

Symantec pcAnywhere™

Guide d'automatisation OLE

Symantec pcAnywhere™ - Guide d'automatisation OLE

Le logiciel décrit dans ce guide est fourni dans le cadre d'un contrat de licence et ne peut être utilisé qu'en conformité avec les termes de ce contrat.

Documentation version 11.5

Copyright

Copyright © 2004 Symantec Corporation.

Tous droits réservés.

Toute documentation technique fournie par Symantec Corporation est soumise à copyright et reste la propriété de Symantec Corporation.

ABSENCE DE GARANTIE. Cette documentation technique vous est fournie en l'état et Symantec Corporation n'offre aucune garantie quant à la validité des informations qu'elle contient. Toute utilisation de la documentation technique et des informations qu'elle contient relève de la seule responsabilité de l'utilisateur. La documentation peut inclure des erreurs techniques ou typographiques, ou d'autres imprécisions. Symantec se réserve le droit d'y apporter des modifications sans avis préalable.

Toute reproduction, même partielle, de ce document est interdite sans l'autorisation écrite expresse de Symantec Corporation, 20330 Stevens Creek Blvd, Cupertino, CA 95014, Etats-Unis.

Marques commerciales

Symantec, le logo Symantec et pcAnywhere sont des marques déposées par Symantec aux Etats-Unis. Microsoft, Windows, Windows NT, MS-DOS et le logo Windows sont des marques déposées ou commerciales de Microsoft Corporation aux Etats-Unis et/ou dans d'autres pays.

Les autres marques et noms de produits mentionnés dans ce guide peuvent être des marques commerciales ou déposées de leurs sociétés respectives et sont reconnues par le présent document.

Imprimé en Irlande.

10 9 8 7 6 5 4 3 2 1

Table des matières

Chapitre 1	Utilisation de l'automatisation OLE avec Symantec pcAnywhere	
	A propos de l'automatisation OLE	7
	A propos du serveur d'automatisation pcAnywhere	8
	Ce que vous pouvez faire avec le serveur d'automatisation pcAnywhere	8
	Avant de commencer	9
	Inscription automatique du moteur Maître	9
	Inscription manuelle du moteur Maître	10
	Accès au serveur d'automatisation pcAnywhere	10
	Accès au serveur d'automatisation pcAnywhere avec Visual Basic	10
	Accès au serveur d'automatisation pcAnywhere avec Visual C++	11
	Démarrage des objets OLE Elève et Maître	13
	Pour plus d'informations	14
Chapitre 2	Définitions d'objets Visual Basic	
	A propos des objets Visual Basic	15
	Méthodes CRemoteDataManager	17
	CurrentDirectory()	17
	ChangeDirectory(nouveau répertoire)	17
	FindFirst(schéma, nom)	18
	FindNext(nom)	18
	RetrieveObject(nom, mode d'accès, mot de passe)	19
	RetrieveObjectEx(nom, mode d'accès, mot de passe)	20
	CreateObject(nom)	20
	CreateObjectEx(nom)	21
	DeleteObject(nom, mot de passe)	21
	Propriétés CRemoteData	22
	Propriétés du type de connexion	25
	Propriétés de numérotation	27
	Propriétés de périphérique COM	28
	Propriétés de périphérique NetBIOS	31
	Propriétés de périphérique RNIS via CAPI 2.0	32
	Détails de périphérique réseau (TCP/IP, SPX) pour les passerelles	32

Objet CRemoteDataEx	33
Exemple de code Visual Basic pour la fonction Maître	34
Méthodes CHostDataManager	36
CurrentDirectory()	36
FindFirst(schéma, nom)	37
FindNext(nom)	37
RetrieveObject(nom, mode d'accès, mot de passe)	38
RetrieveObjectEx(nom, mode d'accès, mot de passe)	39
CreateObject(nom)	39
CreateObjectEx(nom)	40
DeleteObject(nom, mot de passe)	40
Launch(nom)	41
Propriétés CHostData	41
Propriétés du type de connexion	44
Méthode AssignConnection(type de connexion)	46
Méthode UnassignConnection(type de connexion)	47
Propriétés de numérotation	47
Propriétés de périphérique COM	48
Propriétés de périphérique NetBIOS	51
Propriétés de périphérique RNIS via CAPI 2.0	52
Détails de périphérique réseau (TCP/IP, SPX) pour les passerelles	52
Objet CHostDataEx	53
Exemple de code Visual Basic pour la fonction Elève.	60
Fonctions awrem32	62
awConnect(nom de fichier)	62
awDisconnect()	62
FileXferFromHost(fichier Elève, fichier Maître)	63
FileXferToHost(fichier Elève, fichier Maître)	63
CreateFolderOnHost(nom de répertoire)	64
ExecuteHostFile(nom de fichier)	64
GetError()	65
ConnectionStatus()	65

