la station de travail

MQSeries Everyplace

Désigne le troisième groupe de produits de messagerie MQSeries, la messagerie diffuse.

Plateforme client

Petit ordinateur qui n'est capable d'exécuter MQSeries Everyplace qu'en tant que client.

Plateforme serveur

Ordinateur de toute taille capable d'exécuter MQSeries Everyplace en tant que serveur ou client.

Passerelle

Ordinateur de toute taille exécutant des programmes MQSeries
Everyplace incluant la fonction pont MQSeries.

Résumé des modifications

Cette section décrit les modifications apportées à la présente édition du manuel *MQSeries Everyplace for Multiplatforms - Introduction*. Ces modifications sont signalées par des barres verticales en marge gauche.

Modifications apportées dans cette édition (GC11-1701-02)

Modifications apportées dans cette édition (GC11-1701-01)

Les informations suivantes ont été ajoutées :

- Informations sur l'utilisation de MQSeries Everyplace sous AIX et Solaris.
- Conditions requises pour la mémoire.
- Formulaire de commentaire destiné aux lecteurs.

Chapitre 1. Généralités

MQSeries Everyplace fait partie de la famille MQSeries des produits de messagerie professionnelle. Il est conçu pour répondre non seulement aux besoins en messagerie des unités légères, telles que les capteurs, téléphones, assistants numériques personnels (PDA) et ordinateurs portables, mais aussi aux demandes de connexion des utilisateurs nomades et aux conditions induites par l'utilisation de réseaux de communication fragiles. Il maintient la qualité de service standard de MQSeries en offrant la livraison unique garantie, et permet l'échange de messages avec les autres membres de la famille. Dans la mesure où nombre d'applications MQSeries Everyplace sont exécutées sans la protection d'un pare-feu Internet, il propose également des fonctions de sécurisation sophistiquées.

Les unités légères requièrent un système de messagerie peu gourmand en ressources système. Par conséquent, MQSeries Everyplace est doté de fonctions et d'interfaces adaptées au type de clientèle concerné et n'est pas conçu pour offrir exactement les mêmes possibilités que les autres membres de la famille. En revanche, il comporte des fonctions spécifiques qui lui permettent de prendre en charge ses propres classes d'utilisateurs, telles qu'une sécurité globale, des objets messages, une messagerie synchrone et asynchrone, l'accès aux files d'attente éloignées et l'insertion et l'extraction de messages.

MQSeries Everyplace est également conçu pour bien s'intégrer avec les autres membres de la famille de produits d'informatique diffuse d'IBM et les autres composants de Websphere Application Server.

Chapitre 2. Environnements logiciels

Plateformes prises en charge

MQSeries Everyplace ne peut être installé directement que sur certaines plateformes serveur. Pour transférer les programmes et les classes Java sur d'autres plateformes, il est nécessaire d'utiliser un programme de téléchargement ou de transfert de fichiers approprié (non fourni).

Plateformes directement prises en charge, y compris au niveau de l'installation

Les outils intégrés permettent d'installer le produit sur les plateformes suivantes :

- Windows NT v4
- Windows 2000
- Windows 95/98/ME
- AIX version 4.3
- Sun Solaris version 7 ou 8
- Linux Intel Kernel 2.2 (installation à l'aide d'un fichier .zip).
- HP-UX 11.0 (installation à l'aide d'un fichier .zip).

Plateformes directement prises en charge, sauf au niveau de l'installation

Les plateformes suivantes sont prises en charge pour le test et le déploiement de MQSeries Everyplace, mais ne permettent son installation que par transfert de fichier à partir d'une autre plateforme :

- WinCE 2.1 sur unités HP Jornada (modèles 680 et 820)
- EPOC 32 bits Édition 5 sur unités Psion (5MX Pro et NetBook)
- PalmOS version 3.0 ou supérieure sur Palm V et IBM Workpad C3
- IBM 4690 OS avec Java

Plateformes prises en charge indirectement

Les plateformes ci-dessous peuvent être utilisées, mais elles ne sont prises en charge que si leur environnement Java est totalement compatible avec celui des plateformes directement prises en charge. Seules les plateformes testées répertoriées ci-dessus permettent l'identification et la résolution des incidents.

- Linux sur zSeries sous Kernel 2.2
- iSeries
- OS/2

environnements logiciels

- EPOC (sur des unités autres que celles répertoriées ci-dessus)
- WinCE (sur des unités autres que celles répertoriées ci-dessus)
- QNX Neutrino
- Pocket PC
- PalmOS (sur des unités autres que celles répertoriées ci-dessus)
- Toute autre plateforme équipée de l'un des environnements Java répertoriés à la section «Environnement Java»

Environnement Java

L'un des environnements d'exécution Java suivants est requis :

- Module d'exécution de Java IBM (JVM 1.3 ou version ultérieure), y compris Java Micro Edition.
- Tout environnement Java certifié compatible avec Sun Java (version 1.1 ou ultérieure) ¹

Remarque : Pour que vous puissiez bénéficier des services de maintenance, votre environnement Java doit être totalement compatible avec celui testé sur l'une des plateformes suivantes :

- Unités HP Jornada (modèles 680 et 820) équipées du système d'exploitation Windows CE
- Unités Psion (5MX Pro et NetBook) équipées du système d'exploitation EPOC
- Une des plateformes serveur directement prises en charge (voir plus haut)

Le pont MQSeries nécessite MQSeries Classes for Java. Assurez-vous que vous disposez du niveau de Java nécessaire pour l'exécution de MQSeries Classes for Java.

Mémoire requise

Le tableau suivant indique la quantité de mémoire requise pour l'installation de MQSeries Everyplace :

Tableau 1. Mémoire requise pour l'installation

Système d'exploitation	Mémoire requise
Windows NT (système de fichiers = NTFS)	26 Mo

1. Des incidents peuvent se produire si vous exécutez le programme d'installation dans la machine virtuelle Java (JVM) de Sun alors que le compilateur JIT (Just In Time) est activé. Si vous utilisez une JVM Sun, nous vous recommandons de désactiver ce compilateur à l'aide de la commande : `java -Djava.compiler=NONE install`

Tableau 1. Mémoire requise pour l'installation (suite)

Système d'exploitation	Mémoire requise
AIX	29 Mo
Solaris	27 Mo

Le tableau suivant indique la quantité de mémoire requise pour les fichiers de MQSeries Everyplace après l'installation :

Tableau 2. Mémoire requise pour MQSeries Everyplace

Système d'exploitation	Mémoire requise
Windows NT (système de fichiers = NTFS)	9,5 Mo
AIX	11 Mo
Solaris	10 Mo

Chapitre 3. La famille MQSeries

La famille MQSeries comporte de nombreux produits, offrant une vaste gamme de fonctions, comme illustré à la figure 1.

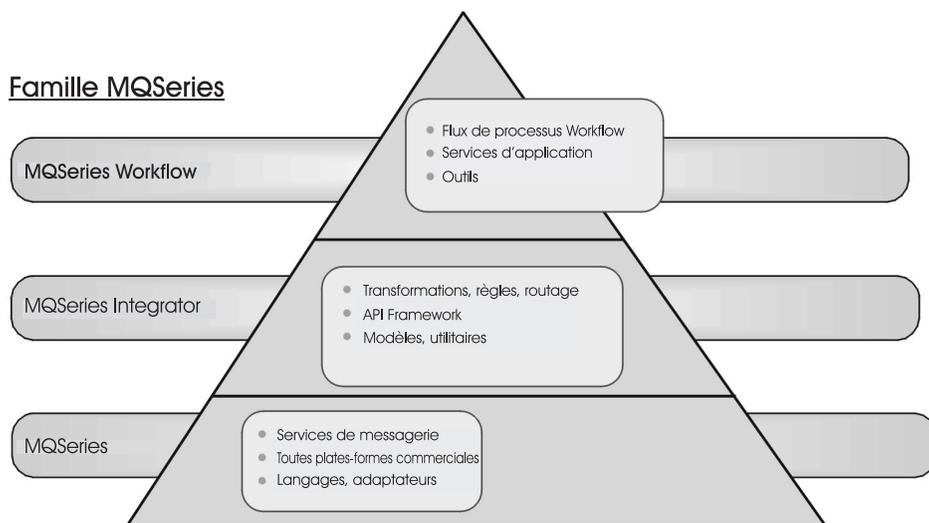


Figure 1. La famille MQSeries

- **MQSeries Workflow** simplifie l'intégration à l'échelle de l'entreprise en automatisant les processus métier qui impliquent personnes et applications.
- **MQSeries Integrator** est un puissant logiciel de routage de messages, qui assure l'acheminement intelligent, basé sur des règles, en temps réel, des messages, ainsi que la transformation et le formatage des contenus.
- **MQSeries Messaging** fournit une connectivité indifférenciée, du bureau à l'ordinateur principal, grâce à une messagerie professionnelle, avec plus de 35 plateformes prises en charge.

Les produits MQSeries Workflow et MQSeries Integrator bénéficient de la connectivité fournie par la couche de messagerie de MQSeries.

La messagerie de la famille MQSeries est fournie par les produits MQSeries et MQSeries Everyplace, chacun étant conçu pour prendre en charge une ou plusieurs plateformes serveur matérielles et/ou systèmes d'exploitation associés. Du fait de la variété des fonctionnalités des plateformes, ces produits

famille MQSeries

individuels sont structurés en groupes de produits, par fonction et conception. Ces groupes de produits sont au nombre de trois :

- **Messagerie distribuée** : MQSeries pour Windows NT, AIX, AS/400, HP-UX, Sun Solaris et autres plateformes
- **Messagerie hôte** : MQSeries pour OS/390
- **Messagerie diffuse** : MQSeries Everyplace

La messagerie en elle-même, indépendamment du produit ou du groupe de produits spécifique, est basée sur les gestionnaires de files d'attente. Ces gestionnaires gèrent des files d'attente qui peuvent contenir des messages. Les applications communiquent avec un gestionnaire de files d'attente locales, et récupèrent ou placent des messages dans les files d'attente. Si un message est placé dans une file d'attente éloignée, appartenant à un gestionnaire de files d'attente éloignées, le message est transmis à ce gestionnaire via des canaux. Ainsi, les messages peuvent passer par un ou plusieurs gestionnaires intermédiaires avant d'atteindre leur destinataire. Le principe de base de la messagerie est de "découpler" l'application expéditrice de l'application destinataire, en mettant les messages en files d'attente intermédiaires si nécessaire. Tous les produits de messagerie MQSeries reposent sur les mêmes éléments fondamentaux (gestionnaire de files d'attente, files d'attente, messages et canaux), malgré des différences notables dans les détails.

Produits hôte et distribués MQSeries

Les produits de messagerie hôte et distribuée MQSeries prennent en charge de nombreuses configurations réseau, qui impliquent toutes des clients et des serveurs² et dont la figure 2 illustre quelques exemples.

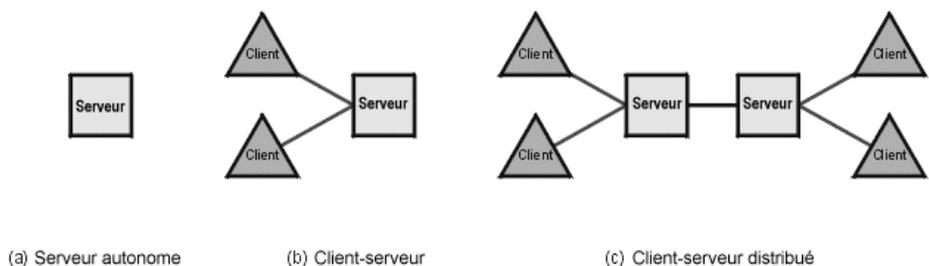


Figure 2. Quelques configurations hôte et distribuées simples

Dans le cas le plus simple, un serveur autonome exécutant un gestionnaire de files d'attente est configuré. Une ou plusieurs applications, exécutées sur ce

2. Notez que ces termes ont une signification spécifique au sein des produits de messagerie hôte, distribuée et de poste de travail MQSeries, qui ne correspond pas toujours à leur utilisation habituelle.

serveur, échangent des messages via des files d'attente. La configuration client-serveur est une autre configuration possible. Dans ce cas, le gestionnaire de files d'attente est localisé sur le serveur, mais est accessible aux clients par un *canal client*. Le canal client est une liaison bidirectionnelle sous un protocole MQSeries unique, implémentant un lien semblable à un appel de procédure éloignée (RPC). Les applications peuvent être exécutées sur les clients, avec accès aux files d'attente du serveur. Un des avantages de la configuration client-serveur est la légèreté de l'infrastructure de la messagerie du client, du fait qu'elle s'appuie sur le gestionnaire de files d'attente du serveur. L'inconvénient est que les clients et leur serveur associé opèrent de façon synchrone et requièrent de ce fait que le canal client soit disponible en permanence.

La configuration client-serveur distribuée est plus complexe, dans la mesure où plusieurs serveurs entrent en jeu. Dans ce type de configuration, l'échange de messages entre serveurs passe par des *canaux de transmission de messages*. Ces canaux de transmission sont unidirectionnels, avec un protocole conçu pour un échange sécurisé et asynchrone des données des messages. Les clients peuvent poursuivre leurs traitements sans que ces canaux de transmission soient disponibles, mais aucun message ne peut alors transiter entre les serveurs.

MQSeries Everyplace

MQSeries Everyplace prend en charge de nombreuses configurations de réseaux différentes par l'intermédiaire des gestionnaires de files d'attente, dont chacun dispose de possibilités qui lui sont propres. Le concept de client et de serveur y est inconnu, ce qui n'est pas le cas dans les produits hôte ou distribués MQSeries. Les gestionnaires de files d'attente MQSeries Everyplace peuvent jouer les rôles traditionnels de client ou de serveur, mais chacun d'eux n'est en fait qu'un gestionnaire de files d'attente capable d'effectuer des tâches définies par une application. Cela se traduit notamment par le fait qu'un gestionnaire de files d'attente MQSeries Everyplace peut être configuré avec ou sans files d'attente locales. S'il dispose de files d'attente locales, il peut stocker des messages localement et offrir aux applications des fonctions de messagerie asynchrone ; sinon, il ne peut leur offrir que des fonctions de messagerie synchrone. Il est également possible de configurer un gestionnaire de files d'attente avec ou sans fonctions de pont. S'il dispose de telles fonctions, il peut échanger des messages avec des gestionnaires de files d'attente hôte ou de poste de travail MQSeries ; sinon, il ne peut communiquer directement qu'avec d'autres gestionnaires de files d'attente MQSeries Everyplace (même s'il peut communiquer indirectement via ceux des gestionnaires de files d'attente du réseau qui disposent de fonctions de pont).

Remarque : Un nouveau nœud pour MQSeries Integrator (MQSI) permet la connexion directe de MQSeries Everyplace, sans utiliser le pont MQSeries.

Les gestionnaires de files d'attente MQSeries Everyplace utilisent des *canaux dynamiques* pour échanger des informations. Les propriétés de ces canaux dynamiques sont différentes de celles des *canaux client* et des *canaux de transmission de messages* utilisés par les autres membres de la famille. Voici quelques-unes des caractéristiques clés des canaux dynamiques :

- Support de la messagerie *synchrone* et *asynchrone* : La messagerie synchrone permet une transmission directe entre l'application source et la file d'attente cible, sans mise en file d'attente au niveau du gestionnaire de files d'attente source. La messagerie asynchrone permet une transmission entre le gestionnaire de files d'attente source et la file d'attente cible, éventuellement avec mise en file d'attente au niveau du gestionnaire de files d'attente source.
- Fourniture d'un *service de bout en bout* : Les canaux relient le gestionnaire de files d'attente source et un gestionnaire de files d'attente de destination, éventuellement en passant par des gestionnaires de files d'attente intermédiaires. Le protocole de transfert sous-jacent utilisé peut changer lorsque le canal passe par ces intermédiaires.
- Support de la *compression*, du *chiffrement* et de *l'authentification* : Les canaux bénéficient de ces sécurités pour protéger les données en transit.
- Support du fonctionnement *entre homologues* et *client-serveur* : Les canaux définis comme étant entre homologues sont fonctionnellement symétriques, de sorte que la source ou la cible peuvent indifféremment être à l'origine d'une opération sur le canal. En revanche, les canaux client-serveur sont de type demande/réponse, ce qui signifie que le client envoie une demande au serveur et que ce dernier y répond. (Cela ne restreint toutefois pas le flux des messages, qui peuvent être transmis du client au serveur et du serveur au client.)

L'option de configuration avec pont permet à un gestionnaire de files d'attente MQSeries Everyplace de communiquer avec des gestionnaires de files d'attente hôte et distribués MQSeries par l'intermédiaire de *canaux client*. Le composant pont gère un pool de canaux client qui peuvent être connectés à un ou plusieurs gestionnaires de files d'attente hôte ou distribués. Plusieurs gestionnaires de files d'attente MQSeries Everyplace avec pont peuvent être configurés dans un même réseau pour permettre d'obtenir la capacité, les performances et la fiabilité requises.

Quelques configurations MQSeries Everyplace classiques (mais arbitraires) sont illustrées dans les figures ci-dessous. Pour plus de clarté, seules sont représentées les connexions directes qui ont été définies. Les connexions indirectes qui les exploitent peuvent également être définies. Les canaux entre

homologues sont représentés pas des lignes terminées par une flèche à chacune de leurs extrémités, tandis que les canaux client-serveur sont représentés par des lignes terminées par une flèche unique orientée vers le serveur. Les clients peuvent utiliser les canaux client-serveur pour envoyer des messages au serveur et pour extraire de ce dernier les messages qui leur sont destinés. Les lignes non terminées par une flèche représentent les canaux client MQSeries qui permettent les communications entre MQSeries Everyplace et MQSeries.



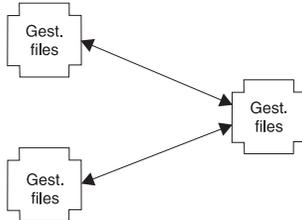
(a) Gestionnaire de files d'attente autonome

(a) Gestionnaire de files d'attente autonome utilisé pour prendre en charge une ou plusieurs applications faisant appel à des files d'attente pour échanger des données.



(b) Configuration d'égal à égal

(b) Deux gestionnaires de files d'attente interconnectés par un canal entre homologues.

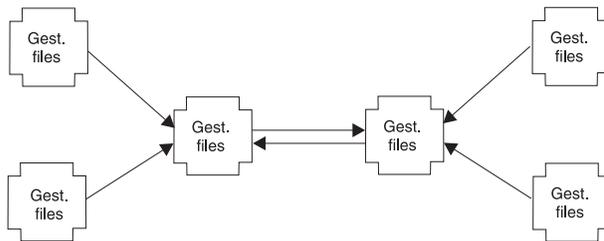


(b) Configuration d'égal à égal

(c) Deux gestionnaires de files d'attente (à gauche) connectés chacun par un canal entre homologues direct à un troisième gestionnaires de files d'attente. Les trois peuvent exploiter les connexions directes sous-jacentes pour

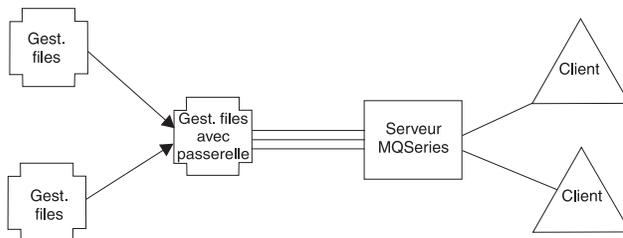
famille MQSeries

échanger des données entre eux.



(d) Un petit réseau

(d) Configuration en petit réseau, dans laquelle les deux gestionnaires de files d'attente serveur situés au centre de la figure utilisent une paire de canaux client-serveur directs pour échanger des informations. Les gestionnaires de files d'attente client utilisent chacun un canal client-serveur direct pour se connecter à l'un des gestionnaires de files d'attente serveur.



(e) Un réseau MQSeries intégré

(e) Configuration MQSeries Everyplace dans laquelle l'un des gestionnaires de files d'attente a été configuré avec l'option de pont et où l'ensemble des canaux client ont été dirigés vers un même serveur hôte/distribué MQSeries cible.

Chapitre 4. Caractéristiques fonctionnelles du produit

Ce chapitre décrit les exigences qui ont régi la conception et la mise en œuvre de MQSeries Everyplace.

Possibilités

MQSeries Everyplace étend la portée de la messagerie de la famille MQSeries :

- En prenant en charge les unités d'entrée de gamme, telles que les assistants numériques personnels (PDA), les téléphones ou les capteurs, et en les autorisant à intégrer un réseau de messagerie MQSeries. MQSeries Everyplace prend également en charge les unités intermédiaires telles que les ordinateurs portables, les postes de travail et les plateformes hôte et distribuées. MQSeries Everyplace offre la même qualité de service et la livraison garantie unique, et permet l'échange de messages avec d'autres membres de la famille.
- En offrant des fonctions de messagerie simples.
- En fournissant des fonctions de sécurisation étendues pour protéger les messages, les files d'attente et les données, pendant leur stockage ou leur transmission.
- En fonctionnant efficacement dans des environnements de communication hostiles, avec des réseaux instables ou une bande passante soumise à de fortes contraintes. MQSeries Everyplace est doté d'un protocole de communication par fil efficace et assure la reprise automatique en cas d'échec de la liaison.
- En prenant en charge les utilisateurs nomades, avec transfert des points de connectivité du réseau au fur et à mesure que les unités sont déplacées. MQSeries Everyplace contrôle également le fonctionnement dans des conditions inhabituelles : limitation ou défaillance des ressources de la batterie ou des réseaux.
- En opérant au travers de pare-feu configurés de manière adéquate.
- En réduisant au minimum les tâches d'administration de l'utilisateur, de sorte que la présence de MQSeries Everyplace sur une unité soit quasiment transparente. MQSeries Everyplace constitue ainsi une base adaptée à la construction d'applications de type utilitaire.
- Par la simplicité de sa personnalisation et son extension, via l'utilisation de règles fournies par l'application et d'autres classes qui modifient le comportement, ou via la transformation en sous-classes de classes d'objets de la base, par exemple, pour représenter différents types de messages.

caractéristiques fonctionnelles du produit

Applications

Les possibilités d'utilisation de MQSeries Everyplace sont nombreuses et variées, mais un grand nombre des applications du produit seront certainement des solutions personnalisées développées pour des groupes d'utilisateurs spécifiques. Voici quelques exemples d'utilisation :

- **Applications de vente au détail** : transmission au fur et à mesure des transactions de point de vente aux systèmes hôte, telles que des courtiers de messages.
- **Applications consommateur** : shopping en ligne via un PDA, collecte d'informations concernant les préférences des voyageurs en matière de compagnies aériennes, transactions financières à partir d'un téléphone portable.
- **Applications de contrôle** : collecte et intégration de données émises par les capteurs placés sur les pipelines pétroliers et transmises par satellite, télécommande d'équipement (soupapes, par exemple) sécurisée garantissant la validité de l'opérateur.
- **Travail mobile** : visite de professionnels (agent d'assurance), émission rapide d'un reçu client pour les entreprises de transport de colis, serveur de restauration rapide échangeant des informations avec la cuisine, suivi des scores d'un tournoi de golf, système de messagerie mobile sécurisé pour la police, informations professionnelles pour les ouvriers d'un chantier où la communication est souvent coupée, lecture des compteurs domestiques.
- **Productivité personnelle** : duplication de courrier/agenda, réplique de bases de données, décentralisation du traitement sur ordinateur portable.