Chapitre 3 Définitions d'objets Visual C++

A propos des objets Visual C++	67
Méthodes CRemoteDataManager	69
BSTR CurrentDirectory();	69
BOOL ChangeDirectory(LPCTSTR lpszNewDirectory);	69
BOOL FindFirst(LPCTSTR lpszPattern, BSTR FAR* pbstrFullQualName);	70
BOOL FindNext(BSTR FAR* pbstrFullQualName);	70
LPDISPATCH RetrieveObject(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);	71
LPDISPATCH RetrieveObjectEx(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);	72
LPDISPATCH CreateObject(LPCTSTR lpszFQName);	73
LPDISPATCH CreateObjectEx(LPCTSTR lpszFQName);	73
BOOL DeleteObject(LPCTSTR lpszFQName, LPCTSTR lpszPassword);	74
BOOL Launch(LPCTSTR lpszFQName);	74
Objet CRemoteData	75
Méthodes Get et Set (obtenir et définir)	75
Méthodes de détail d'objet Maître	77
Méthodes d'objet Maître	81
Objet CRemoteDataEx	84
Exemple de code Visual C++ pour la fonction Maître	84
Méthodes CHostDataManager	85
BSTR CurrentDirectory();	85
BOOL ChangeDirectory(LPCTSTR lpszNewDirectory);	86
BOOL FindFirst(LPCTSTR lpszPattern, BSTR FAR* pbstrFullQualName);	86
BOOL FindNext(BSTR FAR* pbstrFullQualName);	87
LPDISPATCH RetrieveObject(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);	87
LPDISPATCH RetrieveObjectEx(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);	88
LPDISPATCH CreateObject(LPCTSTR lpszName);	89
LPDISPATCH CreateObjectEx(LPCTSTR lpszName);	89
BOOL DeleteObject(LPCTSTR lpszFQName, LPCTSTR lpszPassword);	90
BOOL Launch(LPCTSTR lpszFQName);	91
Objet CHostData	91
Méthodes Get et Set (obtenir et définir)	91
Méthodes de détail d'objet Elève	92
Méthodes d'objet Elève	97

Objet CHostDataEx 102

 Exemple de code Visual C++ pour la fonction Elève 104

Fonctions awrem32 105

 boolean awConnect(BSTR FileName); 105

 boolean awDisconnect(); 105

 boolean FileXferFromHost(BSTR HostFile, BSTR RemoteFile); 106

 boolean FileXferToHost(BSTR HostFile, BSTR RemoteFile); 106

 boolean CreateFolderOnHost(Nom de dossier BSTR); 107

 boolean ExecuteHostFile(BSTR FileName); 107

 BSTR GetError(); 108

 short ConnectionStatus(); 108

Index

Solutions de service et de support

Utilisation de l'automatisation OLE avec Symantec pcAnywhere

Ce chapitre traite des sujets suivants :

- [A propos de l'automatisation OLE](#)
- [A propos du serveur d'automatisation pcAnywhere](#)
- [Ce que vous pouvez faire avec le serveur d'automatisation pcAnywhere](#)
- [Avant de commencer](#)
- [Pour plus d'informations](#)

A propos de l'automatisation OLE

L'automatisation OLE est une technologie qui permet de créer une application externe ou un autre outil de développement (par exemple un script ou une macro) capable de contrôler et d'automatiser toute fonction exposée d'une application.

L'automatisation OLE est constituée des composants suivants :

- Un serveur d'automatisation OLE : application ou composant logiciel qui expose sa fonction afin d'être accessible ou de pouvoir être contrôlé par d'autres applications ou d'autres outils de développement
Le serveur d'automatisation pcAnywhere est un exemple de serveur d'automatisation OLE.

Se reporter à « [A propos du serveur d'automatisation pcAnywhere](#) » à la page 8.