Exigences des clients

Les exigences des clients qui ont influencé la conception de MQSeries Everyplace sont les suivantes :

- **Administration** : configuration et maintenance minimales ; prise en charge de l'administration locale et à distance ; possibilité d'extension et de personnalisation des fonctions administratives pour répondre aux besoins d'applications spécifiques ; amélioration de la reconnaissance et de la reprise automatiques ; disponibilité d'éléments d'administration indépendants, utilisables au choix.
- **Communications** : protocole par fil très efficace ; en-tête minimaux ; pas de zones obligatoires dans les messages (à l'exception d'un identificateur unique) ; possibilité de changer le codage des données ; support pour la compression, le chiffrement et l'authentification ; négociation de bout en bout des caractéristiques de compression et de sécurité ; possibilité de passer outre les pare-feu ; adaptateurs de communications enfichables.
- **Compatibilité** : qualité du service et fluidité des échanges de messages ; possibilité de communiquer avec d'autres systèmes MQSeries sans

caractéristiques fonctionnelles du produit

modification de l'application ; contrôle souple de l'échange de messages entre MQSeries et MQSeries Everyplace.

- **Empreinte du système** : pour les unités Palm, 64 ko. Pour les unités Java, la taille minimum des fichiers de classes est de 100 ko.
- **Fonction** : transfert de messages en mode synchrone et asynchrone, accès aux messages dans les files d'attente locales et éloignées ; possibilité d'utiliser n'importe quelle zone du message pour la recherche sélective ; contrôle sélectif du support de sauvegarde d'une file d'attente.
- **Support de règle** : contrôle de nombreuses facettes du comportement par le biais de règles : moment d'envoi des messages, nombre maximal de relances d'une communication, action à effectuer lorsqu'un message est trop long ou que la file d'attente cible est saturée.
- **Sécurité** : support intégral de la sécurité, de l'authentification et de l'irréfutableté ; sécurité au niveau message et au niveau file d'attente ; protection du système de messagerie contre les attaques ; sécurité enfichable utilisant les algorithmes standard de l'industrie ; possibilité d'intégration avec les accréditations utilisateur du système d'exploitation ; possibilité de mise en conformité avec les normes de sécurité nationales, avec changement possible du support de sécurité lorsque les messages passent les frontières.

caractéristiques fonctionnelles du produit

Chapitre 5. Concepts du produit

Introduction

Les éléments constitutifs du modèle de programmation de MQSeries Everyplace sont les messages, les files d'attente et les gestionnaires de files d'attente. Les messages MQSeries Everyplace sont des objets dont le contenu est défini par l'application. Stockés, ils sont maintenus en file d'attente et peuvent être déplacés au sein d'un réseau MQSeries Everyplace. Pour envoyer des messages vers une file d'attente cible, il suffit de préciser le gestionnaire de files d'attente cible et le nom de la file d'attente. Les applications placent les messages en files d'attente par le biais d'une opération "put", leur récupération étant généralement effectuée par une opération "get". Les files d'attente, locales ou éloignées, sont gérées par des gestionnaires de files d'attente. Les données de configuration sont stockées dans un registre.

L'arborescence des objets MQSeries Everyplace est visible dans la capture d'écran suivante, extraite de l'outil de gestion MQe_Explorer :

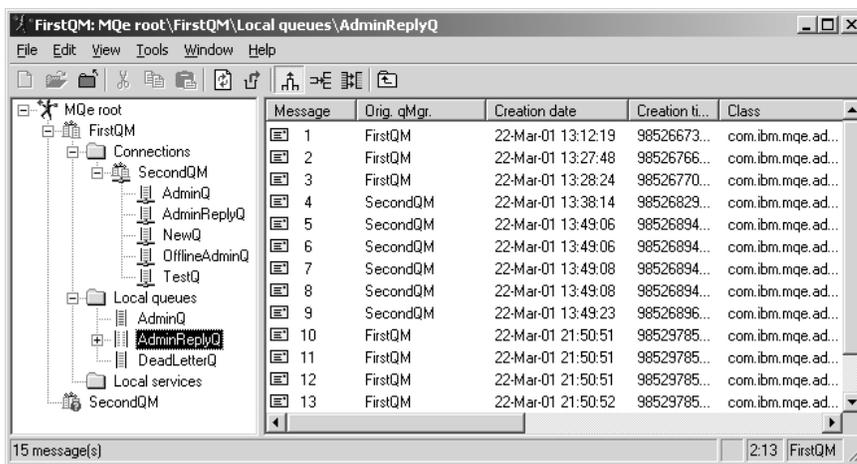


Figure 3. Objets MQSeries Everyplace affichés par MQe_Explorer

La sous-fenêtre de gauche présente les objets sous forme d'arborescence. Sous la racine MQe root se trouvent deux gestionnaires de files d'attente, FirstQM et SecondQM. L'arborescence de FirstQM est développée pour montrer les dossiers Connexions, Files d'attente locales et Local services. Le développement de l'arborescence de Connexions révèle une définition de connexion entre le gestionnaire de files d'attente SecondQM et le gestionnaire de files d'attente FirstQM. Le gestionnaire de files d'attente éloignées, SecondQM, comporte des

concepts du produit

définitions pour cinq des files d'attente de FirstQM. Le développement de l'arborescence de *Files d'attente locales* pour FirstQM révèle cinq files d'attente locales. L'une d'elles est sélectionnée et son contenu est affiché dans la sous-fenêtre de droite.

Pour comprendre la fonctionnement de MQSeries Everyplace, vous devez connaître la nature des messages, des files d'attente, des gestionnaires de files d'attente et des connexions. Ces concepts sont décrits dans les sections suivantes.

Objets messages

Les objets messages MQSeries Everyplace sont fondamentalement différents des messages pris en charge par MQSeries. Dans MQSeries, les messages sont des tableaux d'octets, divisés en en-tête de message et corps de message. L'en-tête de message, interprété par MQSeries, contient des informations essentielles, telles que l'identité de la file d'attente de réponse, le gestionnaire de la file d'attente de réponse, l'ID du message et l'ID de corrélation. Le corps du message n'est pas interprété.

En revanche, dans MQSeries Everyplace, les messages sont des *objets messages* hérités d'un objet MQSeries Everyplace appelé *objet zones*. Ces messages sont dépourvus d'en-tête et de corps mais ils possèdent des propriétés et des méthodes. Pour comprendre les objets messages, vous devez connaître les objets zones dont ils proviennent.

Ces objets zones, abondamment utilisés dans MQSeries Everyplace, sont un cumul de *zones* comportant chacune un nom, un type de données et les données elles-mêmes. Les noms de zones sont des chaînes de caractères ASCII (à l'exception de quelques caractères réservés) de longueur illimitée.

Les objets zones possèdent un *type*, c'est-à-dire une chaîne de caractères abrégée qui correspond au nom de la classe de l'objet de programmation. Les types de zones possibles sont les suivants :

Tableau 3. Types d'objets zones

Type	Description
ASCII	Chaîne ou tableau dynamique de chaînes ASCII
Boolean	Valeur booléenne
Byte	Tableau fixe ou dynamique d'octets
Double floating point	Valeur décimale double à virgule flottante, ou tableau fixe ou dynamique de valeurs décimales double à virgule flottante

Tableau 3. Types d'objets zones (suite)

Type	Description
Fields	Objets zones ou tableau dynamique d'objets zone (l'imbrication d'objets zones est admise)
Floating point	Valeur décimale à virgule flottante, ou tableau fixe ou dynamique de valeurs décimales à virgule flottante
Integer	Entier (4 octets) ou tableau fixe ou dynamique d'entiers
Long integer	Entier long (8 octets) ou tableau fixe ou dynamique d'entiers longs
Object	Valeur
Short integer	Entier court (2 octets) ou tableau fixe ou dynamique d'entiers courts
UNICODE	Chaîne ou tableau dynamique de chaînes UNICODE

Les objets zones prennent en charge plusieurs appels de méthode. Par exemple :

- Énumération de zones
- Copie d'une ou plusieurs zones constitutives (avec l'option de remplacement)
- Comparaison d'objets zones
- Opérations Put et Get sur des zones constitutives
- Inspection d'objets zones (par exemple, "L'objet comporte-t-il une zone constitutive ?")
- Vidage (dump) et restauration (restore) d'un tableau d'octets

Les méthodes de vidage et de restauration sont particulièrement importantes. Par exemple, la méthode de vidage est appelée pour fournir les données nécessaires à la transmission d'un objet message sur une liaison, et la méthode de restauration permet la recréation de cet objet sur le récepteur. Ainsi, chaque objet est responsable de son propre format de transmission, et les méthodes par défaut fournies peuvent être remplacées par d'autres afin de mettre en œuvre un comportement personnalisé. De même, ces deux méthodes sont utilisées lorsque des objets messages sont sauvegardés ou restaurés à partir d'une file d'attente. Le remplacement de ces méthodes permet de mettre en œuvre un comportement très différent. Par exemple, un objet message peut interroger une base de données lors du vidage pour extraire sa valeur avant sa transmission.

objets messages

Le tableau suivant répertorie certaines propriétés des objets zones et de leurs zones constitutives :

Tableau 4. Propriétés des objets zones et de leurs éléments constitutifs

Propriété	Présence	
	Objets zones	Zones
Objet attribut associé	facultatif	
Zone(s) constitutive(s)	oui	
Caché		oui
Nom		oui
Type	oui	oui
Valeur		oui

Deux propriétés sont particulièrement remarquables : *Caché*, qui permet d'ignorer une zone lors d'une comparaison, et la possibilité d'*associer un objet attributs* à un objet zones.

Les *objets attributs*, au coeur du modèle de sécurité MQSeries Everyplace, autorisent l'accès sélectif au contenu et sa protection. Ils possèdent les propriétés importantes suivantes :

Tableau 5. Propriétés de l'objet attributs

Propriété	Description
Authentification :	Contrôle l'accès.
Chiffrement :	Protège le contenu lorsque l'objet est vidé (et autorise sa restauration).
Compression :	Réduit la mémoire requise (pour la transmission et/ou le stockage).
Règle :	Contrôle les opérations autorisées.

Les objets messages héritent des objets zones et offrent donc toutes les possibilités ci-dessus, y compris celle d'associer des objets attributs au niveau de l'objet message et de ses zones constitutives. En outre, ils comportent un UID (identificateur unique) généré par MQSeries Everyplace. Cet UID identifie chaque objet message de manière unique dans l'ensemble du réseau MQSeries Everyplace et est composé :

- **du nom** du gestionnaire de files d'attente émetteur (ajouté par le gestionnaire de files d'attente à réception de l'objet). Ce nom doit être globalement unique.
- **de l'heure** de création de l'objet message (ajouté à la création).

Un objet message possède donc les propriétés suivantes, en plus de celles héritées de l'objet zones dont il provient :

Tableau 6. Propriétés de l'objet message

Propriété	Description
Gestionnaire de files d'attente émetteur	Nom du gestionnaire de files d'attente qui a reçu le premier le nouveau message provenant de l'application.
Heure de création	Heure de création de l'objet message par l'application.
UID	Identificateur MQSeries Everyplace unique.

Aucune autre information n'est requise dans un message destiné à un autre gestionnaire de files d'attente MQSeries Everyplace, mais d'autres zones seront généralement présentes. Il peut s'agir :

- de zones ajoutées par MQSeries Everyplace pour refléter l'état en cours,
- de zones associées à une sous-classe de message particulière,
- de zones personnalisées associées à une instance d'objet message.

MQSeries Everyplace ajoute des zones à un objet message (et les supprime de celui-ci) afin de permettre les opérations de messagerie et de mise en file d'attente. Par exemple, lorsqu'un message se trouve dans une file d'attente, il est possible de l'interroger pour savoir quand il y a été placé. (Cette opération est rendue possible par une règle de file d'attente décrite plus loin.) Lorsqu'un message est transmis par un gestionnaire de files d'attente à un autre gestionnaire de files d'attente, une zone de réexpédition peut y être ajoutée pour signaler que les données ont été retransmises. Les exemples d'utilisation de ces zones supplémentaires sont nombreux.

Généralement, les messages d'applications sont des instances d'un descendant de la classe d'objet message de base et, à ce titre, comportent des zones supplémentaires adaptées à leur usage (par exemple, numéro de facture). Certaines de ces zones seront génériques et communes à de nombreuses applications, telles que la zone contenant le nom du gestionnaire de files d'attente de réponse. Par conséquent, MQSeries Everyplace prend en charge les zones susceptibles d'être présentes dans les messages. Ces zones sont les suivantes :

Tableau 7. Zones d'objet message avec provision

Nom de la zone	Utilisation
Action	Utilisée par la fonction d'administration pour indiquer les actions telles que l'interrogation, la création et la suppression.

objets messages

Tableau 7. Zones d'objet message avec provision (suite)

Nom de la zone	Utilisation
Correlation ID (ID de corrélation)	Chaîne de caractères binaires généralement utilisée pour associer une réponse au message d'origine.
Errors (Erreurs)	Utilisée par la fonction d'administration pour renvoyer les informations d'erreur.
ExpireTime (Délai d'expiration)	Délai au terme duquel le message peut être supprimé (même s'il n'a pas été livré).
Lock ID (ID de verrouillage)	Clé requise pour déverrouiller le message.
Message ID (ID du message)	Identificateur unique du message.
Originating queue manager (Gestionnaire de files d'attente émetteur)	Nom du gestionnaire de files d'attente qui a envoyé le message.
Parameters (Paramètres)	Utilisée par la fonction d'administration pour transmettre des informations d'administration.
Priority (Priorité)	Priorité relative de transmission du message.
Reason (Raison)	Utilisée par la fonction d'administration pour renvoyer les informations d'erreur.
Reply-to queue (File d'attente de réponse)	Nom de la file d'attente dans laquelle la réponse au message doit être envoyée.
Reply-to queue manager (Gestionnaire de files d'attente de réponse)	Nom du gestionnaire de files d'attente auquel la réponse au message doit être envoyée.
Resend (Réexpédition)	Indique que le message est une réexpédition d'un message précédent.
Return code (Code retour)	Utilisée par la fonction d'administration pour transmettre l'état d'une opération d'administration.
Style	Distingue les commandes des demandes/réponses, etc.
Wrap message (Encapsulage du message)	Messages encapsulé pour garantir la protection des données.

Dans tous les cas, une constante définie permet de transférer le nom de la zone dans un seul octet. Toutefois, des dispositions plus étendues sont prises pour certaines zones. Par exemple, la priorité (le cas échéant) joue sur l'ordre de transmission des messages, l'ID de corrélation déclenche l'indexation d'une file d'attente sur les valeurs de cette zone pour accélérer les recherches, le délai d'expiration déclenche la péremption du message, etc.

L'écran d'administration représenté figure 4 à la page 23 illustre la structure d'un message. Notez que l'utilisation de zones imbriquées signifie que des

hiérarchies de données complexes peuvent exister dans les messages et que les outils d'administration (et les applications) peuvent explorer en profondeur la structure autodéscriptive (sous réserve que cette faculté n'ait pas été inhibée dans les paramètres de sécurité).

Les objets messages utilisent les méthodes de vidage (dump) et de restauration (restore) sous-jacentes lorsqu'ils sont sérialisés vers ou depuis un tableau d'octets en vue de leur transmission ou de leur stockage en file d'attente. Par défaut, afin de réduire le volume d'informations, la définition de classe associée ne fait pas partie du flux. Le remplacement de ces méthodes par des méthodes de sérialisation standard permet d'intégrer cette définition dans le flux lorsque cela s'avère nécessaire.

Name	Data type	Value	Length	Array type	Modifier	Hidden
1 admact	Int	6			0	0
2 admerrs	Fields	...			0	0
3 admmaxtry	Int	1			0	0
4 admparms	Fields	...			0	0
5 admqmgr	Ascii	SecondQM	8		0	0
6 admrc	Byte	00			0	0
7 admreason	Unicode	Good	4		0	0
8 admrity	Int	0			0	0
'MQE_Explorer'	Boolean	true			10000	0
'Msg_CorrelID'	Byte array	{00, 00, 00, E5, ...}	8	Static	0	0
'Msg_OriginQMgr'	Ascii	FirstQM	7		0	0
'Msg_ReplyToQ'	Ascii	AdminReplyQ	11		0	0
'Msg_ReplyToQMgr'	Ascii	FirstQM	7		0	0
'Msg_Style'	Int	1			0	0
'Msg_Time'	Long	985267704330			0	0

Figure 4. Structure de message affichée dans MQE_Explorer

Cependant, la classe est généralement rendue disponible de façon indépendante sur chacun des gestionnaires de files d'attente sur lesquels l'objet message doit être instancié. S'il est nécessaire d'instancier un objet (par exemple, en vue d'un stockage intermédiaire lors de sa transmission) et s'il est déterminé que le fichier classe est indisponible, le message peut être encapsulé dans une autre classe. Celle-ci est généralement la classe d'objet message par défaut. Cette technique est également utile lorsqu'un objet attributs a été utilisé pour protéger le message. La présence d'un tel objet signifie qu'il est impossible d'accéder au contenu du message sans les clés de sécurité requises. Toutefois, l'encapsulation d'un tel message permet au message noyau de rester totalement protégé, tandis que le message encapsulé peut être librement vidé et restauré.

La méthode de vidage d'objet message par défaut a été optimisée pour minimiser la taille de la chaîne de caractères binaires générée, afin d'optimiser le stockage et la transmission de message.

Format des données de vidage

Le format des données de vidage par défaut code les zones comme suit :

{Longueur Identificateur Frontière {Données}} {Longueur Identificateur Frontière {

où :

- *Données* : valeur des données. Les entiers sont compressés par suppression des 0 et des F de début. Les valeurs booléennes ne sont pas associées à des octets de données.
- *Frontière* : octet spécial délimitant la frontière entre l'identificateur et l'élément Data facultatif. Cet octet indique également le type de l'élément Data.
- *Identificateur* : conserve le nom de zone dans une chaîne ASCII de longueur variable, terminée par un octet de fin.
- *Longueur* : indique la longueur de la zone de données, qui peut varier de 1 à 4. Les deux premiers bits du premier octet sont réservés à la longueur de la zone Longueur. Longueurs admises : entre 0 et 1 073 741 823.

Le résultat est un flot de données extrêmement compact. La compression des données permet d'économiser encore de la place. La compression XOR avec un flot d'octets antérieur peut donner de bons résultats, mais du fait de la nature et de l'ordre variables de ces zones, un XOR simple peut ne pas produire les résultats escomptés. MQSeries Everyplace intègre un XOR intelligent, fonctionnant zone par zone, qui se révèle mieux à même d'améliorer la compression.

Files d'attente

Les files d'attente sont essentiellement utilisées pour contenir les objets messages en attendant que les programmes d'application les récupèrent. Comme les messages, les files d'attente dérivent d'objets zones. L'accès direct des applications à l'objet file d'attente n'est normalement pas autorisé. En effet, le gestionnaire de files d'attente joue le rôle d'intermédiaire entre les programmes d'application et les files d'attente. Les files d'attente sont identifiées par leur nom, chaîne de caractères ASCII de taille non limitée³, mais qui doit être unique au sein d'un gestionnaire de files d'attente spécifique. MQSeries Everyplace prend en charge de nombreux types de file d'attente :

Files d'attente locales

Les files d'attente locales sont utilisées par les applications pour stocker les messages en toute sécurité. Le message stocké est mappé en mémoire permanente par un adaptateur. Ce mappage s'effectue file par file. Un jeu d'adaptateurs est fourni avec MQSeries Everyplace, mais vous pouvez en créer vous-même ou vous en procurer auprès d'autres sources. L'adaptateur standard, MQeDiskFieldsAdapter, mappe les files d'attente dans le système de fichiers local et met en œuvre la garantie de livraison. L'adaptateur MQeReducedDiskFieldsAdapter mappe également les files d'attente dans le système de fichiers mais offre des performances meilleures. Toutefois, son efficacité dépend de la survie du système d'exploitation car il délègue à ce dernier la tâche de vider ses mémoires tampons dans le sous-système de disques physique. Enfin, l'adaptateur MQeMemoryFieldsAdapter mappe les files d'attente en mémoire. Bien qu'il offre les performances de file d'attente les plus élevées, il présente le désavantage que les messages qu'il traite ne survivent pas à un redémarrage du système d'exploitation ou du gestionnaire de files d'attente. La création d'un adaptateur approprié permet de définir l'emplacement de stockage des messages voulu pour chaque file d'attente. Les bases de données relationnelles et les CD inscriptibles sont des exemples de supports de stockage. Pour le stockage des files d'attente, certains adaptateurs tirent parti du système de fichiers miroir du contrôleur de magasin de détail IBM 4690, ou utilisent DB2.

Les files d'attente locales peuvent être utilisées en ligne ou hors ligne (c'est-à-dire connectées ou non à un réseau). Il est également possible de définir des attributs de sécurité pour les files d'attente, à l'aide d'une procédure très similaire à celle utilisée pour protéger les

3. Pour une meilleure interopérabilité il est conseillé toutefois de respecter les conventions de dénomination MQSeries, et se limiter les noms à 48 caractères. Le système de fichiers utilisé peut également avoir une incidence sur la longueur.

messages et les objets zones avec un objet attributs. La sécurité des files d'attente est décrite à la section «Sécurité» à la page 50. L'accès aux messages d'une file d'attente locale s'effectue toujours de manière synchrone, ce qui signifie que l'application attend que MQSeries Everyplace ait terminé l'opération de placement (put) ou d'extraction (get) en cours.

Files d'attente éloignées

Une file d'attente éloignée est une référence locale à une file d'attente qui réside sur un gestionnaire de files d'attente éloignées. Cette référence locale porte le même nom que la file d'attente cible mais la définition de la file d'attente éloignée identifie le gestionnaire de files d'attente propriétaire de cette file. Les files d'attente éloignées ont également des propriétés qui contrôlent l'accès à leur contenu, telles que le mode d'accès (synchrone ou asynchrone), les caractéristiques de sécurité et les options de transmission.

MQSeries Everyplace peut établir automatiquement les files d'attente éloignées. Lorsqu'une tentative est effectuée pour accéder à une file d'attente située sur un autre gestionnaire de files d'attente (par exemple, pour lui envoyer un message), MQSeries Everyplace recherche la définition de cette file d'attente éloignée. Si cette définition existe, elle est utilisée. Sinon, MQSeries Everyplace exécute une procédure de *reconnaissance de file d'attente*. Au cours de cette procédure, les caractéristiques de la file (authentification, chiffrement et compression) sont reconnues et une définition de file d'attente éloignée est créée. Cette reconnaissance ne peut être effectuée que si la file d'attente cible est accessible. Si elle ne l'est pas, une définition de file d'attente éloignée doit être fournie d'une autre manière. Lors de la procédure de reconnaissance, MQSeries Everyplace définit le mode d'accès comme étant synchrone, puisque la file d'attente a été déterminée comme étant disponible de façon synchrone.

Une file d'attente éloignée *synchrone* est une file d'attente qui n'est accessible que lorsqu'elle est connectée à un réseau qui dispose d'un chemin de communication vers le gestionnaire propriétaire de la file d'attente. Si la connexion au réseau n'est pas établie, les opérations telles que put, get et browse (reportez-vous au tableau «Opérations des gestionnaires de files d'attente» à la page 39) provoquent une erreur. La file d'attente propriétaire contrôle les droits d'accès et les conditions de sécurité requises pour accéder à la file d'attente. C'est à l'application que revient la gestion des erreurs ou des tentatives d'envoi et de réception des messages, car dans ce cas MQSeries Everyplace n'assure plus la livraison garantie unique.