- Un contrôleur d'automatisation OLE : application ou outil de développement pouvant atteindre et contrôler les composants exposés par le serveur d'automatisation OLE. Vous pouvez utiliser tout langage de programmation prenant en charge l'automatisation OLE. Les deux langages de programmation les plus courants sont Microsoft Visual Basic et Microsoft Visual C++.

Une application externe accède à un serveur d'automatisation en se connectant au serveur, puis en demandant l'accès à une ou plusieurs de ses interfaces publiées. Une interface est un point d'entrée permettant d'accéder à une ou plusieurs méthodes ou propriétés rattachées. Une fois que l'application a obtenu une interface vers le serveur, elle peut appeler toute méthode d'interface comme si celle-ci faisait partie de l'application externe.

A propos du serveur d'automatisation pcAnywhere

Le serveur d'automatisation pcAnywhere permet aux applications externes de gérer les fichiers pcAnywhere d'informations Elève, Maître et appelant afin d'automatiser les tâches de téléintervention et de transfert de fichiers. Le serveur d'automatisation pcAnywhere fonctionne comme un substitut programmable de l'interface utilisateur de Symantec pcAnywhere et reprend dans une large mesure son comportement par défaut.

Par exemple, lorsque vous créez un objet Elève dans pcAnywhere, le premier modem TAPI disponible est attribué par défaut. De même, si vous créez un objet Elève à l'aide du serveur d'automatisation pcAnywhere, puis que vous énumérez la liste des connexions attribuées, le premier modem TAPI disponible est déjà attribué.

Ce que vous pouvez faire avec le serveur d'automatisation pcAnywhere

Le serveur d'automatisation pcAnywhere vous permet d'automatiser diverses tâches administratives et de productivité. Vous pouvez par exemple effectuer les opérations suivantes :

- Distribution et installation automatiques de mises à jour logicielles sur plusieurs ordinateurs du réseau
- Programmation de transferts automatiques de fichiers entre ordinateurs à des fins d'audit ou d'archivage
- Ajout ou suppression automatique d'un nom de la liste des appelants autorisés sur chaque Elève pcAnywhere du réseau

Ce document comporte plusieurs exemples, écrits à la fois en Visual Basic et en Visual C++, afin d'illustrer la manière de se connecter au serveur d'automatisation pcAnywhere et de l'utiliser.

Se reporter à « [Définitions d'objets Visual Basic](#) » à la page 15.

Se reporter à « [Définitions d'objets Visual C++](#) » à la page 67.

Avant de commencer

Au cours d'une connexion au serveur d'automatisation pcAnywhere et à ses interfaces, les paramètres d'identification, appelés GUID (Globally Unique Identifiers - identificateurs globalement uniques), sont transmis aux fonctions API de la bibliothèque d'automatisation. Un GUID distinct est attribué au serveur d'automatisation pcAnywhere et à chaque interface exposée. Ces GUID doivent être présents dans le registre système pour connecter une application externe au serveur d'automatisation pcAnywhere et à ses interfaces.

Si vous exécutez l'application externe sur un ordinateur sur lequel Symantec pcAnywhere est installé, vous pouvez inscrire automatiquement les entrées GUID. Dans le cas contraire, vous devez enregistrer les entrées GUID manuellement.

Se reporter à « [Inscription automatique du moteur Maître](#) » à la page 9.

Se reporter à « [Inscription manuelle du moteur Maître](#) » à la page 10.

Inscription automatique du moteur Maître

Avant de vous connecter à un autre ordinateur pour la première fois en utilisant votre client OLE, vous devez inscrire automatiquement le moteur Maître. Cette opération peut s'effectuer automatiquement lorsque vous démarrez un objet Maître dans Symantec pcAnywhere.

Pour inscrire automatiquement le moteur Maître

- 1 Pour ouvrir Symantec pcAnywhere, effectuez l'une des opérations suivantes :
 - Sur le bureau, cliquez deux fois sur l'icône du programme Symantec pcAnywhere.
 - Dans la barre des tâches de Windows, cliquez sur **Démarrer > Programmes > Symantec pcAnywhere**.
- 2 Dans la fenêtre du Gestionnaire pcAnywhere, cliquez sur **Maître**.
- 3 Cliquez deux fois sur un élément de connexion Maître.
Cette procédure inscrit le moteur Maître. Il n'est pas nécessaire de terminer la connexion.