Une file d'attente éloignée *asynchrone* est une file d'attente qui envoie des messages à des files d'attente éloignées mais qui ne peut en

extraire des messages à distance. Si la connexion réseau est établie, les messages sont alors envoyés au gestionnaire de files d'attente propriétaire et à la file d'attente. Sinon, les messages sont stockés localement jusqu'à établissement de la connexion, puis transmis. Cela permet aux applications d'utiliser la file d'attente lorsque l'unité associée est hors ligne. Par conséquent, les files d'attente de ce type possèdent un adaptateur qui effectue un mappage vers un magasin de messages afin que les messages puissent être temporairement stockés sur le gestionnaire de files d'attente émetteur en attendant leur transmission.

Files d'attente de stockage et de réacheminement

Les files d'attente de ce type stockent les messages associés à un ou plusieurs gestionnaires de files d'attente cible. Elles ont deux utilisations principales. La première est de permettre le stockage intermédiaire des messages dans un réseau afin que ces messages puissent progresser étape par étape vers leur destination (rôle de réacheminement) ; la seconde est de stocker les messages en attente de collecte (voir aussi Files d'attente de serveur local).

Ces files d'attente sont associées à l'ensemble des noms des gestionnaires de files d'attente pour lesquels elles stockent les messages. Ces derniers sont appelés gestionnaires de files d'attente cible. Les messages adressés à l'un de ces gestionnaires seront alors placés dans la file d'attente de stockage et de réacheminement correspondante. Un nom de gestionnaire de files d'attente de réacheminement peut également être défini pour cette file d'attente. Si tel est le cas, la file d'attente essaie d'envoyer tous ses messages à ce gestionnaire. Sinon, elle stocke les messages.

Ce type de file d'attente est généralement (mais pas forcément) défini sur un serveur ou une passerelle. Plusieurs files d'attente de stockage et de réacheminement peuvent exister sur un gestionnaire de files d'attente, mais les noms de leurs gestionnaires cible doivent tous être différents. Le contenu d'une telle file d'attente n'est pas accessible aux programmes d'application. De même, une application émettrice de messages n'a aucune connaissance de l'existence ou du rôle des files d'attente de stockage et de réacheminement dans la transmission des messages.

Les messages qui se trouvent dans des files d'attente de stockage et de réacheminement ne sont pas accessibles aux applications.

Files d'attente de serveur local

Les files d'attente éloignées et les files de stockage et de réacheminement expédient les messages dans le réseau, les files d'attente émettrices étant à l'origine de la transmission. Les files

d'attente de serveur local, en revanche, permettent d'extraire des messages d'une file d'attente éloignée. Une définition de file d'attente de serveur local identifie une file d'attente de stockage et de réacheminement sur un gestionnaire de files d'attente éloignées. La file d'attente de serveur local extrait les messages destinés à son gestionnaire de files d'attente locales de cette file de stockage et de réacheminement. Plusieurs définitions de files d'attente de serveur local peuvent être définies sur un gestionnaire de files d'attente, chacune d'elles étant associée à une file d'attente de stockage et de réacheminement différente.

Les files d'attente de serveur local résident normalement sur une unité et sont configurées pour extraire les messages d'un serveur chaque fois que l'unité se connecte au réseau. Lorsqu'une file d'attente de serveur local extrait un message du serveur, le message est placé dans la file d'attente locale cible appropriée. Par conséquent, les applications ne peuvent pas accéder aux messages de la file d'attente de serveur local. La méthode qui consiste à extraire les messages du serveur peut être plus efficace en termes de flux réseau que celle qui consiste à faire expédier les messages par le serveur. En effet, la file d'attente de serveur local se sert de l'accusé de réception du premier message comme demande pour le message suivant (s'il y en a un), alors que la méthode d'expédition par le serveur requiert une demande/réponse pour envoyer le message et une seconde demande/réponse pour le flux de confirmation. Un intervalle d'interrogation est normalement défini pour les files d'attente de serveur local, ce qui entraîne une vérification de l'existence de messages en attente sur le serveur tant qu'il est connecté au réseau. La définition de cet intervalle d'interrogation est une tâche administrative qui s'effectue à l'aide d'une option de configuration. Les files d'attente de serveur local contribuent de manière importante à permettre aux clients de recevoir des messages via les canaux client-serveur. En effet, de par la nature des connexions client-serveur, les serveurs ne peuvent être à l'origine d'un transfert de données.⁴

Les messages qui se trouvent dans des files d'attente de serveur local ne sont pas accessibles aux applications.

Files d'attente d'administration

Les files d'attente d'administration constituent le mécanisme par lequel les gestionnaires de files d'attente (et les objets qui y sont associés) sont configurés, localement ou à distance. Un message envoyé à la file d'attente d'administration est traité par la classe de messages d'administration appropriée, puis, facultativement, une

4. La solution alternative consiste à utiliser des canaux entre homologues, ou à doter les deux gestionnaires de files d'attente de fonctions client et serveur.

réponse est envoyée à l'application d'origine. Pour plus de détails à ce sujet, reportez-vous à la section «Administration» à la page 42.

Files d'attente de pont MQSeries

Il s'agit d'une forme spéciale de file d'attente éloignée, avec définition sur une passerelle et file d'attente cible sur un gestionnaire de files d'attente MQSeries. Cette forme de file d'attente fournit un chemin entre les environnements MQSeries Everyplace et MQSeries. Des fonctions Transformation permettent d'effectuer les reformatages de données ou de message éventuellement requis. Une fonction Transformation élémentaire est fournie avec MQSeries Everyplace ; les programmeurs doivent la personnaliser pour l'adapter à leurs besoins propres.

MQSeries Everyplace stocke les données en toute sécurité dans les files d'attente, vérifiant, en fonction de l'adaptateur, que les données sont physiquement inscrites sur le support et non simplement mises en mémoire tampon par le système d'exploitation. Toutefois, MQSeries Everyplace ne consigne pas séparément les modifications aux messages et aux files d'attente. Si vous voulez pouvoir effectuer des restaurations suite à des défaillances du support, vous devez faire appel à des solutions matérielles telles que des systèmes de disques RAID. Alternative possible : mapper la file d'attente sur un espace de stockage restaurable, tel que certains sous-systèmes de base de données.

MQSeries Everyplace n'impose pas que des files d'attente soient définies pour un gestionnaire de files d'attente. Mais, si besoin est, quatre files d'attente système sont prévues :

- **AdminQ** : requise pour la réception des messages administratifs
- **AdminReplyQ** : utilisée en option pour la réception des réponses aux messages administratifs
- **DeadLetter** : utilisée pour stocker les messages qui ne peuvent pas être délivrés
- **SYSTEM.DEFAULT.LOCAL.QUEUE** : file d'attente qui porte le même nom que la file d'attente système obligatoire sur les serveurs MQSeries

Les propriétés des files d'attente sont répertoriées dans le tableau ci-dessous. Notez qu'elles ne sont pas toutes applicables à tous les types de files d'attente.

Tableau 8. Propriétés des files d'attente

Propriété	Explication
Admin_Class	Classe de file d'attente
Admin_Name	Nom de file d'attente ASCII
Queue_Active	Indique que la file d'attente est active

Tableau 8. Propriétés des files d'attente (suite)

Propriété	Explication
Queue_AttRule	Classe de règle contrôlant les opérations de sécurité
Queue_Authenticator	Classe de l'authentificateur
Queue_BridgeName	Nom du pont MQSeries propriétaire
Queue_ClientConnection	Nom de connexion client
Queue_CloseIdle	Ferme le transport lorsque tous les messages ont été transmis
Queue_CreationDate	Date de création de la file d'attente
Queue_Compressor	Classe du module de compression
Queue_Cryptor	Classe du module de chiffrement
Queue_CurrentSize	Nombre de messages en file d'attente
Queue_Description	UNICODE description
Queue_Expiry	Heure d'expiration des messages
Queue_FileDesc	Emplacement et adaptateur de la file d'attente
Queue_MaxMsgSize	Longueur maximale des messages autorisée dans la file d'attente
Queue_MaxQSize	Nombre maximal de messages autorisés
Queue_Mode	Synchrone ou asynchrone
Queue_MQQMgr	Proxy du gestionnaire de files d'attente MQSeries
Queue_Priority	Priorité à utiliser pour les messages (sauf si une autre valeur est spécifiée pour le message)
Queue_QAliasNameList	Autres noms de la file d'attente
Queue_QMgrName	Gestionnaire de files d'attente propriétaire de la file d'attente réelle
Queue_QMgrNameList	Cibles du gestionnaire de files d'attente
Queue_RemoteQName	Nom de la zone MQSeries éloignée
Queue_Rule	Classe de règle pour les opérations sur files d'attente
Queue_QTimerInterval	Délai de traitement des messages en attente
Queue_TargetRegistry	Type de registre cible
Queue_Transporter	Classe du transport
Queue_TransporterXOR	Transport devant utiliser la compression XOR
Queue_Transformer	Classe de la fonction Transformation

Les fonctions administratives servent à créer et supprimer des files d'attente, et à consulter ou modifier leurs propriétés.

Les écrans MQe_Explorer ci-dessous montrent deux des quatre onglets contenant les propriétés d'une file d'attente locale. Les zones désactivées indiquent que les propriétés correspondantes ne sont pas applicables à ce type de files d'attente.

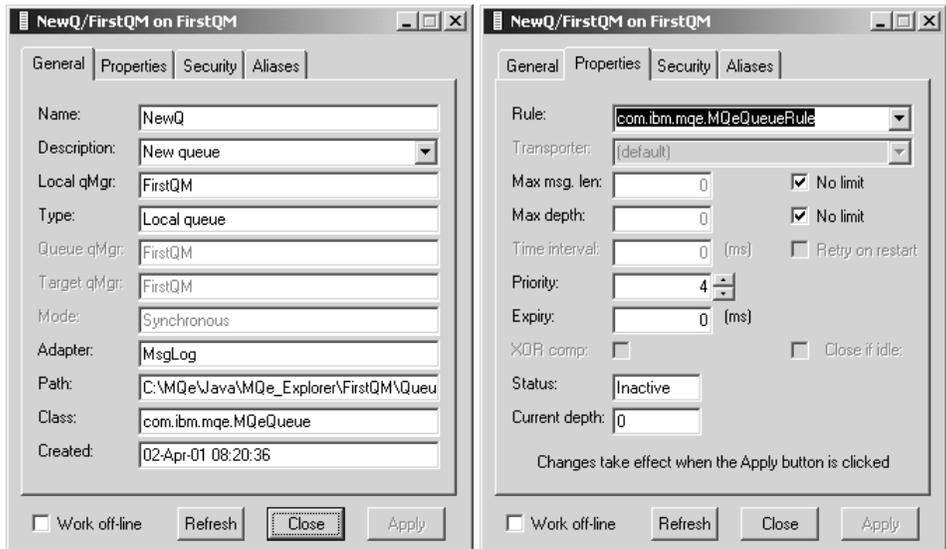


Figure 5. Propriétés d'une file d'attente locale affichées à l'aide de MQe_Explorer

Les files d'attente ne sont pas simplement des espaces de stockage des messages. Des files d'attente transformées en sous-classes peuvent être utilisées dans des scénarios d'applications de contrôle de processus. Par exemple, un objet file d'attente peut contrôler directement une soupape. Un message de la classe appropriée entraîne alors l'ouverture d'une soupape, le changement du volume du flux, etc. L'application n'a plus à extraire les messages de la file d'attente et à exécuter l'action, c'est l'objet file d'attente lui-même qui contrôle l'action. D'autres files d'attente peuvent, par exemple, mettre à jour des feuilles de calcul ou convertir du texte en données vocales. Cette technique a l'avantage de préserver les garanties de sécurité des files d'attente tout comme la garantie de livraison. MQSeries Everyplace assure alors la livraison garantie unique des messages et, associé à un authentificateur et à un module de chiffrement, garantit que seul l'expéditeur habilité du message peut envoyer ce type de messages, le contenu étant hautement sécurisé pendant le transit. Aucune application n'est autorisée à accéder à la file d'attente, aucun accès n'étant par ailleurs requis.

Gestionnaires de files d'attente

Le gestionnaire de files d'attente MQSeries Everyplace donne à l'application accès aux messages et aux files d'attente, et contrôle les éventuels canaux. Dans MQSeries Everyplace version 1.2, un seul gestionnaire de files d'attente peut être actif sur une machine virtuelle Java (JVM) à un moment donné. S'il existe plusieurs JVM sur une machine, il peut y avoir autant de gestionnaires de files d'attente que de JVM. Les gestionnaires de files d'attente sont identifiés par leur nom, qui doit être globalement unique⁵ et par une chaîne de caractères ASCII de taille non limitée. ⁶

Un gestionnaire de files d'attente configuré avec des fonctions de pont, et qui peut donc échanger des messages avec des produits hôte et distribués MQSeries, est appelé passerelle.

Les gestionnaires de files d'attente peuvent être configurés avec ou sans mise en file d'attente locale. Tous les gestionnaires de file d'attente prennent en charge les opérations de messagerie synchrones ; un gestionnaire de files d'attente avec mise en file d'attente locale prend également en charge la livraison des messages en mode asynchrone.

Le choix d'une livraison synchrone ou asynchrone des messages est déterminé par la nature des définitions de files d'attente sur le gestionnaire de files d'attente émetteur. Si une définition de file d'attente éloignée fonctionnant en mode synchrone existe pour la file d'attente cible, la livraison s'effectue en mode synchrone. Si une définition de file d'attente éloignée fonctionnant en mode asynchrone existe, la livraison s'effectue dans ce mode, la définition fournissant le stockage local en attendant la transmission des messages. Si aucune définition de file d'attente éloignée n'existe mais qu'il existe une file d'attente de stockage et de réacheminement chargée de traiter les messages pour ce gestionnaire de files d'attente cible, la livraison s'effectue en mode asynchrone. Dans ce cas, la file d'attente de stockage et de réacheminement fournit le stockage local pour les messages en attente de transmission. S'il n'existe aucune définition de file d'attente, une reconnaissance des files d'attente a lieu, qui, si elle aboutit, a pour effet une livraison synchrone des messages.

Quel que soit le mode de livraison utilisé, MQSeries Everyplace peut procéder à une transmission directe ou indirecte, en fonction des définitions de

5. Cette restriction n'est pas appliquée par MQSeries Everyplace ou MQSeries, mais l'existence de noms de gestionnaire de files d'attente en double peut entraîner la livraison des messages à un gestionnaire de files d'attente incorrect.

6. Pour une meilleure interopérabilité il est conseillé toutefois de respecter les conventions de dénomination MQSeries, et de limiter les noms à 48 caractères. Le système de fichiers utilisé peut également avoir une incidence sur la longueur.

connexions disponibles (voir la section «Connexions» à la page 41). La transmission directe n'implique que deux gestionnaire de files d'attente : le gestionnaire émetteur et le gestionnaire cible. La transmission indirecte implique une succession de gestionnaires de files d'attente, le protocole utilisé pouvant changer en cours de route. Lorsqu'une transmission indirecte est utilisée en mode synchrone, la livraison des messages s'effectue de la même manière que lors d'une transmission directe. Les gestionnaires de files d'attente intermédiaires ne font qu'établir une connexion entre la source et la cible. En revanche, lorsqu'une transmission indirecte est utilisée en mode asynchrone, les intermédiaires de transmission peuvent devenir des étapes de transfert pour les messages sur leur chemin de la source vers la cible. Cela dépend de la création préalable d'un espace de stockage de file d'attente intermédiaire approprié sur les gestionnaires de files d'attente intermédiaires, sous la forme de définitions appropriées de files d'attente éloignées ou de files d'attente de stockage et de réacheminement.

Les caractéristiques et les conséquences de la livraison des messages en mode asynchrone ou synchrone sont très différentes : Avec la livraison asynchrone des messages, l'application transmet le message à MQSeries Everyplace pour qu'il le livre à une file d'attente éloignée. L'application est immédiatement notifiée. Si le message peut être livré immédiatement (ou transféré vers une étape de transfert appropriée), il est envoyé ; sinon, il est stocké localement. La logique qui contrôle les tentatives de transmission est définie par les règles associées au gestionnaire de files d'attente et aux files d'attente concernées (voir la section «Règles» à la page 59). La livraison asynchrone offre une qualité de service de type livraison garantie unique des messages, puisque les messages ont été transmis à MQSeries Everyplace et que ce dernier est devenu responsable de leur livraison.

Avec la livraison synchrone des messages, l'application transmet le message à MQSeries Everyplace pour qu'il le livre à une file d'attente éloignée. MQSeries Everyplace contacte en mode synchrone la file d'attente cible et y place le message. La livraison effectuée, MQSeries Everyplace notifie l'application. L'application est immédiatement notifiée. Si le message ne peut être livré, l'application en est immédiatement notifiée. MQSeries n'est pas responsable de la livraison des messages lorsque celle-ci s'effectue en mode synchrone.

Dans un réseau MQSeries Everyplace, la création de définitions de files d'attente éloignées synchrones et asynchrones n'est limitée par aucune contrainte. En revanche, lorsque le réseau comporte également des gestionnaires de files d'attente MQSeries (et leurs files d'attente), les restrictions importantes suivantes s'appliquent :

1. La livraison synchrone n'est pas possible vers un gestionnaire de files d'attente MQSeries qui n'est pas connecté directement à une passerelle MQSeries Everyplace (en effet, la livraison synchrone n'est pas prise en

gestionnaires de files d'attente

charge par les canaux de messages MQSeries). Dans ce cas, afin de minimiser les conséquences de cette restriction, la définition de la livraison synchrone est modifiée et limitée à la livraison au gestionnaire de files d'attente MQSeries directement connectée à la passerelle. Au delà de ce gestionnaire de files d'attente, la livraison asynchrone MQSeries est utilisée pour permettre au message d'arriver à destination.

2. Dans MQSeries Everyplace version 1.2, seule une définition de file d'attente éloignée synchrone peut être créée pour une file d'attente éloignée de messagerie MQSeries. Compte tenu de la première restriction énoncée ci-dessus, cela signifie que la livraison asynchrone est impossible (à l'aide de cette définition) dans les files d'attente situées sur un gestionnaire de files d'attente MQSeries directement connecté à la passerelle. Toutefois, si une mise en file d'attente est requise dans ce cas, elle peut être mise en œuvre dans le réseau MQSeries Everyplace en utilisant un second gestionnaire de files d'attente MQSeries Everyplace, sur lequel seront créées les définitions de files d'attente éloignées (ou de files d'attente de stockage et de réacheminement) appropriées.

La livraison asynchrone de messages signifie donc que l'application locale transmet le message à MQSeries Everyplace et que ce dernier est responsable de sa transmission à partir du gestionnaire de files d'attente locales. Par conséquent, il n'est pas nécessaire que le réseau et/ou l'application réceptrice soient disponibles. Le moment de livraison effective du message est inconnu de l'application expéditrice. La livraison synchrone d'un message requiert que le réseau soit actif, mais l'application expéditrice est informée de sa transmission à la file d'attente de l'application destinataire. Dans les deux cas (transmission asynchrone ou synchrone), la disponibilité de l'application réceptrice n'est pas requise.

Pour la livraison asynchrone, MQSeries Everyplace n'offre aucune variante de la qualité de service de type livraison garantie unique, contrairement à MQSeries, qui offre les options de persistance et de non-persistance. Dans MQSeries Everyplace, le choix entre fiabilité et performance s'effectue au niveau des files d'attente, en sélectionnant l'adaptateur de stockage approprié. Par exemple :

- L'adaptateur MQeDiskFieldsAdapter garantit l'écriture des données sur disque avant la poursuite du traitement.
- L'adaptateur MQeReducedDiskFieldsAdapter garantit la transmission des données au système d'exploitation (et la possibilité de leur restauration) avant la poursuite du traitement.
- L'adaptateur MQeMemoryFieldsAdapter sauvegarde les données en mémoire.

Dans tous les cas, la livraison garantie unique est mise en œuvre, mais la reprise dépend du type de magasin de messages choisi. Les compromis en termes de performances sont décrits dans le document SupportPac EP01.

Les propriétés des gestionnaires de files d'attente sont répertoriées dans le tableau 9.

Tableau 9. Propriétés d'un gestionnaire de files d'attente locales

Propriété	Description
Admin_Class	Classe du gestionnaire de files d'attente
Admin_Name	Nom du gestionnaire de files d'attente
QMgr_ChnlAttrRules	Règles d'attribut de canal
QMgr_ChnlTimeout	Délai d'expiration du canal
QMgr_Description	UNICODE description
QMgr_QueueStore	Emplacement et adaptateur par défaut des files d'attente
QMgr_Rules	Classe de règle pour les opérations du gestionnaire de files d'attente

L'écran MQE_Explorer suivant montre l'un des deux onglets contenant les propriétés d'une file d'attente locale :

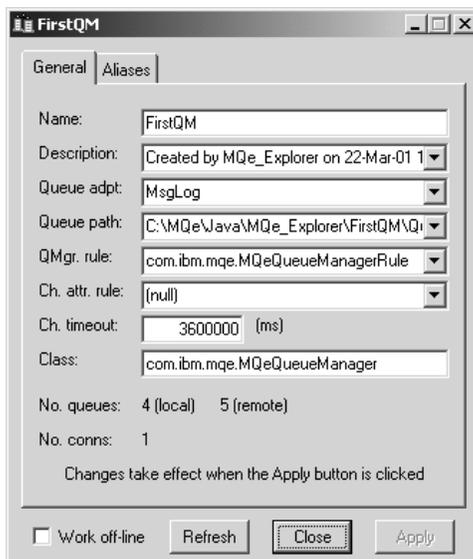


Figure 6. Propriétés d'un gestionnaire de files d'attente affichées par MQE_Explorer

Configuration d'un gestionnaire de files d'attente

Un gestionnaire de files d'attente s'exécute dans un environnement établi par MQSeries Everyplace avant le chargement de ce gestionnaire. Ce dernier stocke ses informations de configuration dans son registre (le registre est décrit à la section «Registre» à la page 55). Les files d'attente proprement dites (contenant les messages) sont stockées dans des magasins de files d'attente.

L'environnement MQSeries Everyplace peut être établi de plusieurs manières : par le biais d'appels effectués via l'API, au moyen d'utilitaires fournis avec MQSeries Everyplace, ou à l'aide d'outils de gestion tels que MQe_Explorer. Un grand nombre de ces outils capturent les paramètres de l'environnement dans un fichier d'initialisation, mais cette capture est facultative. Les paramètres d'environnement clés sont répertoriés dans le tableau ci-dessous. Les noms de sections utilisés respectent les conventions de représentation des informations dans les fichiers d'initialisation.

Tableau 10.

Nom de la section	Propriété	Explication
[Alias]	(définitions d'alias de classes)	Un alias peut être utilisé chaque fois qu'un nom de classe est requis.
[ChannelManager]	MaxChannels	Nombre maximum de canaux client-serveur simultanés à autoriser.
[Listener]	Network	Adaptateur et numéro de port à écouter pour détecter les demandes de connexion de canaux client-serveur entrantes.
	Listen	Adaptateur à utiliser pour gérer les connexions résultantes.
	TimeInterval	Délai d'attente du canal client-serveur.
[MQBridge]	(paramètres d'initialisation du pont)	
[MQe_Explorer]	(informations d'adressage sauvegardées)	MQe_Explorer sauvegarde les informations qui décrivent comment les connexions entrantes peuvent être établies avec ce gestionnaire de files d'attente.