Inscription manuelle du moteur Maître

Si pcAnywhere n'est pas installé sur l'ordinateur sur lequel vous exécutez l'application externe, vous devez inscrire manuellement les GUID en exécutant le fichier exécutable (winawsvr.exe) du serveur d'automatisation pcAnywhere. Il suffit de lancer l'exécutable une seule fois pour ajouter les GUID au registre. Le fichier winawsvr.exe se trouve dans le répertoire de données par défaut de Symantec pcAnywhere.

Accès au serveur d'automatisation pcAnywhere

Vous pouvez accéder au serveur d'automatisation de pcAnywhere en utilisant toute plate-forme prenant en charge l'automatisation OLE. Les deux plates-formes les plus répandues qui prennent en charge l'automatisation OLE sont Visual Basic et Visual C++.

Les principes de codage de ces deux plates-formes sont similaires, bien que dans l'environnement Visual Basic une part importante des tâches de bas niveau est exécutée en arrière-plan par l'exécutable Visual Basic.

Accès au serveur d'automatisation pcAnywhere avec Visual Basic

Le langage de programmation Visual Basic intègre la prise en charge de l'interaction avec les serveurs d'automatisation OLE tels que le serveur d'automatisation pcAnywhere. Vous pouvez créer un projet Exe standard puis, dans chaque méthode, entrer le code pour accéder au serveur d'automatisation pcAnywhere. Visual Basic prend les appels de méthode de haut niveau dans les fichiers source et les développe de façon interne dans les appels de méthode de bas niveau correspondants de l'automatisation OLE.

Se reporter à « [Définitions d'objets Visual Basic](#) » à la page 15.

Pour accéder au serveur d'automatisation pcAnywhere avec Visual Basic

- 1 Ajoutez une paire de variables objet pour chaque objet pcAnywhere auquel accéder.
Par exemple, pour travailler avec des objets Maître, dimensionnez un objet RemoteDataManager et RemoteDataObject.
- 2 Utilisez RemoteDataManager pour le rattachement au gestionnaire de données de l'objet Maître.
Par exemple, appelez la méthode CreateObject avec WINAWSVR.REMOTEDATAMANAGER comme paramètre.
Visual Basic utilise le paramètre textuel pour repérer l'identificateur du gestionnaire dans la base de registres et renvoie l'interface vers ce gestionnaire.

- 3 Une fois qu'il existe un objet gestionnaire de données valide, utilisez-le pour effectuer les opérations suivantes :
 - Déterminer le répertoire en cours.
 - Accéder à un autre répertoire
 - Enumérer les fichiers objet de données associés dans le répertoire en cours
 - Créer, extraire ou supprimer un fichier objet de données
- 4 Après la création ou la récupération d'un objet de données, vous pouvez obtenir (Get) ou définir (Set) ses propriétés.
 La syntaxe de Visual Basic n'utilise pas de nom de propriété pour différencier l'obtention et la définition de sa valeur. C'est la position de la propriété par rapport à l'opérateur d'affectation qui détermine si l'appel de méthode sous-jacent est un Get ou un Set.
 Les exemples suivants illustrent un Get et un Set :
 - Pour obtenir la valeur du numéro de téléphone d'un objet, placez le nom de la propriété à droite de l'opérateur d'affectation.
 Par exemple, `s = RemoteData.PhoneNumber()`, où `s` est une variable de type chaîne.
 - Pour configurer le numéro de téléphone, placez le nom de la propriété à gauche de l'opérateur d'affectation.
 Par exemple, `RemoteData.PhoneNumber = « 55 55 12 12 »`.

Accès au serveur d'automatisation pcAnywhere avec Visual C++

Le serveur d'automatisation pcAnywhere utilise des bibliothèques de type pour exposer les informations concernant ses interfaces et ses méthodes vers les clients d'automatisation, qui sont écrits en Visual C++. Ces bibliothèques de type utilisent des MFC (Microsoft Foundation Classes), qui peuvent être importées dans votre application via l'assistant ClassWizard de Visual C++.

Les classes du gestionnaire de données fournies dans les bibliothèques de type contiennent la fonctionnalité nécessaire pour obtenir une interface vers le serveur d'automatisation pcAnywhere et effectuer des opérations de haut niveau sur le type d'objet associé à l'interface.

Utiliser l'objet gestionnaire de données pour effectuer les opérations suivantes :

- Déterminer ou modifier le répertoire en cours.
- Enumérer les fichiers objet de données du répertoire en cours.
- Créer, extraire ou supprimer un objet nommé.

Une fois créé ou récupéré, un objet utilise la classe d'objet de données associée pour examiner ou modifier ses propriétés exposées. La plupart de ces propriétés sont exposées à travers une paire de méthodes qui commencent par le mot Get ou Set. Par exemple, un utilisateur appelle la méthode GetPhoneNumber pour examiner la propriété de numéro de téléphone en cours de l'objet et appelle SetPhoneNumber pour la définir.

Se reporter à « Définitions d'objets Visual C++ » à la page 67.

Importation et affichage des classes

Le serveur d'automatisation pcAnywhere utilise les bibliothèques de type suivantes :

- Winawsvr.tlb : contient les informations nécessaires pour se connecter au serveur d'automatisation pcAnywhere et accéder à ses interfaces.
- Awrem32.tlb : contient les informations nécessaires au contrôle des connexions pcAnywhere.

Importation et affichage des classes

Les procédures suivantes expliquent comment importer les bibliothèques de type du serveur d'automatisation pcAnywhere et afficher les classes ayant été ajoutées à votre application.

Pour importer des classes

- 1 Créez une application MFC dans Visual C++.
- 2 Dans le menu Affichage, cliquez sur **Assistant de classe**.
- 3 Dans la boîte de dialogue Assistant de classe, cliquez sur **Ajouter une classe**, puis cliquez sur **Depuis une bibliothèque de type**.
- 4 Cliquez deux fois sur **winawsvr.tlb**.
- 5 Dans la boîte de dialogue Confirmer Classes, cliquez sur **OK** pour importer toutes les définitions de classe.
- 6 Dans la boîte de dialogue Assistant de classe, cliquez sur **Ajouter une classe**, puis cliquez sur **Depuis une bibliothèque de type**.
- 7 Cliquez deux fois sur **awrem32.tlb**.
- 8 Dans la boîte de dialogue Confirmer Classes, cliquez sur **OK** pour importer toutes les définitions de classe.

- 9 Dans la boîte de dialogue Assistant de classe, cliquez sur **OK** pour terminer l'importation.
 Les classes sont ajoutées à l'application. Elles permettent de manipuler des objets et de gérer des connexions.
 L'importation des définitions de classe à partir des bibliothèques de type ajoute également des fichiers de prise en charge à l'application. Ces fichiers contiennent les définitions de classe et le code source de mise en oeuvre du serveur d'automatisation pcAnywhere.
 Se reporter à « [Affichage des définitions de classe et des fichiers de mise en oeuvre](#) » à la page 13.

Pour afficher les classes ajoutées

- 1 En Visual C++, ouvrez votre application MFC.
- 2 Dans la fenêtre de l'espace de travail, cliquez sur l'onglet **ClassView**.

Affichage des définitions de classe et des fichiers de mise en oeuvre

Lorsque vous importez les bibliothèques de type du serveur d'automatisation pcAnywhere dans votre application, les fichiers suivants sont ajoutés :

- Winawsvr.h
- Winawsvr.cpp
- Awrem32.h
- Awrem32.cpp

Ces fichiers contiennent les définitions de classe et le code source de mise en oeuvre du serveur d'automatisation pcAnywhere. Il est inutile d'éditer ces fichiers ; cependant, chaque fichier source de l'application contenant des appels des méthodes du serveur d'automatisation pcAnywhere doit inclure Winawsvr.h.

Pour afficher les définitions de classe et les fichiers de mise en oeuvre

- 1 En Visual C++, ouvrez votre application MFC.
- 2 Dans la fenêtre de l'espace de travail, cliquez sur l'onglet **FileView**.

Démarrage des objets OLE Elève et Maître

Symantec pcAnywhere nécessite que tous les objets Elève et Maître soient stockés dans le répertoire de données par défaut. Avant de démarrer un objet Elève ou Maître que vous avez créé avec OLE, vérifiez qu'il se trouve dans le répertoire de données par défaut de pcAnywhere.