Tableau 10. (suite)

Nom de la section	Propriété	Explication
[Permission]	(commandes autorisées)	Commandes de canal autorisées, classe adaptateur et mappes de descripteurs de fichiers.
[PreLoad]	classes à charger lors de l'initialisation du gestionnaire de files d'attente	Fournit un mécanisme de chargement des classes d'application.
[QueueManager]	Nom	Nom du gestionnaire de files d'attente.
[Registry]	DirName	Emplacement du registre.
	LocalRegType	Type du registre (fichier ou privé).
	PIN	PIN du registre (ou demande d'invite).
	CertReqPIN	PIN du certificat (ou demande d'invite).
	KeyRingPassword	Mot de passe du fichier de clés (ou demande d'invite).
	CAIPAddrPort	Adresse IP de l'autorité d'accréditation.

Les quatre illustrations ci-après représentent les principaux onglets à utiliser lors de la création d'un gestionnaire de files d'attente à l'aide de MQe_Explorer. Cette création ne requiert aucun environnement préalable et aucun fichier d'initialisation n'est nécessaire. Au terme de la procédure, un gestionnaire de files d'attente opérationnel est créé, et les données de configuration sont sauvegardées dans un fichier d'initialisation afin que ce gestionnaire puisse être redémarré par simple lecture de ce fichier.

gestionnaires de files d'attente

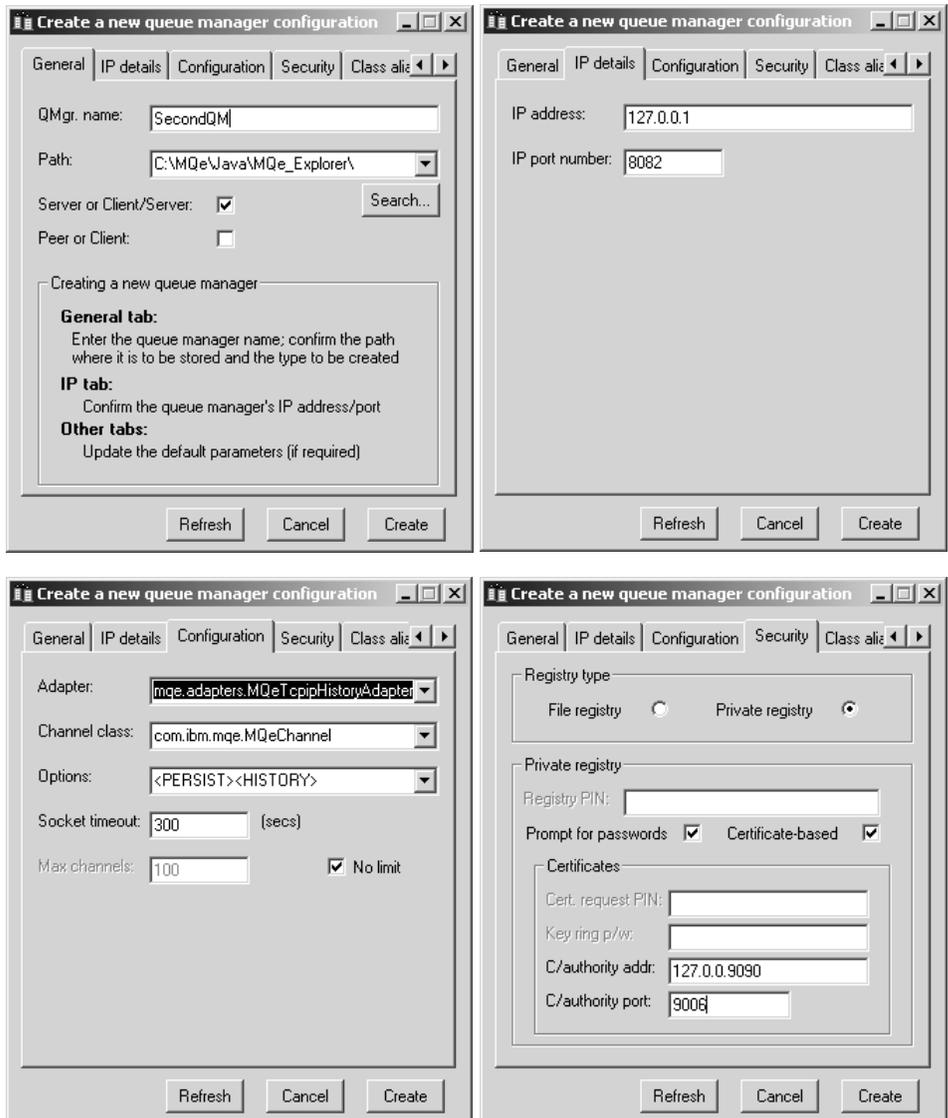


Figure 7. Création d'un gestionnaire de files d'attente à l'aide de MQe_Explorer

Pour créer un gestionnaire de files d'attente, il n'est pas nécessaire de connaître son adresse IP. Cependant, MQe_Explorer capture cette information, dont il a besoin pour configurer les autres gestionnaires de files d'attente qui seront appelés à dialoguer avec ce gestionnaire. Le numéro de port est nécessaire pour permettre l'écoute des demandes de connexions client-serveur entrantes.

Le programme d'écoute de canal a besoin des données de configuration de l'adaptateur, tandis que le gestionnaire de canal a besoin de l'intervalle de temps et du nombre maximum de canaux. MQE_Explorer capture les autres informations comme indiqué précédemment, afin de permettre la configuration d'autres gestionnaires de files d'attente.

Dans cet exemple, un registre sécurisé a été configuré avec une authentification par certificat (voir la section «Sécurité» à la page 50). Les PIN (Numéro d'Identification Personnel) et les mots de passe sont désactivés car MQE_Explorer les demande ultérieurement, lorsqu'ils sont nécessaires.

Opérations des gestionnaires de files d'attente

Les gestionnaires de files d'attente gèrent les opérations de messagerie et, éventuellement, les files d'attente. Les applications accèdent aux messages en utilisant les services du gestionnaire de files d'attente, à l'aide de méthodes telle que *get*, *put*, *browse*, *wait*, *listen* et *delete*. Un grand nombre de ces opérations comportent un *filtre* parmi leurs paramètres. Un filtre est un objet zones qui sert de base de comparaison, toutes les zones d'un message pouvant être utilisées pour une extraction sélective. La plupart des appels de méthodes peuvent également comporter un objet attributs, utilisable pour le codage et le décodage d'un message.

L'opération d'extraction (*get*) supprime les messages d'une file d'attente de façon destructive. Selon les conditions imposées par le filtre, les messages sont extraits par ordre de priorité et d'arrivée dans la file d'attente. Par conséquent, à conditions égales, le premier message placé dans la file sera le premier extrait. L'opération d'extraction peut être effectuée en une ou deux étapes. La version en deux étapes est prévue pour les situations dans lesquelles il est impératif que le message ne puisse pas être perdu lors de son transfert entre MQSeries Everyplace et l'application. Dans un premier temps, une demande d'extraction (*get*) est émise avec un *ID de confirmation* (dont la valeur est définie par l'application). Cette opération extrait le message mais, au lieu de le supprimer immédiatement de la file d'attente, elle le masque dans celle-ci. Une opération de confirmation, spécifiant l'UID message d'origine, indique ensuite que l'opération d'extraction a abouti pour l'application et le message est alors supprimé. Si l'opération d'extraction échoue, le message peut ainsi être restauré. Le comportement des opérations d'insertion (*put*) est similaire.

il est possible, en spécifiant l'UID correspondant, de *supprimer* un message d'une file d'attente sans l'extraire.

Si une lecture non destructive est requise, il est possible de *parcourir* les files d'attente à la recherche de messages (éventuellement sous contrôle d'un filtre). Cette exploration extrait tous les objets messages correspondant au filtre, mais les laisse dans la file d'attente. L'*exploration sous verrou* est également prise en charge. Cette fonction verrouille les messages qui correspondent au filtre dans

gestionnaires de files d'attente

la file d'attente. Le verrouillage peut être individuel ou groupé à l'aide d'un filtre. Cette opération renvoie un *ID de verrou*. Les messages verrouillés ne peuvent être récupérés ou supprimés que si l'ID de verrou est fourni. Une option d'exploration permet de spécifier si l'intégralité des messages ou seuls les UID doivent être renvoyés.

Les applications peuvent *attendre*, pendant un laps de temps défini, l'arrivée des messages dans une file d'attente. Il est également possible de définir un filtre pour identifier les messages intéressants, ou encore de spécifier un *ID de confirmation*. Autre possibilité : les applications peuvent écouter les événements relatifs aux message MQSeries Everyplace, là encore éventuellement avec un filtre. Les programmes d'écoute sont avertis de l'arrivée des messages dans les files d'attente.

Le tableau 11 indique les opérations possibles sur les différents types de files d'attente.

Tableau 11. Opérations de messagerie sur les files d'attente MQSeries Everyplace

	File d'attente locale	File d'attente éloignée ¹	
		Synchrone	Asynchrone
Browse (Parcourir) (±verrou, ±filtre)	Oui	Oui	
Delete (Supprimer)	Oui	Oui	
Get (Extraire) (±filtre)	Oui	Oui	
Listen (Écouter) (±filtre)	Oui		
Put (Insérer)	Oui	Oui	Oui
Wait (Attendre) (±filtre)	Oui	Oui	
Remarques :			
1. L'opération d'attente à distance synchrone est mise en œuvre via une interrogation de la file d'attente éloignée, ce qui fait que le délai d'attente réel est un multiple du délai d'interrogation.			
2. ¹ Le pont MQSeries Everyplace MQSeries fourni avec MQSeries Everyplace version 1.2 ne prend en charge que les opérations put garanties/non garanties, get non garanties, et browse non garanties (sans verrou).			

Connexions

La topologie et l'accès via le réseau MQSeries Everyplace sont définis par des objets connexions. Ces définitions sont stockées localement sur chaque gestionnaire de files d'attente. Elles sont créées, modifiées et détruites selon les procédures administratives standard.

Un objet connexion définit généralement l'accès à un gestionnaire de files d'attente éloignées (par conséquent, cet objet est parfois appelé définition de gestionnaire de files d'attente éloignées). Les propriétés de ce type d'objet sont répertoriées dans le tableau suivant :

Tableau 12. Propriétés de connexion (gestionnaire de files d'attente éloignées)

Propriété	Explication
Admin_Name	Nom du gestionnaire de files d'attente
Con_Adapter	Descripteur du fichier de l'adaptateur
Con_AdapterOptions	Options de l'adaptateur (utilisation d'un historique, par exemple)
Con_AdapterParm	Données ASCII à utiliser par un adaptateur (nom de servlet, par exemple)
Con_Aliases	Autres noms du gestionnaire de files d'attente/des connexions
Con_Channel	Type de canal à utiliser par la connexion
Con_Description	Description UNICODE
Queue_QMgrName	Propriétaire de la définition

Les captures d'écrans d'administration suivantes représentent deux des onglets de propriétés associés à une définition de connexion :

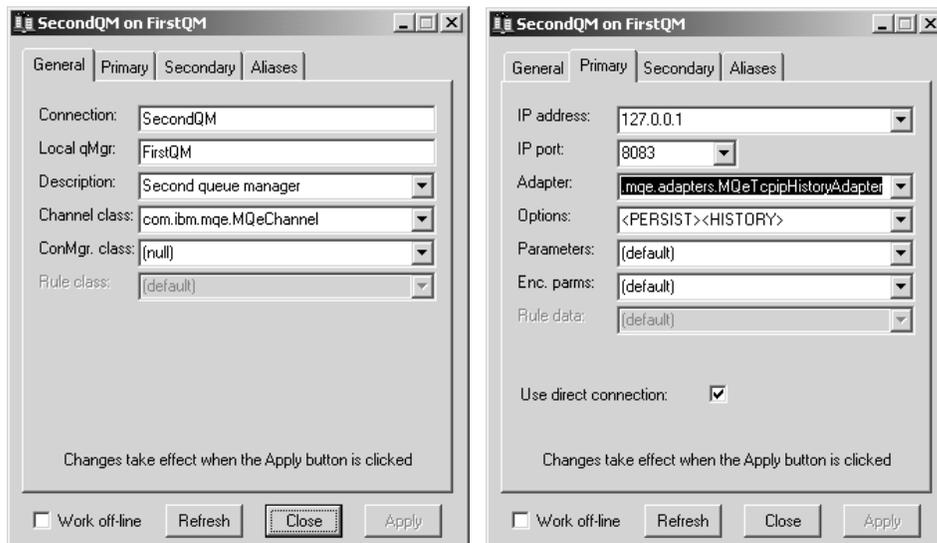


Figure 8. Propriétés d'une définition de connexion affichées à l'aide de MQe_Explorer

Les données peuvent être transmises à l'adaptateur de communication choisi sous la forme d'options, de paramètres et de paramètres codés.

Des connexions indirectes peuvent être indiquées. Dans ce cas, MQSeries Everyplace achemine la connexion via d'autres gestionnaires de files d'attente (qui peuvent être chaînés), et le protocole peut changer en cours de route. Les connexions indirectes sont particulièrement utiles pour permettre à des unités de disposer d'un point d'entrée unique sur un réseau MQSeries Everyplace.

Les objets connexions servent également à définir des programmes d'écoute pour les canaux entre homologues entrants.

Comme pour la plupart des objets MQSeries, des alias peuvent être définis pour les connexions. Une connexion locale, c'est-à-dire dont le nom correspond à celui du gestionnaire de files d'attente locales, est utilisée pour définir les alias de ce gestionnaire.

Administration

L'administration fournit des fonctions permettant de configurer et de gérer les ressources MQSeries Everyplace, telles que les files d'attente et les connexions. Les fonctions relatives aux messages sont de la responsabilité des applications. L'administration est activée via une interface qui gère la génération et la

réception des messages administratifs, conçue de sorte que l'administration locale et éloignée soit gérée de la même manière. Les demandes sont envoyées à la file d'attente d'administration du gestionnaire de files d'attente cible et les réponses peuvent être réceptionnées si nécessaire. Tout programme d'application MQSeries Everyplace local ou éloigné peut créer et traiter les messages d'administration, directement ou indirectement, à l'aide de méthodes auxiliaires (helper). Les messages d'administration peuvent également être générés indirectement via MQe_Explorer⁷, un outil de gestion qui fournit une interface graphique d'administration du système.

La file d'attente d'administration ne "sait" pas administrer les ressources individuelles. Cette connaissance est encapsulée dans chaque ressource et dans le message d'administration correspondant.

Messages d'administration

Les messages d'administration étendent l'objet message MQSeries Everyplace de base. Le tableau 13 recense les classes de messages fournies pour l'administration des ressources MQSeries Everyplace. Ces messages d'administration de base peuvent être définis comme sous-classes pour fournir l'administration d'autres objets ; par exemple, une file d'attente de type différent peut être gérée via une sous-classe de MQeQueueAdminMsg. Le pont MQSeries Everyplace vers MQSeries utilise des sous-classes de MQeAdminMsg de la manière suivante :

Tableau 13. Classes de messages d'administration

Classe de messages d'administration	Utilisation
MQeAdminMsg	Classe abstraite utilisée comme base de tous les messages d'administration
MQeQueueManagerAdminMsg	Administration des gestionnaires de files d'attente
MQeQueueAdminMsg	Administration des files d'attente locales
MQeRemoteQueueAdminMsg	Administration des files d'attente éloignées
MQeAdminQueueAdminMsg	Administration de la file d'attente d'administration
MQeHomeServerQueueAdminMsg	Administration des files d'attente du serveur local
MQeStoreAndForwardQueueAdminMsg	Administration des files d'attente de stockage et réacheminement
MQeConnectionAdminMsg	Administration des connexions entre gestionnaires de files d'attente

7. MQe_Explorer n'est pas fourni avec MQSeries Everyplace mais peut être téléchargé gratuitement à partir du site Web de MQSeries Everyplace (<http://www.ibm.com/software/ts/mqseries/everyplace>).

Tableau 13. Classes de messages d'administration (suite)

Classe de messages d'administration	Utilisation
MQeClientConnectionAdminMsg	Administration d'un objet connexion de client de pont, utilisé pour la connexion à MQSeries
MQeListenerAdminMSg	Administration d'un objet programme d'écoute de file d'attente de transmission de pont, utilisé pour collecter les messages à partir de MQSeries
MQeBridgeAdminMsg	Administration d'un pont vers MQSeries
MQeMQBridgesAdminMsg	Administration d'une liste de ponts MQSeries
MQeMQQMgrProxyAdminMsg	Administration de la représentation d'un pont d'un gestionnaire de files d'attente MQSeries
MQeMQBridgeQueueAdminMsg	Administration d'une file d'attente d'un pont MQSeries

La structure d'un message administratif dépend de sa classe spécifique, c'est-à-dire de la nature de la ressource qu'il gère et des détails de l'opération à effectuer sur cette ressource. Génériquement toutefois, les messages administratifs sont structurés comme indiqué au tableau 14.

Tableau 14. Structure générique d'un message administratif

Zones de niveau 1	Zones de niveau 2 et inférieurs	Utilisation
Admin_Action		Création, suppression, interrogation, etc.
Admin_Errors		Parent de l'objet zones
	Plusieurs zones	Informations détaillées, erreur par erreur
Admin_MaxAttempts		Nombre maximal de tentatives d'exécution de l'action administrative
Admin_Parameters		Parent de l'objet zones
	Ressource	Nom de la ressource à gérer
	Plusieurs zones	Données détaillées sur le paramètre spécifique de la classe du message et de l'action

Tableau 14. Structure générique d'un message administratif (suite)

Zones de niveau 1	Zones de niveau 2 et inférieurs	Utilisation
Admin_Reason		Message texte indiquant la raison de l'échec
Msg_ReplyToQ		Nom de la file d'attente à laquelle la réponse doit être envoyée
Msg_ReplyToQMgr		Nom du gestionnaire de files d'attente auquel la réponse doit être envoyée
Admin_RC		Code retour numérique indiquant le résultat
Msg_Style		Commande ou demande/réponse
Admin_TargetQMgr		Nom du gestionnaire de files d'attente propriétaire de la ressource cible

Il existe trois styles de messages administratifs, à savoir les commandes (datagrammes) qui indiquent une action administrative ne nécessitant pas de réponse, les demandes qui requièrent une réponse, et les réponses elles-mêmes. La réponse est construite à partir d'une copie du message original ; ainsi, l'expéditeur peut ajouter des zones utilisables par le destinataire.

Administration sélective

L'accès à l'administration peut être contrôlée via l'authentificateur sur la file d'attente d'administration. L'authentificateur fourni considère toutes les applications locales comme représentant le même utilisateur local et, de ce fait, leur octroie ou leur refuse à toutes l'accès à l'administration. Les applications d'administration éloignées sont contrôlées par l'appel de l'authentificateur sur le canal avant tout flux de messages d'administration. Différents utilisateurs éloignés peuvent ainsi être distingués et agréés ou non séparément. Dans tous les cas, pour tout utilisateur, l'accès à l'administration est octroyé ou refusé dans sa totalité. Pour obtenir un niveau de contrôle plus précis (par exemple, pour accorder à certains utilisateurs l'accès à certaines files d'attente), il faut le programmer. Un authentificateur plus sophistiqué peut garder trace des autorisations en les associant aux identités des utilisateurs, les messages d'administration pouvant alors être traités sur la base de ces autorisations (reportez-vous à la section «Sécurité» à la page 50). Les règles associées aux files d'attente peuvent également être exploitées pour autoriser ou interdire des actions de manière similaire (reportez-vous à la section «Règles» à la page 59).

Surveillance et actions connexes

L'administration recouvre souvent davantage que la création et la modification d'objets. En effet, il peut être nécessaire de surveiller un système et d'informer un opérateur qu'une file d'attente est saturée, ou de réagir suite à une erreur, par exemple lorsqu'un message entrant est trop grand pour la file d'attente cible. Tous ces aspects sont gérés dans MQSeries Everyplace par l'intermédiaire de règles, c'est-à-dire de classes qui sont appelées chaque fois que l'état des objets change de façon significative ou que se produisent certaines erreurs. Un jeu de classes de règles par défaut est fourni avec MQSeries Everyplace, mais ces classes sont généralement remplacées par des classes personnalisées (reportez-vous à la section «Règles» à la page 59).

Canaux dynamiques

MQSeries Everyplace établit la communication entre les gestionnaires de files d'attente via des liaisons logiques appelées canaux dynamiques. Ces canaux prennent en charge les flux bidirectionnels et sont établis à la demande par le gestionnaire de files d'attente. Asynchrone ou synchrone, la messagerie utilise les mêmes canaux, et le protocole exploité est unique dans MQSeries Everyplace. Ce protocole peut être personnalisé pour chaque message en remplaçant la méthode de vidage du message. En revanche, MQSeries utilise généralement des canaux client pour le trafic synchrone et une paire de canaux de messages pour la messagerie asynchrone bidirectionnelle. Quelques-unes des caractéristiques des *canaux de message de cluster* MQSeries sont semblables à celles des canaux dynamiques MQSeries Everyplace, mais il existe de nombreuses différences importantes.

Un canal dynamique est une connexion logique entre deux gestionnaires de files d'attente, établie dans le but d'envoyer ou de recevoir des données. Plusieurs canaux concurrents peuvent coexister, y compris entre les mêmes éléments. Des caractéristiques leur sont associées : authentification, chiffrement, compression, protocole de transport utilisé, etc. Ces caractéristiques sont des fonctions complémentaires (des versions différentes peuvent être utilisées sur les différents canaux) et, par conséquent, chaque canal possède ses propres attributs de qualité de service :

- **Authentificateur** : NULL ou objet *authentificateur* capable d'authentifier un canal ou un utilisateur.
- **Canal** : classe fournissant les services de transport.
- **Compresseur** : NULL ou objet *module de compression* capable d'effectuer la compression et la décompression des données.
- **Module de chiffrement** : NULL ou objet *module de chiffrement* capable d'effectuer le chiffrement et le déchiffrement.
- **Destination** : cible de ce canal, par exemple SERVER.XYZ.COM

L'authentificateur n'est généralement utilisé que lors de la configuration du canal. Les modules de compression et de chiffrement sont généralement utilisés sur tous les flux.

Le type de module de chiffrement le plus simple est MQeXorCryptor, qui chiffre les données envoyées en effectuant un OU exclusif (XOR) sur les données. Ce mode de chiffrement n'est pas sécurisé, mais il modifie les données afin qu'elles ne puissent pas être visualisées. En revanche, MQe3DESCryptor met en œuvre un triple DES. Le type le plus simple de compresseur est MQeRleCompressor, qui compresse les données en remplaçant les caractères répétitifs par un compteur. D'autres authentificateurs, modules de compression et modules de chiffrement sont fournis : reportez-vous au tableau 15 à la page 51.

L'établissement d'un canal fait appel aux spécifications de l'adaptateur du protocole pour déterminer les liaisons et les protocoles à utiliser. À chaque nœud intermédiaire, les définitions du canal sont explorées pour répondre aux besoins d'adressage requis par la liaison suivante. Lorsqu'il n'existe plus de définition suivante, le canal est arrêté et tout message en transit est transféré vers le gestionnaire de files d'attente. De cette manière, il est possible d'exploiter les files d'attente de stockage et de réacheminement et les définitions de files d'attente éloignées intermédiaires.