Pour plus d'informations

Pour plus d'informations sur l'automatisation OLE, reportez-vous aux documents de référence suivants (en anglais) :

- Blaszczyk, Mike. 1997. *Professional MFC with Visual C++ 5*. Birmingham, RU : Wrox Press.
- Box, Don. 1998. *Essential COM*. Reading, Mass.: Addison-Wesley.
- Brockschmidt, Kraig. 1995. *Inside OLE, Second Edition*. Redmond, Wash.: Microsoft Press.
- Horton, Ivor. 1997. *Beginning MFC Programming*. Birmingham, RU : Wrox Press.
- Rogerson, Dale. 1997. *Inside COM*. Redmond, Wash.: Microsoft Press.
- Templeman, Julian. 1997. *Beginning MFC COM Programming*. Birmingham, RU : Wrox Press.

Définitions d'objets Visual Basic

Ce chapitre traite des sujets suivants :

- [A propos des objets Visual Basic](#)
- [Méthodes CRemoteDataManager](#)
- [Propriétés CRemoteData](#)
- [Objet CRemoteDataEx](#)
- [Méthodes CHostDataManager](#)
- [Propriétés CHostData](#)
- [Objet CHostDataEx](#)
- [Fonctions awrem32](#)

A propos des objets Visual Basic

Le serveur d'automatisation pcAnywhere offre les composants suivants pour la prise en charge de l'automatisation OLE :

- Winawsvr : contient les informations nécessaires pour se connecter au serveur d'automatisation pcAnywhere et accéder à ses interfaces.
- Awrem32 : contient les informations nécessaires au contrôle des connexions pcAnywhere.

Le [Tableau 2-1](#) décrit les objets qui composent Winawsvr.

Tableau 2-1 Objets Winawsvr

Objet	Description	Référence
CRemoteDataManager	Fournit les méthodes pour créer, ouvrir, modifier, enregistrer et supprimer les objets CRemoteData.	Se reporter à « Méthodes CRemoteDataManager » à la page 17.
CRemoteData	Définit les paramètres permettant d'accéder à la fonction Maître de pcAnywhere et de la contrôler.	Se reporter à « Propriétés CRemoteData » à la page 22. Se reporter à « Objet CRemoteDataEx » à la page 33.
CHostDataManager	Fournit les méthodes pour créer, ouvrir, modifier, enregistrer et supprimer les objets ChostData.	Se reporter à « Méthodes CHostDataManager » à la page 36.
CHostData	Définit les paramètres permettant d'accéder à la fonction Elève de pcAnywhere et de la contrôler.	Se reporter à « Propriétés CHostData » à la page 41. Se reporter à « Objet CHostDataEx » à la page 53.

Awrem32 possède un objet, constitué de huit interfaces, destiné à prendre en charge les sessions de téléintervention et de transfert de fichiers.

Se reporter à « [Fonctions awrem32](#) » à la page 62.

Certaines fonctions, notamment les passerelles, ne sont plus prises en charge. Cependant, les définitions d'objets sont utilisables avec les versions antérieures de pcAnywhere.

Pour les fonctions nécessitant des mots de passe, les valeurs des mots de passe peuvent être définies mais non récupérées, pour des raisons de sécurité.

Méthodes CRemoteDataManager

Les méthodes CRemoteDataManager fournissent les paramètres et renvoient les valeurs permettant d'accéder aux objets CRemoteData et de les contrôler.

CurrentDirectory()

Renvoie le chemin d'accès complet du dossier actuel dans lequel les objets Maître pcAnywhere sont stockés.

Le [Tableau 2-2](#) définit la valeur renvoyée.

Tableau 2-2 CurrentDirectory() - Valeur renvoyée

Valeur renvoyée	Description
Chaîne	Chemin d'accès complet du répertoire de données actuel de pcAnywhere

ChangeDirectory(nouveau répertoire)

Change le dossier actuel dans lequel les objets Maître pcAnywhere sont stockés.

Le [Tableau 2-3](#) définit le paramètre.

Tableau 2-3 ChangeDirectory - Paramètre

Paramètre	Description
Nouveau répertoire	Nom d'un répertoire existant

Le [Tableau 2-4](#) définit la valeur renvoyée.

Tableau 2-4 ChangeDirectory - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si réussi

FindFirst(schéma, nom)

Recherche le premier fichier d'objet Maître pcAnywhere (*.CHF) dans le dossier en cours, sur la base du schéma de nom de fichier spécifié.