Les canaux ne sont pas directement visibles par les applications ou les administrateurs, et sont établis à la demande par le gestionnaire de files d'attente. Ils relient les gestionnaires de files d'attente entre eux et leurs caractéristiques sont négociées et renégociées par MQSeries Everyplace en fonction des informations à transférer. Les transporteurs sont les composants MQSeries Everyplace qui exploitent les canaux pour fournir les communications au niveau file d'attente. Là encore, ces canaux ne sont pas visibles par le programmeur de l'application ou l'administrateur.

Quand la livraison garantie est impérative, MQSeries Everyplace livre les messages à l'application une fois et une seule. Pour ce faire, il vérifie qu'un message a bien été transmis d'un gestionnaire de files d'attente à un autre, avec accusé de réception, avant de supprimer la copie à la fin de la transmission. En cas d'échec de la communication (accusé de réception non reçu), un message peut être retransmis (une seule livraison ne signifie pas une seule transmission), mais il n'est pas livré deux fois.

Adaptateurs

Les *adaptateurs* permettent de mapper MQSeries Everyplace vers les interfaces des unités. Les canaux exploitent les adaptateurs du protocole pour s'exécuter sous HTTP, TCP/IP natif, UDP et d'autres protocoles. De même, les files d'attente exploitent les adaptateurs de stockage de zones pour établir

canaux dynamiques

l'interface avec un sous-système de stockage tel que la mémoire ou le système de fichiers. Les adaptateurs fournissent un mécanisme permettant à MQSeries Everyplace à la fois d'étendre son support des unités et d'autoriser la gestion des versions.

Un *descripteur de fichier* est une chaîne utilisée pour identifier, charger et activer un adaptateur.

Gestion de connexion commutée

La prise en charge de l'accès réseau à distance pour les unités est géré par leur système d'exploitation. Lorsque MQSeries Everyplace tente, sur une unité déconnectée, d'utiliser le réseau (pour envoyer un message par exemple) et que la pile réseau n'est pas active, le système d'exploitation démarre lui-même les services RAS (remote access services). Généralement, cette opération se matérialise par l'affichage d'un panneau proposant un profil de connexion commutée. Le système d'exploitation conserve le contrôle jusqu'à l'établissement de la connexion. Par conséquent, l'utilisateur de l'unité doit vérifier que les profils de connexion commutée appropriés sont à la disposition du système d'exploitation. MQSeries Everyplace version 1.2 n'offre aucun support explicite pour l'accès réseau à distance.

Trace

La trace est activée par un programme indépendant. Des appels sont intégrés à MQSeries Everyplace pour effectuer la trace des informations, des avertissements et des erreurs, avec les variantes du système et des utilisateurs. Les applications peuvent également appeler directement la fonction de trace et ajouter des messages de trace ou les modifier. Le programme de trace exemple fourni permet d'afficher les messages sélectionnés, de les imprimer et/ou de les transférer dans le journal des événements. D'autres programmes de trace peuvent être écrits, dotés de fonctions supplémentaires ou conçus pour formater et livrer les résultats de manière différente.

La majorité des erreurs MQSeries Everyplace sont transmises à l'application pour traitement et le gestionnaire d'exceptions de l'application peut également les transférer dans le journal de trace.

Journal des événements

MQSeries Everyplace fournit des mécanismes de consignation des événements et des interfaces, utilisables pour consigner les états, le démarrage du gestionnaire de files d'attente, par exemple. La consignation peut être lancée et effectuée par défaut dans un fichier, mais il est possible de réacheminer la

sortie. Le journal des événements MQSeries Everyplace ne consigne pas les données des messages et ne peut pas être utilisé pour restaurer des messages ou des files d'attente.

Livraison des messages

Les réseaux MQSeries Everyplace sont constitués de gestionnaires de files d'attente interconnectés et éventuellement de passerelles. Ils peuvent s'étendre sur plusieurs réseaux physiques et acheminer des messages de l'un à l'autre. Ils fournissent généralement un accès synchrone et asynchrone aux files d'attente avec un modèle de programmation indépendant de l'emplacement de la file d'attente.

Livraison asynchrone des messages

Lorsqu'un message est transmis en mode asynchrone à une file d'attente éloignée, l'objet message est placé logiquement dans l'espace de stockage de sauvegarde associé à la définition de cette file d'attente, avec le nom de son gestionnaire de files d'attente de destination et celui de la file d'attente, ainsi que les caractéristiques des modules de compression, de chiffrement et de l'authentificateur correspondant à la cible du message. La méthode de vidage de l'objet est appelée au moment où il est sauvegardé en mémoire persistante dans un format sécurisé, tel que défini par la file d'attente de destination. Le gestionnaire de files d'attente contrôle la livraison du message. Il identifie (ou établit) un canal doté des caractéristiques adéquates vers le gestionnaire de files d'attente pour le tronçon suivant, puis il crée (ou réutilise) un transporteur vers la file d'attente cible. Le transporteur effectue un vidage de l'objet et transmet la chaîne de caractères binaires résultante. Notez que le gestionnaire de files d'attente cible et le nom de la file d'attente ne sont pas intégrés au flux du message.

Le cas échéant, le message est chiffré et compressé sur le canal. S'il atteint le gestionnaire de files d'attente de destination, il est déchiffré et décompressé. Un nouvel objet message est créé, via la méthode de restauration de cette classe d'objet, l'objet résultant étant placé dans la file d'attente de destination. Si le message n'est pas parvenu au gestionnaire de files d'attente de destination, il est déchiffré et décompressé, puis placé dans une file d'attente de stockage et réacheminement avec les caractéristiques appropriées pour la transmission suivante. Dans les deux cas, il est placé en mémoire persistante dans un format sécurisé, tel que défini par la file d'attente de destination.

La livraison asynchrone des messages présente la particularité de transmettre les messages au gestionnaire de files d'attente de chaque étape intermédiaire, sur lequel ils sont mis en file d'attente en vue de la transmission suivante. Les messages sont extraits des files d'attente intermédiaires selon leur priorité, puis selon leur horodatage.

livraison des messages

Livraison synchrone des messages

La livraison synchrone de messages est semblable à la livraison asynchrone décrite plus haut, mais l'intervention du gestionnaire de files d'attente lors des étapes intermédiaires a lieu à un niveau bien inférieur, impliquant le transporteur et les canaux. Un canal est établi de bout en bout, à l'aide des adaptateurs définis dans les spécifications du protocole à chaque nœud intermédiaire, pour identifier la liaison suivante. À la fin de la dernière liaison, lorsqu'il n'existe plus de descripteur de fichier concerné, le message est transmis aux couches supérieures du gestionnaire de files d'attente pour y être traité. Ainsi le nœud expéditeur ne met pas le message en file d'attente, mais le transmet par le canal, via des étapes intermédiaires, puis le remet au gestionnaire de files d'attente destinataire pour qu'il le place dans la file d'attente cible.

La liaison dans MQSeries utilise une file d'attente de pont au niveau de la passerelle, qui convertit le message en un format MQSeries. Ce mécanisme signifie que la messagerie synchrone de type MQSeries Everyplace est possible à partir d'une unité vers MQSeries, le canal dynamique se terminant à la passerelle. Le message est livré en temps réel à partir de la passerelle, via un canal client, à un serveur MQSeries. À ce stade, sa destination peut imposer qu'il soit acheminé en mode asynchrone le long des canaux de message MQSeries.

De même, une unité qui ne peut travailler en mode synchrone peut envoyer des messages vers une file d'attente MQSeries Everyplace asynchrone, sous réserve qu'un intermédiaire approprié soit disponible.

Sécurité

MQSeries Everyplace fournit un ensemble intégré de fonctions de sécurité permettant de protéger les données des messages lors de leur stockage local et de leur transfert.

Les fonctions de sécurité de MQSeries Everyplace assurent une protection à un triple niveau :

- Sécurité locale - protection locale des données des messages (et autres données)
- Sécurité au niveau des files d'attente - protection de messages entre le gestionnaire de files d'attente expéditeur et la file d'attente cible
- Sécurité au niveau message - protection des messages entre l'expéditeur et le destinataire

Les fonctions de sécurité MQSeries Everyplace locales et au niveau message sont utilisées en interne par MQSeries Everyplace, mais sont également à

disposition des applications MQSeries Everyplace. La fonction de sécurité MQSeries Everyplace au niveau des files d'attente est un service interne.

Les trois types de fonctions de sécurité MQSeries Everyplace assurent la protection des messages via un attribut (MQeAttribute ou descendant). Selon le type, l'attribut est appliqué explicitement ou implicitement.

Chaque attribut peut contenir les objets suivants :

- Authentificateur
- Module de chiffrement
- Module de compression
- Clé
- Nom de l'entité cible

Ces objets sont utilisés différemment, selon le type de fonction de sécurité MQSeries Everyplace, mais, dans tous les cas, la protection est appliquée lors de l'appel de l'attribut associé à un objet message. Ce qui se produit lors de l'appel d'une méthode de vidage ("dump") MQSeries Everyplace du message (par exemple, lorsque la méthode "encodeData" de l'attribut est utilisée pour chiffrer et compresser les données du message). La protection par les fonctions de sécurité MQSeries Everyplace cesse lorsque la méthode "restore" du message est appelée (par exemple, lorsque la méthode "decodeData" de l'attribut est utilisée pour décompresser et déchiffrer les données du message).

Les algorithmes pris en charge par MQSeries Everyplace version 1.2 pour l'authentification, le chiffrement et la compression sont répertoriés dans le tableau 15.

Tableau 15. Support de l'authentification, du chiffrement et de la compression

Fonction	Algorithme
Authentification	WTLS mini-certificat
	Validation d'identité Windows NT/2000, AIX ou Solaris
Compression	LZW
	RLE
Chiffrement	Triple DES
	DES
	MARS
	RC4
	RC6
	XOR

Sécurité locale MQSeries Everyplace

Les fonctions de sécurité locales protègent localement les données des messages MQSeries Everyplace (ou les descendants MQeFields ou MQeFields). Pour ce faire, il convient de créer un attribut, avec un module de chiffrement et de compression symétrique approprié, de créer et de configurer une *clé* adéquate (par mot de passe simple ou composé). Cette clé doit être associée explicitement à l'attribut, puis l'attribut au message MQSeries Everyplace. MQSeries Everyplace fournit la classe MQeLocalSecure, qui aide à la configuration des fonctions de sécurité locales. Toutefois, il incombe dans tous les cas à l'utilisateur de la fonction de sécurité locale (MQSeries Everyplace en interne ou application MQSeries Everyplace) de configurer l'attribut approprié et de gérer la clé par mot de passe simple ou composé.

Sécurité MQSeries Everyplace au niveau des files d'attente

Les fonctions de sécurité au niveau des files d'attente sont applicables aux messages synchrones et asynchrones.

Sécurité au niveau des files d'attente synchrones

L'utilisation de fonctions de sécurité fondées sur les files d'attente permet à une application de déléguer à MQSeries Everyplace tout souci concernant la sécurité des messages. Les fonctions d'authentification, de chiffrement et de compression associées aux files d'attente sont utilisées pour déterminer le niveau de sécurité requis pour protéger les flux de message (ainsi que les messages stockés en mémoire persistante).

Lorsqu'un message est sur le point d'être envoyé, les fonctions de sécurité de la file d'attente cible sont extraites du registre local. En leur absence, le gestionnaire de files d'attente tente de les retrouver à partir du gestionnaire cible et les place en mémoire cache pour les utiliser ultérieurement. S'il existe un canal vers ce gestionnaire de files d'attente, il est utilisé. Sinon, il est créé. Les attributs de la file d'attente cible sont extraits.

Selon la qualité du service requis, les attributs du canal vers le gestionnaire de files d'attente cible sont modifiés dynamiquement. Cette opération est soumise à toutes les règles éventuellement établies. En général, une règle autorise la progression du niveau de sécurité (par exemple, de l'absence de protection à une protection de faible niveau, ou de faible à fort). S'il est impossible de mettre à jour le canal ou que le niveau de sécurité est jugé excessif (par exemple, aucune protection requise et un canal implémentant un niveau de protection maximal), un nouveau canal est créé. Il existe un pool de canaux, réutilisés chaque fois que possible, dont les caractéristiques sont modifiées dynamiquement selon les impératifs du trafic. Les canaux devenus inutiles sont automatiquement détruits. Les messages sont toujours placés dans les files d'attente au niveau de sécurité défini par les caractéristiques de la file d'attente cible.

L'authentification a lieu au niveau du canal, pour réduire au minimum la charge par message. De même, la sécurité au niveau des files d'attente synchrones est généralement associée à des modules de chiffrement symétriques, ce qui accélère les processus de chiffrement/déchiffrement. Toutefois, dans ces cas de symétrie, MQSeries Everyplace utilise au départ le chiffrement asymétrique RSA, afin de protéger le flux requis pour établir une clé partagée pour l'expéditeur et le destinataire. Le chiffrement symétrique est utilisé ensuite pour protéger la confidentialité des données transmises. MQSeries Everyplace rend difficile toute attaque visant à déchiffrer ces données, en modifiant dynamiquement la clé sur chaque flux du canal. MQSeries Everyplace augmente encore la sécurité en ajoutant aux données transmises la synthèse de celles-ci avant leur envoi, puis en la régénérant et en la validant à réception.

Sécurité au niveau des files d'attente asynchrones

La transmission asynchrone des messages diffère de leur transmission synchrone, décrite plus haut, en ceci qu'il n'existe pas de garantie que la file d'attente cible soit accessible au moment de l'exécution de `putMessage`. Dans ce cas, le gestionnaire de files d'attente ne peut envoyer le message immédiatement et le place dans sa file d'attente de transmission. Le message est toutefois chiffré en fonction des caractéristiques de la file d'attente cible. Lorsqu'il peut être transmis, il est déchiffré, puis envoyé via un canal doté des caractéristiques appropriées. Les messages sont ainsi toujours protégés, même lorsqu'ils sont en attente de transmission. La transmission asynchrone des messages nécessite une définition de file d'attente éloignée, en l'absence de laquelle il est impossible de déterminer les caractéristiques de la file d'attente cible.

En mode asynchrone, l'authentification entre l'expéditeur et le destinataire est impossible. Si l'authentification est importante (lorsque, par exemple, le destinataire doit déterminer qui envoie le message pour l'accepter ou établir son irréfutabilité, ou que l'expéditeur souhaite s'assurer que le message ne peut être traité que par le destinataire indiqué), il convient d'utiliser les fonctions de sécurité au niveau message.

Les fonctions de sécurité au niveau des files d'attente peuvent être utilisées simultanément avec les fonctions de sécurité au niveau message, mais cela est superflu, les données des messages étant déjà protégées.

Sécurité au niveau message

Les fonctions de sécurité au niveau message assurent la protection des données des messages entre une application MQSeries Everyplace expéditrice et une application réceptrice.

La sécurité au niveau message est un service de la couche application qui requiert que l'application MQSeries Everyplace émettrice définisse un attribut

de niveau message et le fournisse lors de l'exécution de `putMessage` pour placer le message dans une file d'attente cible. L'application réceptrice doit définir et transmettre un attribut de niveau message correspondant au gestionnaire de files d'attente récepteur de sorte que l'attribut soit disponible lorsque l'application appelle `getMessage` pour récupérer le message de la file d'attente cible.

À l'instar de la sécurité locale, la sécurité au niveau message exploite l'application d'un attribut sur un objet message. Le gestionnaire de files d'attente de l'application émettrice gère la demande `putMessage` avec la méthode "dump", laquelle se sert de la méthode "encodeData" de l'attribut pour protéger les données du message. Le gestionnaire de files d'attente de l'application réceptrice gère la demande `getMessage` de l'application par la méthode "restore", laquelle se sert de la méthode "decodeData" de l'attribut pour restaurer les données du message d'origine.

MQSeries Everyplace propose deux autres attributs pour la sécurité au niveau message :

MQeMAttribute

Cet attribut est utilisé pour les communications inter-entreprises, lorsque la sécurisation mutuelle est strictement gérée dans la couche application et ne requiert pas de tiers sécurisé. Tous les modules de chiffrement et de compression MQSeries Everyplace symétriques disponibles peuvent être utilisés. Comme pour la sécurité locale, la clé de l'attribut doit être prédéfinie, puis fournie avec `putMessage` ou `getMessage`. MQeAttribute fournit une méthode à la fois simple et puissante pour la protection au niveau message, permettant un chiffrement efficace pour protéger la confidentialité des messages, sans la surcharge induite par toute infrastructure de clés publiques (PKI, public key infrastructure).

MQeMTrustAttribute

Cet attribut propose une solution plus élaborée, exploitant les signatures numériques et l'infrastructure de clé publique (PKI) par défaut. Il utilise la signature numérique/validation ISO9796 pour permettre à l'application réceptrice d'établir la preuve que le message provient effectivement de l'expéditeur supposé. Le module de chiffrement d'attribut fourni sert à protéger la confidentialité des messages. La synthèse SHA1 garantit l'intégrité des messages et le chiffrement/déchiffrement RSA assure que le message ne peut être restauré que par le destinataire indiqué. Comme avec MQeMAttribute, tous les modules de chiffrement et de compression MQSeries Everyplace symétriques disponibles peuvent être utilisés. Sélectionnés pour l'optimisation de la taille, les certificats utilisés sont des mini-certificats WTLS. La disponibilité mutuelle des informations

requis pour authentifier (valider les signatures) et pour chiffrer/déchiffrer est fournie via l'infrastructure MQSeries Everyplace par défaut.

Un message protégé par MQeMTrustAttribute a généralement le format :

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

où :

RSA-enc:	chiffré par RSA avec la clé publique du destinataire indiqué
SymKey	clé symétrique pseudo-aléatoire générée
SymKey-enc	chiffrée symétriquement avec SymKey
Data	données du message
DataDigest	synthèse des données du message
DigSignature	signature numérique de l'expéditeur des données du message

La sécurité au niveau message est indépendante de la sécurité au niveau des files d'attente.

Registre

Le registre est le principal espace de stockage des informations concernant le gestionnaire de files d'attente. Il en existe un par gestionnaire de files d'attente. Chaque gestionnaire utilise le registre pour y conserver :

- ses données de configuration,
- les définitions des files d'attente,
- les définitions des files d'attente éloignées,
- les définitions du gestionnaire de files d'attente éloignées,
- les données utilisateur (dont les informations de sécurité dépendant de la configuration).

L'accès au registre, protégé par un PIN, est normalement réservé à l'utilisateur légitime du gestionnaire de files d'attente, mais une option configurable permet aux utilisateurs plus soucieux de la taille des empreintes que de la sécurité de passer outre.

Entités MQSeries Everyplace authentifiables

La sécurité au niveau des files d'attente, qui utilise l'authentification mutuelle par mini-certificats et la protection au niveau message, qui utilise les signatures numériques, ont amené à définir le concept d'"entité authentifiable". Dans le cas d'authentification mutuelle, il paraît normal de penser à l'authentification entre deux utilisateurs (personnes), mais en général, la fonction de messagerie n'intègre pas le concept d'utilisateur. Ce concept est

généralement géré au niveau application, c'est-à-dire par l'utilisateur des services de messagerie. MQSeries Everyplace abstrait délibérément le concept de *cible de l'authentification* en abandonnant toute référence à un *utilisateur*) au profit de l'*entité authentifiable*. Ce qui n'exclut pas la possibilité qu'une entité authentifiable soit une personne, mais il s'agit alors d'un mappage sélectionné par une application. En interne, MQSeries Everyplace définit comme entités authentifiables tous les gestionnaires de files d'attente qui peuvent être soit l'origine, soit la cible de services dépendant de mini-certificats. En outre, MQSeries Everyplace définit également comme entités authentifiables les files d'attente configurées pour utiliser des authentificateurs basés sur des mini-certificats. Ainsi, un gestionnaire de files d'attente qui prend en charge ces services peut être associé à une entité authentifiable, le gestionnaire de files d'attente, ou à un ensemble d'entités authentifiables, le gestionnaire de files d'attente et toutes les files d'attente qui utilisent un authentificateur basé sur les certificats.

Registre privé et accréditations

Pour être utile, chaque entité authentifiable doit être dotée de ses propres accréditations. Ce qui pose deux questions. La première consiste à savoir comment effectuer l'enregistrement pour obtenir les accréditations, et la seconde, à déterminer où gérer les accréditations de façon sécurisée. Normalement, ces questions sont plus difficiles à résoudre que ne le sont les techniques de chiffrement sous-jacentes. MQSeries Everyplace fournit des services par défaut qui permettent aux entités authentifiables de s'enregistrer automatiquement. Un registre privé (descendant du registre de base) pour la gestion sécurisée des accréditations privées d'une entité authentifiable, et un registre public (également descendant du registre de base) pour la gestion de l'ensemble des accréditations publiques. Le registre privé fournit un registre de base avec plusieurs des qualités d'un jeton sécurisé ou chiffré. Ce peut être, par exemple, un référentiel sécurisé pour des objets publics tels que les mini-certificats, et des objets privés tels que les clés privées. Il fournit un mécanisme qui réserve aux seuls utilisateurs autorisés l'accès aux objets privés. Il assure également le support de services (par exemple, signature numérique, déchiffrement RSA) de telle sorte que les objets privés ne quittent jamais le registre privé. En fournissant une interface commune, il cache le support d'unité sous-jacent, actuellement restreint au système de fichiers local, mais qui pourrait être étendu pour correspondre aux jetons des portables dans un proche avenir.

Enregistrement automatique

MQSeries Everyplace fournit des services par défaut qui prennent en charge l'enregistrement automatique. Ces services sont automatiquement déclenchés lorsqu'une entité authentifiable est configurée, par exemple au démarrage d'un gestionnaire de files d'attente ou à la définition d'une nouvelle file d'attente. Dans les deux cas, l'enregistrement est déclenché et des accréditations sont créées et stockées dans le registre privé de l'entité privée.

Les étapes d'enregistrement automatique comprennent la génération d'une nouvelle paire de clés RSA, la protection et la sauvegarde de la clé privée dans le registre privé et l'intégration de la clé publique dans une demande de "nouveau certificat" adressée au serveur de mini-certificats par défaut. Si le serveur de certificats est configuré et disponible, il renvoie les nouveaux mini-certificats de l'entité authentifiable, avec son propre mini-certificat, et ceux-ci, avec la clé privée protégée, sont stockés dans le registre privé de l'entité authentifiable comme ses nouvelles accréditations. Alors que l'enregistrement automatique fournit un mécanisme simple pour établir les accréditations d'une entité authentifiable, la protection au niveau message (MQeMTrustAttribute, voir plus haut) impose l'accès à la clé publique (mini-certificat) du destinataire désigné.

Registre public et réplication de certificat

MQSeries Everyplace fournit des services par défaut qui permettent le partage des accréditations publiques d'une entité authentifiable (mini-certificats) entre les composants MQSeries Everyplace. Ce sont des conditions préalables pour la sécurité au niveau message MQeMTrust. Le registre public MQSeries Everyplace fournit un référentiel publiquement accessible pour les mini-certificats. Cela est analogue au répertoire téléphonique personnel sur les téléphones mobiles, la différence étant que les numéros de téléphone sont ici un ensemble de mini-certificats des entités authentifiables les plus souvent contactées. Le registre public n'est pas complètement passif dans ses services. S'il fait l'objet d'un accès pour fournir un mini-certificat qu'il ne détient pas, et sous réserve qu'il soit configuré avec un composant serveur local valide, le registre public tente automatiquement de récupérer le mini-certificat demandé dans le registre public du serveur local. Ces services peuvent être utilisés pour fournir un service de réplication de mini-certificats automatisé intelligent, qui assure la disponibilité du mini-certificat requis au moment requis.

Utilisation des services de registre par les applications

Dans la mesure où le gestionnaire de files d'attente MQSeries Everyplace est conçu pour exploiter les avantages des services de registres privés et publics, l'accès à ces services n'est pas limité. Les solutions MQSeries Everyplace peuvent souhaiter définir et gérer leur propres entités authentifiables, les utilisateurs par exemple. Les services de registre privé peuvent dans ce cas être utilisés pour enregistrer automatiquement et gérer les accréditations des nouvelles entités authentifiables, les services de registre public assurant alors la mise à disposition des accréditations publiques lorsque cela est nécessaire. Toutes les entités authentifiables enregistrées peuvent être utilisées comme expéditeur ou récepteur des services au niveau message, protégés via MQeMTrustAttribute.

Service d'émission de mini-certificats par défaut

MQSeries Everyplace fournit un service d'émission de mini-certificats par défaut, qui peut être configuré pour répondre aux demandes d'enregistrement automatique de registres privés. Grâce aux outils fournis par MQSeries

Everyplace, une solution peut configurer et gérer un service d'émission de mini-certificats, qui délivre des mini-certificats à un ensemble strictement contrôlé de noms d'entités. Les caractéristiques de ce service d'émission sont les suivantes :

- Gestion de l'ensemble des entités authentifiables enregistrées
- Émission de mini-certificats
- Gestion du référentiel de mini-certificats WAP WTLS

Les outils fournis par MQSeries Everyplace permettent à un administrateur du service d'émission de mini-certificats d'autoriser la livraison de mini-certificat à une entité spécifique, en enregistrant son nom et son adresse et en définissant un PIN de demande de certificat à usage unique. Cette opération est normalement effectuée une fois que le contrôle hors ligne a vérifié l'authenticité du demandeur. Le PIN de demande de certificat est posté à l'utilisateur désigné (de la même manière, par exemple, que les PIN de carte bancaire sont postés aux utilisateurs à l'émission d'une nouvelle carte). L'utilisateur du registre privé (par exemple, l'application MQSeries Everyplace ou le gestionnaire de files d'attente MQSeries Everyplace) peut alors être configuré pour fournir ce PIN de demande de certificat au moment du démarrage. Lorsque le registre privé déclenche l'enregistrement automatique, le service d'émission de mini-certificats valide la nouvelle demande de certificat résultante (sur la base d'une correspondance entre le nom d'entité présentée et le PIN de demande de certificat avec les valeurs préenregistrées), délivre le nouveau mini-certificat et réinitialise le PIN de demande du certificat enregistré de sorte qu'il ne puisse pas être réutilisé. Toutes les demandes d'enregistrement automatique de nouveaux mini-certificats sont traitées via un canal sécurisé.

L'ensemble des mini-certificats délivrés par un service d'émission de mini-certificats est conservé dans le registre propre au service. Lorsqu'un mini-certificat est ré-émis (par exemple, suite à une expiration), le mini-certificat arrivé à expiration est archivé.

Interface de sécurité

Une interface en option peut être implémentée par un gestionnaire de sécurité personnalisé. Les méthodes permettent à ce gestionnaire d'accepter ou de rejeter les demandes associées à :

- L'ajout ou le retrait d'alias de classe
- La définition d'adaptateurs
- Le mappage de descripteurs de fichier
- Le traitement des commandes de canaux

Personnalisation

Règles

Les règles sont des classes Java utilisées pour personnaliser le comportement de MQSeries Everyplace lorsque se produisent différents changements d'état. Des règles par défaut sont fournies lorsque cela est nécessaire, mais elles peuvent être remplacées par des règles spécifiques à l'application - ou à l'installation - pour mieux répondre aux besoins du client. Les types de règles pris en charge varient selon leur mode de déclenchement, et non selon leur action. Les règles comportent une logique et sont donc susceptibles d'effectuer de nombreuses fonctions.

Règles d'attribut

Cette classe de règles prend le contrôle à chaque tentative de changement d'état ; par exemple, un changement de l'un des éléments suivants :

- Authentificateur
- Module de compression
- Module de chiffrement

La règle autorise ou refuse le changement.

Règles du pont MQSeries

Ces classes de règles prennent le contrôle lorsque le code MQSeries Everyplace vers le code du pont MQSeries subit un changement d'état. Il existe une classe de règles de pont distincte pour déterminer ce qui suit :

- Que faire d'un message issu de MQSeries lorsqu'un programme d'écoute ne peut le livrer sur MQSeries Everyplace. Par exemple, lorsque le message est trop volumineux et qu'il n'existe pas de file d'attente.
- L'état initial dans lequel les objets administrés par le pont doivent démarrer dès que le serveur est instancié.
- Que faire lorsque le pont découvre une anomalie au niveau de la file d'attente Sync sur MQSeries (espace de stockage persistant utilisé pour la reprise en cas de panne). La règle par défaut se contente de signaler le problème.
- Comment convertir un message MQSeries Everyplace en message MQSeries, et vice versa. Les fonctions Transformation susceptibles d'effectuer la conversion entre messages MQSeries Everyplace et messages MQSeries ne sont pas dérivées de classes MQeRule. Elles doivent implémenter l'interface MQeTransformerInterface. Cette différence mise à part, les fonction Transformation ont le même rôle que les règles et sont appelées lorsqu'un message doit être converti.

personnalisation

Règles de file d'attente

Cette classe de règles prend le contrôle à chaque changement d'état de la file d'attente associée ; par exemple :

- Ajout d'un message à la file d'attente. Par exemple, pour vérifier s'il y a dépassement d'un seuil (nombre de messages, taille de message, priorité incorrecte).
- Affectation ou modification des caractéristiques de la file d'attente.
- Ouverture ou fermeture de la file d'attente.
- File d'attente sur le point d'être supprimée.

Règles du gestionnaire de files d'attente

Cette classe de règles prend le contrôle à chaque changement d'état du gestionnaire de files d'attente ; par exemple :

- Ouverture du gestionnaire de files d'attente. Par exemple, au démarrage d'une horloge en arrière-plan pour permettre l'exécution d'actions programmées.
- Fermeture du gestionnaire de files d'attente. Par exemple, pour arrêter l'horloge en arrière-plan.
- Ajout d'une nouvelle file d'attente.

Styles de connexions

MQSeries Everyplace prend en charge les modes de fonctionnement client-serveur et entre homologues. Un *client* peut être à l'origine d'une communication avec un serveur. En revanche, un *serveur* ne peut que répondre aux demandes émises par un client. En mode *entre homologues* (c'est-à-dire d'égal à égal), les deux homologues peuvent initier des flux dans l'un ou l'autre sens. Ces styles de connexions supposent la disponibilité et l'activation de composants MQSeries Everyplace différents. Les composants impliqués sont les suivants :

- **Programme d'écoute de canal** : est à l'écoute des demandes de connexion entrantes.
- **Gestionnaire de canal** : prend en charge les canaux de communication concurrents logiques multiples entre nœuds d'extrémité.
- **Gestionnaire de files d'attente** : prend en charge les applications via des fonctions de messagerie et de mise en file d'attente.

Le tableau 16 à la page 61 illustre les relations entre ces composants et le style de connexion. Le style de connexion client-serveur décrit la situation dans laquelle MQSeries Everyplace peut opérer indifféremment en mode client ou serveur. L'option servlet décrit le cas où MQSeries Everyplace est configuré comme servlet HTTP, le serveur HTTP lui-même étant responsable de l'écoute des demandes de connexion entrantes.

Tableau 16. Styles de connexions

	Gestionnaire de files d'attente	Gestionnaire de canal	Programme d'écoute de canal
Client	Oui		
Client-serveur	Oui	Oui	Oui
Homologue	Oui		
Serveur	Oui	Oui	Oui
Servlet	Oui	Oui	

Les applications MQSeries Everyplace ne sont pas directement avisées du style de connexion utilisé par les gestionnaires de files d'attente. Le style est toutefois significatif en ceci qu'il a une incidence sur les ressources disponibles pour les deux parties, sur les gestionnaires de files d'attente susceptibles de se connecter à d'autres, sur l'empreinte MQSeries Everyplace, et sur les connexions admises à coexister.

Connexion entre homologues

Un canal entre homologues comprend les fonctions d'un gestionnaire de canal et un programme d'écoute pour un seul canal. À la création d'un canal entre homologues entre deux gestionnaires de files d'attente, l'un de ceux-ci doit jouer le rôle de programme d'écoute, et l'autre, celui d'initiateur de la connexion. Un gestionnaire de files d'attente connecté en mode d'égal à égal peut initier plusieurs connexions d'égal à égal avec d'autres gestionnaires de files d'attente, mais ne peut répondre qu'à une demande de connexion entrante. Il doit ensuite attendre la fermeture du canal entre homologues avant de répondre à une autre demande. Via un canal entre homologues quelconque, les deux gestionnaires de files d'attente concernés peuvent tous deux initier des actions. Ainsi, par exemple, les applications de chacun des gestionnaires ont accès aux files d'attente de l'autre.

Les canaux entre homologues peuvent être inutilisables à travers des pare-feu, dans la mesure où la cible de la demande de connexion entrante peut être rejetée par le pare-feu.

Connexion client-serveur

Les canaux standard, utilisés pour le style de connexion client-serveur, sont dépourvus de fonctions d'écoute, et dépendent d'un programme d'écoute indépendant, au niveau du serveur, lequel serveur requiert un gestionnaire de canal pour gérer les canaux concurrents multiples. Le client initie la demande de connexion et le serveur répond. Un serveur est généralement capable de gérer plusieurs demandes entrantes issues des clients. Via un canal standard, le client a accès à des ressources sur le serveur. Si une application sur le serveur requiert un accès synchrone aux ressources sur le client, il est nécessaire de configurer un second canal, où les rôles sont inversés. Toutefois,

styles de connexions

les canaux standard étant eux-mêmes bidirectionnels, les messages destinés à un client issus de la file d'attente de transmission de son serveur sont livrés via le canal (client-serveur) standard qui est lancé.

Un client peut être simultanément client de plusieurs serveurs. (Notez qu'il n'est pas besoin d'un gestionnaire de canal pour prendre en charge cette configuration, car les gestionnaires de canaux gèrent plusieurs canaux internes.)

Le style de connexion client-serveur est généralement adapté aux transmissions via un pare-feu, dans la mesure où la cible de la connexion entrante est normalement identifiée comme acceptable par le pare-feu.

Styles de connexions multiples

Un gestionnaire de files d'attente peut à lui seul être capable d'initier des connexions d'égal à égal ou client-serveur, et de répondre à l'un ou l'autre en tant que serveur ou en tant qu'homologue. Dans ce cas, le programme d'écoute du canal homologue et celui du canal standard doivent être dotés de numéros de ports distincts.

Classes

MQSeries Everyplace propose un choix de classes pour certaines fonctions pour que le comportement de MQSeries Everyplace puisse être personnalisé et réponde aux conditions spécifiques des applications. Parfois, les interfaces avec les classes sont documentées, de sorte qu'il est possible de développer d'autres alternatives. Le tableau 17 à la page 63 recense les possibilités. Les classes peuvent être identifiées explicitement ou via des alias.

Tableau 17. Options de classes

Classe	Alternative proposée	Interfaces documentées
Administration	non	oui
Authentificateurs	oui	non
Adaptateur de communications	oui	oui
Style de communication	oui	non
Modules de compression	oui	non
Modules de chiffrement	oui	non
Journal des événements	exemple fourni	oui
Messages	non	oui
Stockage de file d'attente	oui	non
Règles	classes par défaut fournies	oui
Trace	exemples fournis	oui

Chargement des applications

Lorsqu'un gestionnaire de files d'attente MQSeries Everyplace est configuré pour fonctionner en tant que client (ou homologue), le chargement de toute application dans la JVM incombe à l'application initiatrice. À cet effet, vous pouvez utiliser les fonctions Java standard ou le chargeur de classe intégré à MQSeries Everyplace. Ainsi, plusieurs applications peuvent être exécutées sur un seul gestionnaire de files d'attente dans la même JVM. Il est également possible d'utiliser plusieurs JVM, mais chacune requiert son propre gestionnaire de files d'attente, doté chacun d'un nom unique.

Lorsqu'un gestionnaire de files d'attente MQSeries Everyplace est configuré en tant que serveur, MQSeries Everyplace est lui-même l'application initiatrice. MQSeries Everyplace prend en charge une liste de classes préchargées, lesquelles sont chargées successivement avant le gestionnaire de files d'attente.

Chapitre 6. Réseaux MQSeries Everyplace et MQSeries

Bien qu'un réseau MQSeries Everyplace puisse exister de façon autonome, sans serveur ni réseau MQSeries, dans la pratique, MQSeries Everyplace est souvent utilisé pour compléter une installation MQSeries existante, étendant sa portée à de nouvelles plateformes et unités, ou fournissant des fonctions avancées, telles que des fonctions de sécurité au niveau message ou file d'attente, ou des fonctions de messagerie synchrone. Du point de vue d'une application MQSeries Everyplace, les files d'attente et les gestionnaires de files d'attente MQSeries peuvent être considérés comme de simples files d'attente et gestionnaires éloignés supplémentaires. Il existe toutefois un certain nombre de restrictions fonctionnelles, car l'accès à ces files d'attente ne s'effectue pas directement via des canaux dynamiques MQSeries Everyplace et un gestionnaire de files d'attente MQSeries Everyplace, mais requiert une passerelle MQSeries Everyplace. La passerelle peut envoyer des messages à plusieurs gestionnaires de files d'attente MQSeries, directement ou indirectement, via les canaux client MQSeries. Si la connexion est indirecte, les messages sont transmis via les canaux client MQSeries à un gestionnaire de files d'attente MQSeries intermédiaire, puis, via les canaux de message MQSeries, au gestionnaire de files d'attente cible.

Les messages issus d'une application MQSeries et destinés à MQSeries Everyplace sont adressés au gestionnaire de files d'attente MQSeries Everyplace et placés normalement dans la file d'attente, le routage MQSeries (définitions du gestionnaire de files d'attente éloignées) étant défini de sorte que les messages MQSeries Everyplace parviennent dans des files d'attente de transmission MQSeries spécifiques. Les canaux MQSeries ne sont pas définis pour les files d'attente de transmission, comme d'habitude : la passerelle MQSeries Everyplace extrait les messages de ces files d'attente et s'assure de leur livraison à leur destinataire MQSeries Everyplace. Le nombre de files d'attente de transmission à utiliser (c'est-à-dire le nombre de routes) est configurable et doit être défini en fonction du volume de messages délivrés.

Interface avec MQSeries

L'architecture de MQSeries Everyplace prend en charge le concept d'un ou de plusieurs types de ponts en option entre MQSeries Everyplace et les autres systèmes de messagerie.

Dans MQSeries Everyplace version 1.2, un seul type de pont est admis, le *pont MQSeries*, qui assure les interfaces entre les réseaux MQSeries Everyplace et MQSeries. Ce pont se sert du client Java MQSeries pour établir une interface avec un ou plusieurs gestionnaires de files d'attente MQSeries, permettant

connexion à MQSeries

ainsi aux messages de transiter de MQSeries Everyplace à MQSeries et vice versa. Dans la version actuelle de MQSeries Everyplace, un seul pont de ce type par passerelle est conseillé, chacun étant associé à plusieurs *proxy de gestionnaire de files d'attente MQSeries* (définitions de gestionnaires de files d'attente MQSeries). Une définition de proxy de gestionnaire de files d'attente est requise pour chaque gestionnaire MQSeries communiquant avec MQSeries Everyplace. À chacune de ces définitions peuvent être associés un ou plusieurs *services de connexion client*, chacun représentant une connexion à un seul gestionnaire de files d'attente MQSeries. Chacun de ces gestionnaires peut utiliser une connexion distincte du serveur MQSeries distincte vers le gestionnaire de files d'attente et, éventuellement, un ensemble différent de propriétés, telles que les sorties utilisateur et les ports.

Une passerelle peut être associée à un certain nombre de *programmes d'écoute*, qui passent par cette passerelle pour se connecter au gestionnaire de files d'attente MQSeries et pour extraire les messages de MQSeries vers MQSeries Everyplace. Un programme d'écoute n'utilise qu'un service pour établir la connexion, chaque programme d'écoute étant connecté à une seule file d'attente de transmission sur le gestionnaire de files d'attente MQSeries. Chaque programme d'écoute déplace les messages d'une seule file d'attente de transmission MQSeries vers un emplacement quelconque sur le réseau MQSeries Everyplace, via son gestionnaire de files d'attente de passerelle. Ainsi un seul gestionnaire de files d'attente de passerelle peut canaliser plusieurs sources de messages MQSeries dans le réseau MQSeries Everyplace.

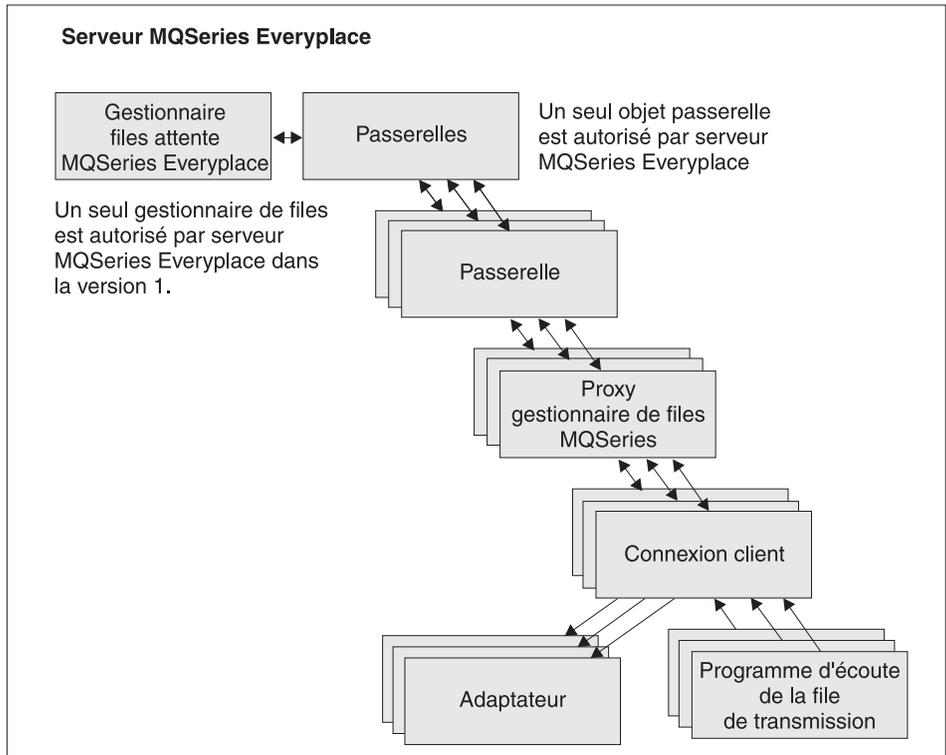


Figure 9. Hiérarchie des objets ponts MQSeries

Lors de la transmission des messages dans l’autre direction, de MQSeries Everyplace vers MQSeries, le gestionnaire de files d’attente de passerelle configure un ou plusieurs objets *files d’attente de pont*. Chaque objet file d’attente de pont peut être connecté directement à n’importe quel gestionnaire de files d’attente et envoyer ses messages vers la file d’attente cible. De la sorte, une passerelle peut distribuer les messages MQSeries Everyplace routés via un seul gestionnaire de files d’attente MQSeries Everyplace vers n’importe quel gestionnaire de files d’attente MQSeries, directement ou indirectement. Les propriétés de l’objet pont sont indiquées dans le tableau 18.

Tableau 18. Propriétés de l’objet pont

Propriété	Explication
Nom pont	Liste des noms de ponts
État exécution	État : actif ou arrêté

Les objets ponts, de même que les autres objets passerelles, peuvent être démarrés et arrêtés indépendamment du gestionnaire de files d’attente MQSeries Everyplace. Si un objet de ce type est démarré (ou arrêté), l’action

connexion à MQSeries

s'applique également à tous ses enfants (tous les ponts, les proxy de gestionnaire de files d'attente, les connexions client et les programmes d'écoute de file d'attente de transmission). Les propriétés de l'objet pont sont indiquées dans le tableau 19.

Tableau 19. Propriétés du pont

Propriété	Explication
Classe	Classe de pont
Fonction Transformation par défaut	Classe par défaut (classe de règles) à utiliser pour transformer un message de MQSeries Everyplace à MQSeries (ou vice versa) si aucune autre classe de fonction Transformation n'a été associée à la file d'attente de destination.
Intervalle des pulsations	Unité de fréquence de base à utiliser pour les opérations sur les objets ponts.
Nom	Nom de l'objet pont.
État exécution	État : actif ou arrêté
Classe de règle de démarrage	Classe de règles utilisée au démarrage de l'objet pont.
Enfants du proxy du gestionnaire de files d'attente MQSeries	Liste de tous les proxy de gestionnaire de files d'attente appartenant à ce pont.

Dans les cas simples, une fonction Transformation par défaut (règle) peut être utilisée pour gérer toutes les conversions de messages. En outre, il est possible de définir une fonction Transformation par programme d'écoute (pour les messages transitant de MQSeries vers MQSeries Everyplace) qui prime sur l'option par défaut. Pour des contrôles plus spécifiques, les règles de transformation peuvent être définies sur la base de la file d'attente cible en utilisant les définitions de files d'attente de pont sur la passerelle. Cela s'applique aux files d'attente cible MQSeries Everyplace et MQSeries.

Le proxy du gestionnaire de files d'attente MQSeries contient les propriétés spécifiques d'un seul gestionnaire de files d'attente MQSeries. Les propriétés du proxy sont indiquées dans le tableau 20.

Tableau 20. Propriétés du proxy du gestionnaire de files d'attente MQSeries

Propriété	Explication
Classe	Classe du proxy du gestionnaire de files d'attente MQSeries

Tableau 20. Propriétés du proxy du gestionnaire de files d'attente MQSeries (suite)

Propriété	Explication
Nom hôte MQSeries	Nom hôte IP utilisé pour créer des connexions avec le gestionnaire de files d'attente MQSeries via les classes client Java. À défaut, le gestionnaire de files d'attente MQSeries est supposé se trouver sur la même machine que le pont et les liaisons Java sont utilisées.
Nom du proxy du gestionnaire de files d'attente MQSeries	Nom du gestionnaire de files d'attente MQSeries
Nom du pont propriétaire	Nom de l'objet pont à qui appartient le proxy du gestionnaire de files d'attente MQSeries.
État exécution	État : actif ou arrêté
Classe de règle de démarrage	Classe de règles utilisée au démarrage de l'objet gestionnaire de files d'attente MQSeries.
Enfants de la connexion client	Liste de tous les objets de connexion client qui appartiennent à ce proxy.

La définition du service de connexion par passerelle contient le détail des informations requises pour établir une connexion avec un gestionnaire de files d'attente MQSeries. Les propriétés de connexion sont indiquées dans le tableau 21.