Le [Tableau 2-5](#) définit les paramètres.

Tableau 2-5 FindFirst - Paramètres

Paramètre	Description
Schéma comme chaîne	Schéma de nom de fichier pour filtrer les fichiers d'objet (un astérisque [*] trouve tous les fichiers dans le répertoire en cours)
Nom comme chaîne	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié

Le [Tableau 2-6](#) définit la valeur renvoyée.

Tableau 2-6 FindFirst - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si un fichier d'objet Maître correspondant au schéma spécifié est trouvé. Le chemin d'accès complet du fichier est stocké dans Nom.

FindNext(nom)

Après l'appel réussi de FindFirst() pour obtenir le nom d'un fichier d'objet Maître dans le dossier en cours, FindNext() peut être appelé pour trouver le fichier suivant correspondant au schéma.

Le [Tableau 2-7](#) définit le paramètre FindNext.

Tableau 2-7 FindNext - Paramètre

Paramètre	Description
Nom comme chaîne	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié dans l'appel initial de FindFirst().

Le [Tableau 2-8](#) définit la valeur renvoyée.

Tableau 2-8 FindNext - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si un autre fichier d'objet Maître correspondant au schéma spécifié dans l'appel de FindFirst() est trouvé. Le chemin d'accès complet du fichier est stocké dans Nom.

RetrieveObject(nom, mode d'accès, mot de passe)

Récupère un objet CRemoteData par nom de fichier.

Le [Tableau 2-9](#) définit les paramètres RetrieveObject.

Tableau 2-9 RetrieveObject - Paramètres

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Maître à charger
Mode d'accès comme valeur entière	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none">■ 0 = Non spécifié■ 1 = Afficher uniquement■ 2 = Afficher et modifier■ 3 = Exécuter
Mot de passe comme chaîne	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 2-10](#) définit la valeur renvoyée.

Tableau 2-10 RetrieveObject - Valeur renvoyée

Valeur renvoyée	Description
Objet	Objet CRemoteData du fichier spécifié

RetrieveObjectEx(nom, mode d'accès, mot de passe)

Récupère un objet CRemoteDataEx par nom de fichier.

Le [Tableau 2-11](#) définit les paramètres RetrieveObjectEx.

Tableau 2-11 RetrieveObjectEx - Paramètres

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Maître à charger.
Mode d'accès comme valeur entière	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none"> ■ 0 = Non spécifié ■ 1 = Afficher uniquement ■ 2 = Afficher et modifier ■ 3 = Exécuter
Mot de passe comme chaîne	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 2-12](#) définit la valeur renvoyée.

Tableau 2-12 RetrieveObjectEx - Valeur renvoyée

Valeur renvoyée	Description
Objet	Objet CRemoteDataEx du fichier spécifié.

CreateObject(nom)

Crée un objet CRemoteData et renvoie un pointeur LPDISPATCH qui le désigne.

Le [Tableau 2-13](#) définit le paramètre CreateObject.

Tableau 2-13 CreateObject - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Maître pour le nouvel objet.

Le [Tableau 2-14](#) définit la valeur renvoyée.

Tableau 2-14 CreateObject - Valeur renvoyée

Valeur renvoyée	Description
Objet	CRemoteData

CreateObjectEx(nom)

Crée un objet CRemoteDataEx et renvoie un pointeur LPDISPATCH qui le désigne.

Le [Tableau 2-15](#) définit le paramètre CreateObjectEx.

Tableau 2-15 CreateObjectEx - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Maître pour le nouvel objet.

Le [Tableau 2-16](#) définit la valeur renvoyée.

Tableau 2-16 CreateObjectEx - Valeur renvoyée

Valeur renvoyée	Description
Objet	CRemoteDataEx

DeleteObject(nom, mot de passe)

Supprime un fichier d'objet Maître.

Le [Tableau 2-17](#) définit les paramètres DeleteObject.

Tableau 2-17 DeleteObject - Paramètres

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Maître de l'objet à supprimer.
Mot de passe comme chaîne	Mot de passe de l'objet

Le [Tableau 2-18](#) définit la valeur renvoyée.