Tableau 21. Propriétés du service de connexion client

Propriété	Explication
Classe d'adaptateur	Classe à utiliser comme adaptateur de passerelle
CCSID*	Valeur entière du CCSID MQSeries à utiliser
Classe	Classe du service de connexion client par pont
Délai d'inactivité de connexion	Délai maximal d'inactivité de la connexion au-delà duquel elle est arrêtée
Mot de passe MQSeries*	Mot de passe à utiliser par le client Java
Port MQSeries*	Numéro de port IP à utiliser pour créer des connexions vers le gestionnaire de files d'attente MQSeries via les classes client Java. À défaut, le gestionnaire de files d'attente MQSeries est supposé se trouver sur la même machine que le pont et les liaisons Java sont utilisées.
Classe exit de réception MQSeries*	Utilisée pour assurer la correspondance avec l'exit de réception utilisé à l'autre extrémité du canal client ; à l'exit est associée une chaîne qui permet de transmettre les données au code de sortie.

connexion à MQSeries

Tableau 21. Propriétés du service de connexion client (suite)

Propriété	Explication
Classe exit de sécurité MQSeries*	Utilisée pour assurer la correspondance avec l'exit de sécurité utilisé à l'autre extrémité du canal client ; à l'exit est associée une chaîne qui permet de transmettre les données au code de sortie.
Classe exit d'émission MQSeries*	Utilisée pour assurer la correspondance avec l'exit d'émission utilisé à l'autre extrémité du canal client ; à l'exit est associée une chaîne qui permet de transmettre les données au code de sortie.
ID utilisateur MQSeries*	ID utilisateur à utiliser par le client Java
Nom du service de connexion client	Nom du canal de connexion du serveur sur la machine MQSeries.
Nom du proxy propriétaire du gestionnaire de files d'attente	Nom du proxy propriétaire du gestionnaire de files d'attente
Classe de règle de démarrage	Classe de règles utilisée au démarrage de l'objet service de connexion client par pont.
Nom de file d'attente synchrone	Nom de la file d'attente MQSeries utilisée par le pont à des fins de synchronisation
Classe de règles du programme de purge de file d'attente synchrone	Classe de règles à utiliser lorsqu'un message est détecté dans la file d'attente synchrone
État exécution	État : actif ou arrêté
Nom du pont propriétaire	Nom de l'objet pont à qui appartient cette connexion client
Enfants du programme d'écoute MQ XmitQ	Liste de tous les objets programme d'écoute qui utilisent cette connexion client.
*Ces paramètres sont décrits en détail dans la documentation <i>MQSeries - Utilisation de Java</i> .	

La *classe d'adaptateur* est utilisée pour envoyer des messages de MQSeries Everyplace à MQSeries, et la *file d'attente synchrone*, pour garder trace de l'état de ce processus. Son contenu est utilisé dans les situations de reprise pour garantir le fonctionnement sûr de la messagerie ; après un arrêt normal, la file d'attente est vide. Elle peut être partagée par plusieurs connexions client et plusieurs définition de ponts, sous réserve que les exit de réception, d'envoi et de sécurité soient identiques. Cette file d'attente peut également servir au stockage de l'état des messages transitant entre MQSeries et MQSeries Everyplace, selon les propriétés du programme d'écoute utilisé. La *classe de règles du programme de purge de file d'attente synchrone* est utilisée lorsqu'un

message est détecté dans la file d'attente synchrone, indiquant l'échec de MQSeries Everyplace à confirmer un message.

Le délai maximal d'inactivité permet de contrôler le pool de connexions client Java, gérées par le service de connexion client de pont sur son système MQSeries. Lorsqu'une connexion MQSeries devient inactive, parce que non utilisée, un chronomètre est démarré et la connexion est abandonnée si le délai imparti expire avant que la connexion soit réutilisée. La création de connexions MQSeries étant une opération coûteuse, ce processus garantit la réutilisation efficace des connexions sans consommation excessive de ressources. La valeur zéro indique qu'aucun pool de connexions ne doit être utilisé.

Les propriétés de l'objet programme d'écoute, qui transmet les messages de MQSeries vers MQSeries Everyplace, sont indiquées dans le tableau 22.

Tableau 22. Propriétés du programme d'écoute

Propriété	Explication
Classe	Classe du programme d'écoute
Nom de la file d'attente de rebut	File d'attente utilisée pour conserver les messages de MQSeries vers MQSeries Everyplace qui ne peuvent être délivrés.
Adaptateur de stockage de l'état du programme d'écoute	Nom de la classe de l'adaptateur utilisé pour stocker les informations d'état.
Nom du programme d'écoute	Nom de la file d'attente XMIT MQSeries fournissant les messages
Nom du service de connexion client propriétaire	Nom du service de connexion client
État exécution	État : actif ou arrêté
Classe de règle de démarrage	Classe de règles utilisée au démarrage de l'objet programme d'écoute.
Classe de la fonction Transformation	Classe de règles utilisée pour déterminer la conversion d'un message MQSeries en MQSeries Everyplace
Classe de règle de message non livré	Classe de règle utilisée pour déterminer l'action déclenchée lorsque des messages de MQSeries vers MQSeries Everyplace ne peuvent pas être livrés.
Délai d'attente d'un message (en secondes)	Option avancée qui peut servir à contrôler les performances du programme d'écoute dans des circonstances exceptionnelles.

connexion à MQSeries

La *classe de règle de message non livré* détermine l'action effectuée lorsqu'un message de MQSeries vers MQSeries Everyplace ne peut pas être livré. Il est généralement placé dans la *file d'attente de rebut* du système MQSeries.

Pour assurer la livraison des messages, la classe du programme d'écoute utilise l'*adaptateur de stockage d'état du programme d'écoute* pour conserver les informations d'état, sur le système MQSeries Everyplace ou dans la file d'attente synchrone du système MQSeries.

Le programme d'écoute de la file d'attente de transmission permet aux files d'attente éloignées MQSeries de se référer aux files d'attente locales MQSeries Everyplace. Vous pouvez également créer des files d'attente éloignées MQSeries Everyplace qui se réfèrent aux files d'attente locales MQSeries. Ces définitions de files d'attente éloignées MQSeries Everyplace, appelées *files d'attente pont MQSeries*, peuvent être utilisées pour extraire (get), insérer (put) et parcourir (browse) des messages dans une file d'attente MQSeries.

Une définition de file d'attente pont MQSeries peut contenir les attributs suivants :

Tableau 23. Propriétés d'une file d'attente pont MQSeries

Propriété	Explication
Noms d'alias	Autres noms pour la file d'attente
Authentificateur	Doit être NULL
Classe	Classe d'objet
Connexion client	Nom du service de connexion client à utiliser
Module de compression	Doit être NULL
Module de chiffrement	Doit être NULL
Expiration	Transmise à la fonction Transformation
Taille maximale de message	Transmise à la classe de règles
Mode	Doit être synchrone
Proxy du gestionnaire de files d'attente MQ	Nom du gestionnaire de files d'attente MQSeries auquel le message doit être envoyé en premier
Pont MQSeries	Nom du pont par lequel transitera le message vers MQSeries
Nom	Nom qui identifie la file d'attente MQSeries éloignée pour MQSeries Everyplace
Gestionnaire de files d'attente propriétaire	Gestionnaire de files d'attente propriétaire de la définition

Tableau 23. Propriétés d'une file d'attente pont MQSeries (suite)

Propriété	Explication
Priorité	Priorité à utiliser pour les messages (sauf si une autre valeur est spécifiée pour le message)
Nom de la file d'attente MQSeries éloignée	Nom de la file d'attente MQSeries éloignée
Règle	Classe de règle pour les opérations sur la file d'attente
Cible du gestionnaire de files d'attente	Gestionnaire de files d'attente MQSeries propriétaire de la file d'attente
Fonction Transformation	Nom de la classe de la fonction Transformation qui convertit le message du format MQSeries Everyplace au format MQSeries
Type	File d'attente de pont MQSeries

Remarque : Les classes de l'*authentificateur* et des modules de *chiffrement* et de *compression* définissent un ensemble d'attributs de file d'attente qui dictent le niveau de sécurité de chaque message transmis à cette file d'attente. Depuis l'instant, sur MQSeries Everyplace, où le message est envoyé initialement, jusqu'au moment où il est transmis à la file d'attente pont MQSeries, le message est protégé avec un niveau de sécurité au moins égal à celui de la file d'attente. Ces niveaux de sécurité *ne sont pas* applicables lorsque la file d'attente pont MQSeries transmet le message au système MQSeries. Ce transfert est effectué sous la protection des exit d'émission et de réception sur la connexion client. Aucun contrôle ne vérifie que le niveau de sécurité de la file d'attente est géré.

Les files d'attente pont MQSeries sont exclusivement synchrones. Les applications asynchrones doivent donc utiliser soit une combinaison de files d'attente MQSeries Everyplace de stockage et de réacheminement et de serveur local, soit des définitions de files d'attente éloignées asynchrones en tant qu'étapes intermédiaires lors de l'envoi de messages aux files d'attente pont MQSeries.

Les applications utilisent les files d'attente pont MQSeries comme une file d'attente MQSeries Everyplace quelconque, et font appel aux méthodes `putMessage`, `browseMessages` et `getMessage` de la classe `MQeQueueManager`. Le nom de file d'attente indiqué comme paramètre dans ces appels est le nom de la file d'attente pont MQSeries, et le nom de gestionnaire de files d'attente est le nom du gestionnaire de files d'attente MQSeries. Toutefois, pour que le nom du gestionnaire de files d'attente MQSeries soit accepté par le serveur

connexion à MQSeries

MQSeries Everyplace local, une définition de connexion portant ce nom doit exister et tous ses paramètres, y compris le nom du canal, doivent avoir la valeur NULL.

Remarque : L'utilisation de getMessage et browseMessages avec des files d'attente pont MQSeries est soumise à certaines restrictions. Il n'est pas possible d'extraire ou de parcourir des messages de files d'attente pont MQSeries qui pointent vers des définitions de files d'attente éloignées MQSeries. Il n'est pas non plus possible d'utiliser des ID de confirmation différents de zéro dans les opérations d'extraction de files d'attente pont MQSeries. Cela signifie que l'opération getMessage ne permet pas la livraison garantie pour des files d'attente pont MQSeries. Si vous souhaitez une telle garantie, vous devez utiliser des programmes d'écoute de files d'attente de transmission pour transférer les messages à partir de MQSeries.

L'administration du pont MQSeries est assurée de la même manière que l'administration d'un gestionnaire de files d'attente MQSeries Everyplace normal - via les messages d'administration. De nouvelles classes de messages sont définies comme le veut l'objet géré. Le tableau 13 à la page 43 recense les classes de message d'administration de passerelle.

Conversion de message

Les messages MQSeries Everyplace destinés à MQSeries transitent par le pont et sont convertis dans un format MQSeries, via une fonction Transformation par défaut ou spécifique de la file d'attente cible. Une fonction Transformation personnalisée offre davantage de souplesse. Il est judicieux, par exemple, d'utiliser une sous-classe de la classe d'objet message MQSeries Everyplace pour représenter les messages d'un type particulier sur le réseau MQSeries Everyplace. Au niveau de la passerelle, une fonction Transformation peut convertir le message dans un format MQSeries par le biais de toute correspondance appropriée entre les zones et les valeurs MQSeries, ou ajouter des données spécifiques représentant la signification de la sous-classe.

La fonction Transformation MQSeries Everyplace vers MQSeries par défaut ne peut exploiter les informations de la sous-classe, mais il a été conçu pour être utilisable dans un maximum de situations. Ses caractéristiques sont les suivantes :

- **Flux des messages de MQSeries Everyplace vers MQSeries :**

La fonction Transformation MQSeries Everyplace vers MQSeries par défaut fonctionne conjointement avec la classe QeMQMMsgObject. Cette classe est une représentation de toutes les zones susceptibles de figurer dans un en-tête de message MQSeries. À l'aide de QeMQMMsgObject, votre

application peut définir des valeurs (priorité, par exemple) via les méthodes `set()`. Ainsi lorsqu'un `MQeMQMsgObject` (ou un objet dérivé de la classe `MQeMQMsgObject`) est transmis via la fonction Transformation `MQSeries Everyplace` par défaut, celui-ci (`MQeBaseTransformer`) récupère les valeurs à l'intérieur de `MQeMSMsgObject` et définit les valeurs correspondantes dans le message `MQSeries` (par exemple, la valeur de la priorité est copiée dans le message `MQSeries`).

Si le message transmis n'est ni un `MQeMQMsgObject`, ni dérivé de la classe `MQeMQMsgObject`, l'intégralité du message `MQSeries Everyplace` est copié dans le corps du message `MQSeries` (*canalisé*). La zone de format du message dans l'en-tête du message `MQSeries` est définie de façon à indiquer que le message `MQSeries` contient un message au format "canalisé" `MQSeries Everyplace`.

- **Flux des messages de `MQSeries` vers `MQSeries Everyplace` :**

Les messages de `MQSeries` vers `MQSeries Everyplace` et les messages transitant dans l'autre sens sont traités de façon similaire. La fonction Transformation par défaut inspecte la zone du type de message de l'en-tête `MQSeries` et réagit en conséquence.

Si l'en-tête `MQSeries` indique un message `MQSeries Everyplace` "canalisé", le corps du message `MQSeries` est reconstitué à l'identique du message `MQSeries Everyplace` d'origine, puis posté dans le réseau `MQSeries Everyplace`.

Si le message n'est pas un message `MQSeries Everyplace` "canalisé", le contenu de l'en-tête du message `MQSeries` est extrait et placé dans un objet `MQeMQMsgObject`. Le corps du message `MQSeries` est traité comme une simple zone d'octets, et est également placé dans l'objet `MQeMQMsgObject`. Le `MQeMQMsgObject` est ensuite posté vers le réseau `MQSeries Everyplace`.

Cette classe `MQeMQMsgObject` et le comportement de la fonction Transformation par défaut signifient que :

- Un message `MQSeries Everyplace` peut transiter sans changement d'un réseau `MQSeries` à un réseau `MQSeries Everyplace`.
- Un message `MQSeries` peut transiter sans changement d'un réseau `MQSeries Everyplace` à un réseau `MQSeries`.
- Une application `MQSeries Everyplace` peut piloter toute application `MQSeries` existante sans la modifier.

Fonction

Au niveau de la messagerie MQSeries Everyplace, les opérations de placement (put) en mode synchrone à partir d'un gestionnaire de files d'attente MQSeries Everyplace sont activées pour les files d'attente MQSeries éloignées. Toutes les autres opérations de messagerie doivent être effectuées en mode asynchrone.

Les messages d'administration MQSeries Everyplace ne peuvent pas être envoyés à un gestionnaire de files d'attente MQSeries. Dans MQSeries Everyplace, AdminQ n'existe pas et le format des messages d'administration est différent de celui utilisé par MQSeries.

Compatibilité

Un réseau MQSeries Everyplace peut exister indépendamment de MQSeries, mais dans de nombreux cas, les deux produits sont nécessaires pour répondre aux conditions de l'application. MQSeries Everyplace peut intégrer un réseau MQSeries préexistant sous réserve de compatibilité des éléments suivants :

Adresse et nom :

- Sémantique d'adressage identique utilisant l'adresse d'un gestionnaire de files d'attente/file d'attente.
- Utilisation commune d'un espace nom ASCII.

Applications :

MQSeries Everyplace peut prendre en charge les applications MQSeries existantes sans changement.

Canaux :

Les passerelles MQSeries Everyplace utilisent les canaux client MQSeries.

Echange et contenu des messages :

- Echange de messages entre MQSeries Everyplace et MQSeries.
- Transparence réseau des messages (les messages issus de MQSeries Everyplace ou de MQSeries peuvent transiter par l'autre réseau sans changement).
- Support mutuel des zones identifiées dans l'en-tête des messages MQSeries.
- Livraison garantie unique des messages.

MQSeries Everyplace version 1.2 ne prend pas en charge toutes les fonctions de MQSeries. Outre les remarques concernant l'environnement, le système d'exploitation et les communications, quelques différences non négligeables sont explicitées ci-après. Notez toutefois qu'au sein de MQSeries Everyplace, nombre de tâches de l'application peuvent être exécutées par d'autres moyens

utilisant les fonctions de MQSeries Everyplace, ou via l'exploitation de la transformation en sous-classes, le remplacement des classes fournies ou l'exploitation des règles, interfaces et autres fonctions de personnalisation intégrées au produit.

- Pas de support de cluster.
- Pas de support de liste de distribution.
- Pas de messages groupés/segmentés.
- Pas de fonctions d'équilibrage de charge/option de secours.
- Pas de message de référence.
- Pas d'options de rapport.
- Pas de support de file d'attente partagée.
- Pas de déclenchement.
- Pas de support d'unité de travail, pas de coordination XA.

Les caractéristiques d'évolutivité et les performances sont différentes.

Livraison garantie unique

MQSeries Everyplace et MQSeries garantissent tous deux la livraison des messages, mais à des niveaux différents. Lorsqu'un message transite de MQSeries Everyplace vers MQSeries, le transfert du message n'est assuré que si la combinaison `putMessage` et `confirmPutMessage` est utilisée (reportez-vous à la section «Configuration d'un gestionnaire de files d'attente» à la page 36). Lorsqu'un message transite de MQSeries vers MQSeries Everyplace, le transfert n'est assuré que si le message est défini comme persistant.

Chapitre 7. Interfaces de programmation

L'interface *SPI (Systems Programming Interface) MQSeries Everyplace* est l'interface de programmation de MQSeries Everyplace. Elle prend en charge les langages Java et C :

La version Java donne accès à toutes les fonctions MQSeries Everyplace. Les classes, méthodes et procédures sont décrites en détail dans le manuel *MQSeries Everyplace for Multiplatform Programming Reference*. Des exemples de programmation MQSeries Everyplace figurent dans le manuel *MQSeries Everyplace for Multiplatforms Programming Guide*.

Le support C pour Palm permet d'accéder à un sous-ensemble de fonctions MQSeries Everyplace utilisables sur les unités Palm. Une description détaillée de ces classes et procédures, ainsi que des conseils de programmation, figurent dans le manuel *MQSeries Everyplace Native Client Information*

Chapitre 8. Initiation à MQSeries Everyplace

La gamme de produits MQSeries Everyplace fournit de façon collective les outils nécessaires au développement, au déploiement et à la gestion de solutions de messagerie et de mise en file d'attente MQSeries Everyplace. Cette gamme comprend :

1. Le logiciel sous licence MQSeries Everyplace (disponible sur support physique auprès d'IBM ou par téléchargement à partir du site Web <http://software.boulder.ibm.com/dl/mqsem/mqsem-p>). Ce logiciel comprend les éléments suivants :
 - classes Java MQSeries Everyplace
 - classes auxiliaires
 - exemples de code source d'applications
 - utilitaires
 - manuels de référence
 - informations de licence

Le progiciel sur support physique comporte également l'autorisation d'utiliser le produit à des fins autres que le développement (production) sur certaines plateformes. Si vous voulez utiliser le produit sur des machines plus importantes ou avec le pont MQSeries, vous devez acquérir des unités de capacité supplémentaires.

2. SupportPacs MQSeries Everyplace (disponibles par téléchargement à partir du site Web <http://software.boulder.ibm.com/dl/mqsem/mqsem-p> [comme ci-dessus] ou <http://www.ibm.com/software/mqseries/everyplace>). Ces compléments essentiels du logiciel sous licence incluent, par exemple :

EAP1 : MQSeries Everyplace - Device code for the Palm OS

Support du langage de programmation C pour le développement d'applications MQSeries Everyplace version 1.0.1 sur Palm OS. (Ce code figure également sur le CD-ROM du produit, dans le fichier eap1.zip.)

EP01 : MQSeries Everyplace - Performance Report

Analyse les performances de MQSeries Everyplace sur différentes plateformes client.

ES01 : MQSeries Everyplace - Administration Tool (MQeExplorer v1.0)

Outil générique pour toutes les plateformes Java qui permet une administration graphique facile des gestionnaires de files d'attente MQSeries Everyplace.

ES02: MQSeries Everyplace - Explorer (MQe_Explorer v1.2)

Outil d'administration MQSeries Everyplace développé exclusivement pour prendre en charge les systèmes d'exploitations Microsoft Windows.

Les outils de gestion inclus dans les SupportPacs MQSeries Everyplace jouent un rôle important dans toutes les phases du développement et de la mise en œuvre des applications. Plus perfectionnés que les utilitaires intégrés au logiciel sous licence, ils apportent une aide déterminante lors de l'initiation à l'utilisation du produit, de l'étape de configuration, de l'inspection de réseaux pilotes et de la gestion de systèmes de production.

Utilisation de MQSeries Everyplace

Compte tenu de la grande diversité des utilisations possibles de MQSeries Everyplace, ce produit ne peut être installé, configuré et déployé de la même manière que les autres produits de la famille MQSeries. En fait, l'adoption de MQSeries Everyplace par une entreprise s'effectue généralement en trois phases :

1. Développement et prototypage

Lors de la phase d'apprentissage, de développement et de prototypage, le produit MQSeries Everyplace peut être installé et utilisé gratuitement, selon les termes de la licence de développement MQSeries Everyplace accordée par IBM. Les applications MQSeries Everyplace sont développées à l'aide des fonctions des classes Java et des sous-programmes C MQSeries Everyplace. Ces applications peuvent être mises en œuvre de différentes manières ; par exemple :

- Un gestionnaire de files d'attente MQSeries Everyplace peut être configuré en tant que démon, une ou plusieurs applications étant lancées dans la même JVM (machine virtuelle Java) et partageant le même gestionnaire de files d'attente.
- L'application incorpore les classes MQSeries Everyplace requises, de sorte qu'elle peut fonctionner sur des postes non équipés de MQSeries Everyplace et lance son propre gestionnaire de files d'attente dans sa propre JVM.
- L'application utilise les classes MQSeries Everyplace qui existent sur la machine cible.

La licence de développement ne prévoit aucune assistance technique de la part d'IBM. Toutefois, une assistance au cours de la phase de développement d'applications et au delà est prévue dans la licence de déploiement (voir ci-dessous).

2. Déploiement

La phase de déploiement recouvre la mise en production et l'utilisation des applications développées et, par conséquent, selon les

termes de la licence MQSeries Everyplace accordée par IBM, elle nécessite l'acquisition d'unités de capacités. Les classes ne peuvent être distribuées avec l'application qu'avec l'accord d'IBM, ou lorsque les utilisateurs bénéficiaires sont déjà autorisés à les utiliser. Sinon, les utilisateurs doivent fournir eux-mêmes les classes nécessaires.

3. Gestion

Lorsque des gestionnaires de files d'attente MQSeries Everyplace sont actifs dans un réseau, des outils sont nécessaires pour les inspecter et les gérer. La prise en charge de MQSeries Everyplace est fournie selon les termes des Conditions internationales d'utilisation des logiciels IBM.

La durée de ce cycle d'adoption justifie les variations du niveau de prise en charge en fonction des plateformes. Pour MQSeries Everyplace avec unités de capacité, (et les SupportPacs de la catégorie 3), IBM effectue la distinction suivante :

- Plateformes sur lesquelles l'installation et le développement d'applications sont pris en charge :
 - Les rapports d'incidents relatifs à l'installation et/ou au développement d'applications et leur utilisation seront acceptés.
- Plateformes sur lesquelles le développement d'applications est autorisé mais n'est pas directement pris en charge :
 - Les incidents devront peut-être être reproduits sur une plateforme prise en charge.
- Plateformes sur lesquelles le déploiement d'applications est pris en charge :
 - Les rapports d'incidents relatifs au déploiement d'applications seront acceptés.

Acquisition d'expérience

Vous pouvez vous initier à MQSeries Everyplace de plusieurs manières. L'expérience montre toutefois que créer et mettre en œuvre un gestionnaire de files d'attente, puis un réseau MQSeries Everyplace simple, constitue une façon productive de se familiariser avec le produit et ses concepts. L'écriture d'une application simple constitue ensuite une bonne préparation à l'étude approfondie des fonctions du produit. Au début, il est généralement inutile d'étudier les autres membres de la famille MQSeries. En revanche, plus tard, lorsque la fonction de pont présente de l'intérêt, leur compréhension devient indispensable.

Compte tenu de cette stratégie, il est recommandé aux utilisateurs néophytes d'adopter l'approche suivante :

Familiarisez-vous avec les concepts essentiels décrits dans le manuel *MQSeries Everyplace for Multiplatforms - Introduction* (c'est-à-dire le présent ouvrage), puis :

1. Si vous avez accès à une machine équipée d'un système d'exploitation Microsoft Windows, téléchargez MQe_Explorer, SupportPac ES02 (MQe_Explorer v 1.2). Il n'est pas nécessaire d'installer au préalable le logiciel sous licence, mais si vous ne le faites pas, vous ne pourrez effectuer que des tâches de développement (selon les termes de l'accord de licence).
2. Suivez les instructions spécifiques fournies ci-dessous à la section «Première utilisation du produit ES02 : MQe_Explorer».

Première utilisation du produit ES02 : MQe_Explorer

MQe_Explorer est livré avec un guide de l'utilisateur complet qui explique comment configurer rapidement un premier gestionnaire de files d'attente. La lecture de ce guide est généralement conseillée à un public plus vaste que les seuls administrateurs de MQe_Explorer, car il contient plusieurs scripts exemples qui illustrent des exemples importants d'opérations MQSeries Everyplace. Le SupportPac contient deux versions exécutable de MQe_Explorer :

MQe_ExplorerX.exe

Le fichier .exe de cette version contient toutes les classes MQSeries Everyplace nécessaires à son fonctionnement. MQe_ExplorerX.exe peut donc fonctionner même si MQSeries Everyplace n'est pas installé sur la machine. Cette version est idéale pour les utilisateurs néophytes.

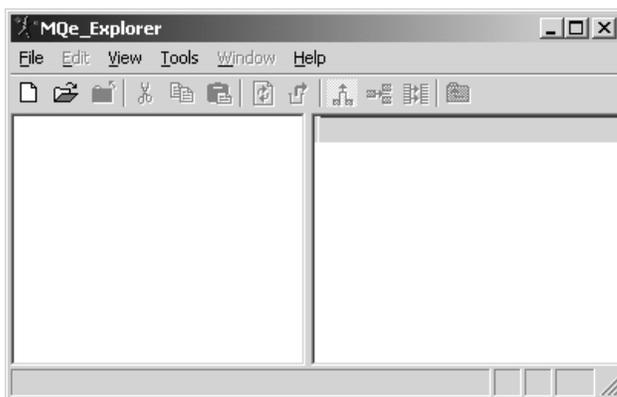
MQe_Explorer.exe

Pour que vous puissiez utiliser cette version, MQSeries Everyplace doit être installé sur le poste de travail. Elle présente l'avantage d'utiliser le dernier niveau des bibliothèques MQSeries Everyplace et d'être d'une taille nettement plus réduite. Elle est destinée aux développeurs et aux administrateurs.

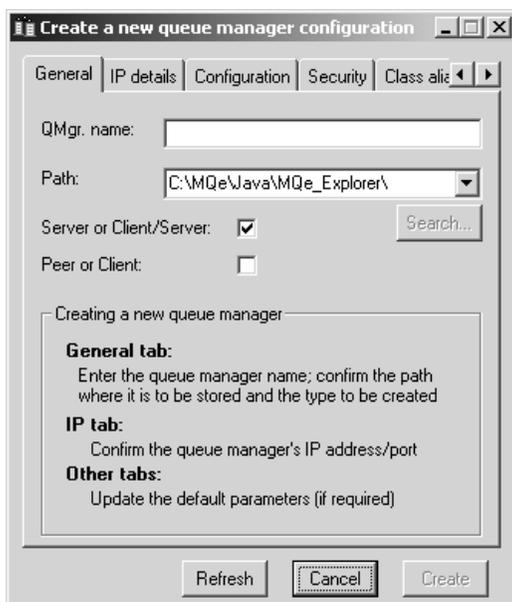
Les instructions abrégées suivantes permettent d'illustrer la puissance de MQSeries Everyplace et la facilité avec laquelle il est possible de créer des gestionnaires de files d'attente :

1. Cliquez deux fois sur l'icône de MQe_ExplorerX.exe .

Lorsqu'un message vous indique qu'aucune option sauvegardée n'a été trouvée, cliquez sur **OK** (ce message ne s'affichera plus). La fenêtre d'administration suivante s'affiche :

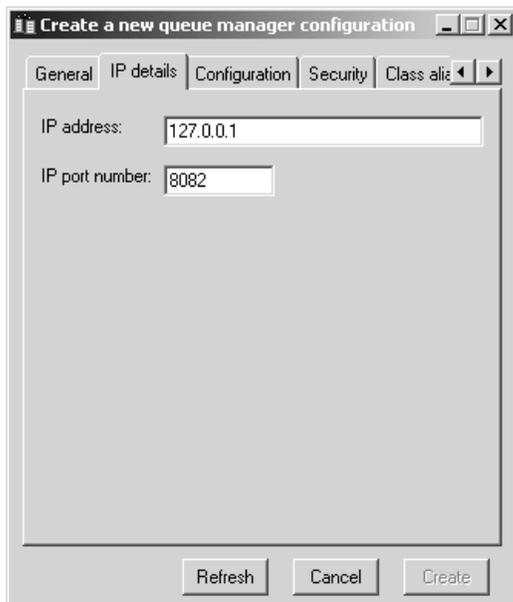


2. Cliquez sur l'icône Nouveau  dans la barre d'outils.
Un nouveau gestionnaire de files d'attente est créé. La fenêtre suivante s'affiche :



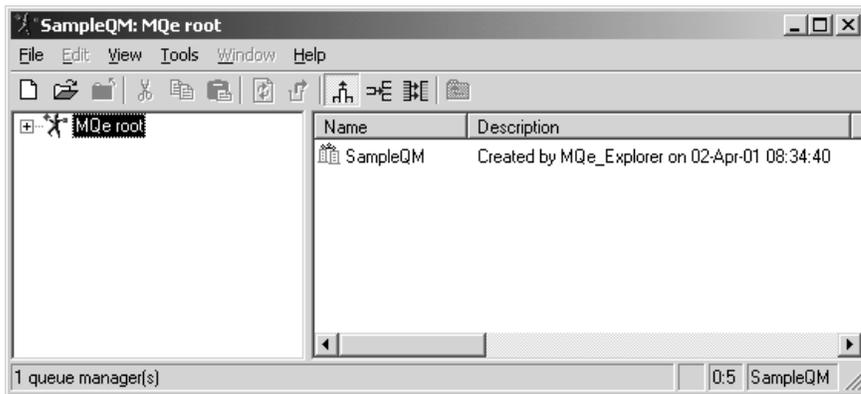
initiation

3.
 - a. Tapez le nom que vous voulez attribuer au gestionnaire de files d'attente (par exemple, SampleQM).
 - b. Cliquez sur l'onglet **IP details**.

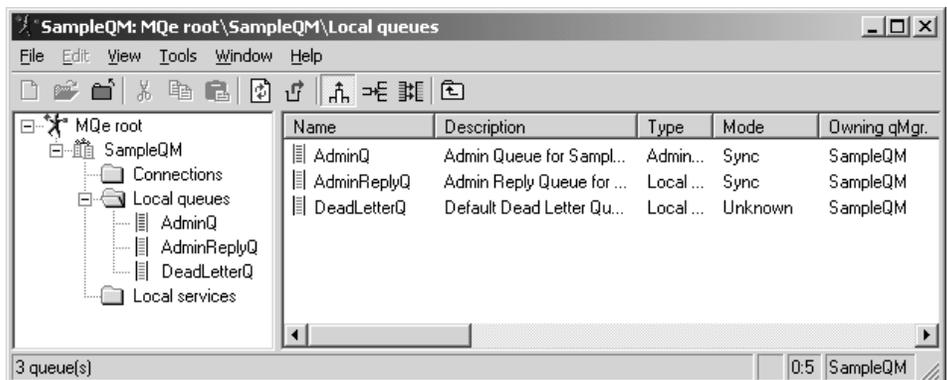


4.
 - a. Tapez l'adresse IP de la machine utilisée (toutefois, à ce stade de l'exercice, il est préférable de taper 127.0.0.1, qui correspond à l'hôte local).
 - b. Cliquez sur le bouton **Create**. Un message affiche le nom du fichier d'initialisation qui a été créé (ce nom est nécessaire pour pouvoir accéder ensuite à ce gestionnaire de files d'attente).

c. Cliquez sur le bouton OK pour afficher la fenêtre suivante :



5. Un gestionnaire de files d'attente serveur est créé, qui s'exécute dans sa propre JVM. Il écoute les demandes de connexion de canal client-serveur entrantes sur le port 8081. Si vous cliquez sur tous les signes + de l'arborescence affichée dans la sous-fenêtre de gauche et que vous redimensionnez les fenêtres et sous-fenêtres, vous pouvez voir que quatre files d'attente ont été créées.



6. Pour une étude plus en profondeur, suivez les instructions du manuel *MQE_Explorer User Guide*. Vous pourrez ainsi créer des files d'attente, des connexions et des messages, et même des réseaux MQSeries Everyplace. Vous pouvez charger des applications dans ce même gestionnaire de files d'attente et les exécuter en même temps que MQE_Explorer.

Annexe. Remarques

Ce document contient des informations sur des produits et des services proposés aux États-Unis. Ce document peut contenir des informations sur des produits, logiciels ou services IBM non annoncés dans d'autres pays. Pour plus de détails, consultez les documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM EMEA Director of Licensing
IBM Europe Middle-East Africa
Tour Descartes
La Défense 5
2, avenue Gambetta
92066 - Paris-La Défense CEDEX
France

Pour le Canada, veuillez adresser votre courrier à :

IBM Director of Commercial Relations
IBM Canada Ltd.
3600 Steeles Avenue East
Markham, Ontario
L3R 9Z7
Canada

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE «EN L'ETAT». IBM DECLINE TOUTE RESPONSABILITE, EXPLICITE OU IMPLICITE, RELATIVE AUX INFORMATIONS QUI Y SONT CONTENUES, Y COMPRIS EN CE QUI CONCERNE LES GARANTIES DE VALEUR MARCHANDE OU

remarques

D'ADAPTATION A VOS BESOINS. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Il est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut modifier sans préavis les produits et logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

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

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans ce document et tous les éléments sous licence disponibles s'y rapportant sont fournis par IBM conformément aux termes du Contrat sur les produits et services IBM, des Conditions internationales d'utilisation des logiciels IBM ou de tout autre accord équivalent.

Marques

Les termes qui suivent sont des marques d'International Business machines Corporation dans certains pays :

AIX AS/400 IBM MQSeries OS/390

Microsoft, Windows, Windows NT et le logo Windows sont des marques de Microsoft Corporation dans certains pays.

Java et toutes les marques et logos incluant Java sont des marques de Sun Microsystems, Inc. dans certains pays.

D'autres sociétés sont propriétaires des autres marques, noms de produits ou logos qui pourraient apparaître dans ce document.

Glossaire

Ce glossaire porte sur des termes spécifiques de ce document et sur des termes utilisés avec une signification inhabituelle. Parfois, la définition donnée peut ne pas être la seule possible, mais elle indique le sens utilisé dans le document.

Si vous ne trouvez pas le terme que vous recherchez, consultez l'index ou reportez-vous à l'ouvrage *IBM Dictionary of Computing*, New York: McGraw-Hill, 1994.

assistant PDA (Personal Digital Assistant) : Ordinateur de poche.

authentificateur : Programme qui vérifie l'identité des expéditeurs et des destinataires des messages.

canal : Voir *canal dynamique* et *canal MQI*.

canal dynamique : Un canal dynamique connecte les unités MQSeries Everyplace et transfère les messages synchrones et asynchrones en mode bidirectionnel.

canal MQI : Un canal MQI connecte un client MQSeries à un gestionnaire de files d'attente sur un système serveur et transfère les appels et les réponses MQI en mode bidirectionnel.

classe : Collection encapsulée de données et de méthodes opérant sur les données. Une classe peut être instanciée, générant alors un objet qui est une instance de la classe.

client : Dans MQSeries, un client est un composant exécutable qui permet aux applications utilisateur locales d'accéder aux services de mise en file d'attente sur un serveur.

encapsulation : Technique de programmation orientée objet qui privatise ou protège les données d'un objet de sorte que les

programmeurs ne puissent y accéder et les manipuler que par le biais d'appels de méthodes.

file d'attente : Une file d'attente est un objet MQSeries. Les applications qui mettent des messages en file d'attente peuvent y placer et y récupérer des messages.

file d'attente des messages : Voir *File d'attente*

gestionnaire de canal : Objet MQSeries Everyplace qui prend en charge plusieurs canaux de communication logiques concurrents entre points terminaux.

gestionnaire de files d'attente : Programme système qui fournit aux applications des services de mise en file d'attente des messages.

instance : Une instance est un objet. Lorsqu'une classe est instanciée pour générer un objet, on dit que l'objet est une instance de la classe.

interface : Classe ne contenant que des méthodes abstraites, sans aucune variable d'instance. Une interface fournit un ensemble de méthodes communes susceptibles d'être mises en oeuvre par les sous-classes de plusieurs classes différentes.

interface de programmation (API) : Une API (Application Programming Interface) est constituée des fonctions et des variables que les programmeurs sont autorisés à utiliser dans leurs applications.

Internet : Internet est un réseau public coopératif d'informations partagées. Physiquement, Internet utilise un sous-ensemble des ressources de tous les réseaux de télécommunication publics existants. Techniquement, c'est son exploitation d'un jeu de protocoles, TCP/IP (Transport Control Protocol/Internet Protocol), qui définit Internet comme réseau public coopératif.

kit JDK (Java Developers Kit) : Package logiciel destiné aux développeurs, distribué par Sun Microsystems. Il contient l'interpréteur Java, les classes et les outils de développement Java : compilateur, débogueur, désassembleur, afficheur d'applets, générateur de fichier raccord et générateur de documentation.

langage HTML (Hypertext Markup Language) : Langage utilisé pour définir les informations à afficher sur le Web.

message : Dans les applications de mise en file d'attente de messages, un message est une communication envoyée d'un programme à un autre.

messagerie asynchrone : Méthode de communication entre programmes dans laquelle les programmes placent les messages dans des files d'attente. En mode messagerie asynchrone, le programme expéditeur poursuit les traitements en cours sans attendre la réponse à son message. S'oppose à *messagerie synchrone*.

messagerie synchrone : Méthode de communication entre programmes dans laquelle les programmes placent les messages dans des files d'attente. En mode messagerie synchrone, le programme expéditeur attend la réponse à son message avant de poursuivre les traitements en cours. S'oppose à *messagerie asynchrone*.

méthode : Terme utilisé en programmation orientée objet pour désigner une fonction ou une procédure.

mise en file d'attente de messages : Technique de programmation dans laquelle chaque programme d'une application communique avec les autres programmes en plaçant les messages en files d'attente.

module de chiffrement : Programme qui chiffre un message pour sécuriser sa transmission.

module de compression : Programme qui compresse un message pour réduire le volume des données à transmettre.

MQSeries : MQSeries est une famille de logiciels sous licence IBM qui fournissent des services de mise en file d'attente de messages.

navigateur Web : Programme qui formate et affiche des informations diffusées sur le Web.

objet : (1) En Java, un objet est une instance d'une classe. Une classe modélise un groupe d'objets ; un objet modélise un membre spécifique de ce groupe. (2) Dans MQSeries, un objet est un gestionnaire de files d'attente, une file d'attente ou un canal.

package : En Java, un package est un moyen de donner à un module de code Java accès à un ensemble spécifique de classes. Le code Java qui fait partir d'un package a accès à toutes les classes du package et à toutes les méthodes et les zones non privées de ces classes.

passerelle : Une passerelle (ou serveur) MQSeries Everyplace est un ordinateur exécutant du code MQSeries Everyplace comportant un gestionnaire de canal.

pont : Objet MQSeries Everyplace qui permet aux messages de transiter entre MQSeries Everyplace et d'autres systèmes de messagerie, MQSeries compris.

privé : Une zone privée n'est pas visible en dehors de sa propre classe.

protégé : Une zone protégée n'est visible qu'au sein de sa propre classe, d'une sous-classe et des packages qui contiennent la classe.

protocole LDAP (Lightweight Directory Access Protocol) : Protocole client-serveur pour l'accès à un service de répertoire.

Protocole TCP/IP (Transmission Control Protocol/Internet Protocol) : Ensemble de protocoles de communication prenant en charge les fonctions de connectivité d'égal à égal sur les réseaux locaux (LAN) et longue distance (WAN).

public : Une classe ou une interface publique est visible partout. Une méthode ou une variable publique est visible partout où sa classe l'est.

serveur : (1) Un serveur MQSeries Everyplace est une unité sur laquelle est configuré un gestionnaire de canal MQSeries Everyplace. (2) Un serveur MQSeries est un gestionnaire de files d'attente qui fournit des services de mise en file d'attente de messages aux applications client exécutées sur un poste de travail éloigné. (3) Plus généralement, un serveur est un programme qui répond aux demandes d'informations, dans le modèle spécifique client/serveur de flux d'informations entre deux programmes. (3) Ordinateur sur lequel est exécuté un programme serveur.

Service JNDI (Java Naming and Directory Service) : API spécifiée en langage de programmation Java. Elle fournit des fonctions d'annuaire et de répertoire aux applications écrites dans le langage de programmation Java.

servlet : Programme Java conçu pour être exécuté exclusivement sur un serveur Web.

sous-classe : Classe en étendant une autre. La sous-classe hérite des méthodes publiques et protégées et des variables de sa superclasse.

superclasse : Classe étendue par une autre. Les méthodes publiques et protégées et les variables de la superclasse sont accessibles à la sous-classe.

Web : Voir World Wide Web.

World Wide Web (Web) : Le Web est un service d'Internet, basé sur un ensemble commun de protocoles, qui permet à un serveur spécialement configuré de diffuser des documents sur Internet selon une méthode standard.

Bibliographie

Documentation connexe :

- *MQSeries Everyplace for Multiplatforms - Lisez-moi d'abord*, GC11-1702
- *MQSeries Everyplace for Multiplatform Programming Reference*, SC34-5846
- *MQSeries Everyplace for Multiplatforms Programming Guide*, SC34-5845
- *MQSeries Everyplace for Multiplatforms Native Client Information*, GC34-5883
- *Introduction à MQSeries*, GC11-1091-00
- *MQSeries pour Windows NT - Mise en route - Version 5.1*, GC11-1509-00

Index

A

adaptateurs, MQSeries
 Everyplace 47
adaptateurs MQSeries
 Everyplace 47
administration, messages 43
administration avec MQSeries
 Everyplace 42
applications, chargement 63
applications MQSeries
 Everyplace 14

C

canal dynamique 9
canaux 9
canaux, client 17
canaux, dynamiques 46
canaux client 17
canaux client-serveur 9
canaux dynamique 17
canaux dynamiques 17, 46
canaux entre homologues 9
chargement des applications 63
chiffrement 50
classes, MQSeries Everyplace 62
classes MQSeries Everyplace 62
client, exigences 14
client MQSeries 8
communication 60
compatibilité avec MQSeries 76
compression 50
concepts du produit 17
configuration 59
configurations, exemples 11
configurations exemples 11
connaissances requise vii
connexion, client-serveur 61
connexion client-serveur 61
connexion d'égal à égal 61
connexion entre homologues 61
conversion de message 74

D

description 1

E

enregistrement automatique 56
entités, authentifiables 55
entités authentifiables 55
environnements, logiciels 3

environnements logiciels 3
exigences des clients 14

F

famille MQSeries 7
files d'attente, de pont
 MQSeries 28, 29
files d'attente, éloignées 26
files d'attente, locales 25
files d'attente, MQSeries
 Everyplace 25
files d'attente de pont MQSeries 28,
 29
files d'attente de serveur local 27
files d'attente de stockage et de
 réacheminement 27
files d'attente éloignées 26
files d'attente locales 25
files d'attente MQSeries
 Everyplace 25
fonction Transformation 74
format des données de vidage 24

G

généralités 1
gestion de connexion commutée 48
gestionnaire de canal 60
gestionnaire de files d'attente 8, 9,
 60
gestionnaires de files d'attente,
 MQSeries Everyplace 32
gestionnaires de files d'attente,
 opérations 39
gestionnaires de files d'attente
 MQSeries Everyplace 32

I

interface, de sécurité 58
interface avec MQSeries 65
interface de sécurité 58
interfaces, de programmation 79
interfaces de programmation 79

J

journal des événements 48

L

lecteur du manuel vii
légal, remarques 89
livraison garantie unique des
 messages 77

M

marques 91
message, livraison garantie
 unique 77
messagerie, asynchrone 49
messagerie, synchrone 50
messagerie asynchrone 49
messagerie diffuse viii, 8
messagerie distribuée viii, 8
messagerie hôte viii, 8
messagerie MQSeries 7
messagerie synchrone 50
messages d'administration 43
mini-certificats 55
MQeAttribute 54
MQeMTrustAttribute 54
MQSeries, compatibilité 76
MQSeries, interface 65
MQSeries Everyplace,
 administration 42
MQSeries Integrator viii, 7
MQSeries Workflow viii, 7

O

objet pont 68
objet programme d'écoute 71, 72
objet proxy du gestionnaire de files
 d'attente 68
objets, messages 18
objets, MQSeries Everyplace 18
objets messages 18
objets MQSeries Everyplace 18
opérations des gestionnaires de files
 d'attente 39

P

passerelles MQSeries Everyplace 17
personnalisation 59
pont, objet 67
pont MQSeries 9, 65
possibilités 13
préalables 3
présentation vii
programme d'écoute de canal 60

R

registre 55
registre, privé 56
registre, public 57
registre Everyplace MQSeries 17

- registre MQSeries Everyplace 17, 55
- registre privé 56
- registre public 57
- règles, MQSeries Everyplace 59
- règles d'attribut 59
- règles de file d'attente 60
- règles du gestionnaire de files d'attente 60
- règles du pont MQSeries 59
- règles MQSeries Everyplace 59
- remarques légales 89
- réplication de certificat 57
- réplication de certificats 57
- réseaux, MQSeries 65
- réseaux, MQSeries Everyplace 65
- réseaux MQSeries 65
- réseaux MQSeries Everyplace 49, 65

S

- sécurité, au niveau des files d'attente 52
- sécurité, au niveau message 53
- sécurité, locale 52
- sécurité, MQSeries Everyplace 50
- sécurité au niveau des files d'attente 52
- sécurité au niveau message 53
- sécurité locale 52
- sécurité MQSeries Everyplace 50
- serveur local, MQSeries Everyplace 27
- serveur MQSeries 8
- service d'émission de mini-certificats 57
- SPI 79
- stockage et réacheminement, files d'attente 27
- styles de connexions 60
- styles de connexions, multiples 62
- styles de connexions multiples 62
- surveillance 46
- systèmes d'exploitation, pris en charge 3
- systèmes d'exploitation pris en charge 3
- systèmes d'exploitation requis 3

T

- terminologie viii
- trace MQSeries Everyplace 48

U

- unités MQSeries Everyplace 17



GC11-1701-02