Tableau 2-18 DeleteObject - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si l'objet est supprimé

Propriétés CRemoteData

Le [Tableau 2-19](#) définit les propriétés et paramètres disponibles pour l'objet CRemoteData. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-19 CRemoteData - Propriétés et paramètres

Propriété	Paramètre	Description
<CRemoteData>.ComputerName(chaine)	Chaîne	Définit le nom ou l'adresse IP de l'ordinateur Elève.
<CRemoteData>.PhoneNumber(chaine)	Chaîne	Définit le numéro de téléphone de l'ordinateur Elève.
<CRemoteData>.UseDialingProperties(Bool)	Bool	Définit les propriétés de numérotation du système.
<CRemoteData>.RedialCount(entier)	Entier	Définit le nombre de tentatives de rappel avant l'annulation de l'appel.
<CRemoteData>.RedialDelay(entier)	Entier	Définit le nombre de secondes d'attente entre les tentatives de rappel.
<CRemoteData>.AutoLoginName(chaine)	Chaîne	Définit le nom de l'utilisateur pour la connexion automatique. Pour plus d'informations sur l'utilisation de la connexion aux domaines, reportez-vous à la section « Objet CRemoteDataEx » à la page 33.
<CRemoteData>.AutoLoginPassword(chaine)	Chaîne	Définit le mot de passe pour la connexion automatique dans l'objet Maître. Pour des raisons de sécurité, le serveur d'automatisation pcAnywhere ne permet pas de lire la valeur du mot de passe. Aucune valeur de mot de passe n'est renvoyée.
<CRemoteData>.Password(chaine)	Chaîne	Définit le mot de passe sur l'objet Maître, à utiliser avec les méthodes ExecuteProtection, ReadProtection et WriteProtection. Pour des raisons de sécurité, le serveur d'automatisation pcAnywhere ne permet pas de lire la valeur du mot de passe. Aucune valeur de mot de passe n'est renvoyée.
<CRemoteData>.ExecuteProtection (Bool)	Bool	Définit la nécessité d'un mot de passe pour exécuter l'objet. Définir par mot de passe.

Tableau 2-19 CRemoteData - Propriétés et paramètres

Propriété	Paramètre	Description
<CRemoteData>.ReadProtection(Bool)	Bool	Définit la nécessité d'un mot de passe pour afficher les propriétés de l'objet Maître. Définir par mot de passe.
<CRemoteData>.WriteProtection(Bool)	Bool	Définit la nécessité d'un mot de passe pour enregistrer les modification de l'objet Maître. Définir par mot de passe.
<CRemoteData>.LogSession (Bool)	Bool	Active et désactive la consignation des sessions
<CRemoteData>.RecordFile(chaine)	Chaîne	Définit le nom et le chemin d'accès intégralement qualifié du fichier d'enregistrement de la session active.
<CRemoteData>.RecordSession(Bool)	Bool	Active et désactive la consignation automatique de session
<CRemoteData>.ReadObject(chaine)	Chaîne	Définit le mot de passe de l'objet. Utilisez cette propriété pour actualiser la copie locale des données de l'objet Maître.
<CRemoteData>.WriteObject(chaine)	Chaîne	Définit le mot de passe de l'objet. Utilisez cette propriété pour créer l'objet Maître ou pour écrire les changements apportés à celui-ci.

Le [Tableau 2-20](#) définit les propriétés et les valeurs renvoyées pour CRemoteData. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-20 CRemoteData - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CRemoteData>.ComputerName	Chaîne	Renvoie le nom ou l'adresse IP de l'ordinateur Elève
String = <CRemoteData>.PhoneNumber	Chaîne	Renvoie le numéro de téléphone de l'ordinateur Elève
Bool = <CRemoteData>.UseDialingProperties	Bool	Renvoie les propriétés de numérotation du système définies dans l'objet Maître
Integer = <CRemoteData>.RedialCount	Entier	Renvoie le nombre de tentatives d'appel défini dans l'objet Maître
Integer = <CRemoteData>.RedialDelay	Entier	Renvoie le nombre de secondes d'attente entre les tentatives de rappel
String = <CRemoteData>.AutoLoginName	Chaîne	Renvoie le nom de l'utilisateur pour la connexion automatique
Bool = <CRemoteData>.ExecuteProtection	Bool	Renvoie la valeur du paramètre ExecuteProtection
Bool = <CRemoteData>.ReadProtection	Bool	Renvoie la valeur du paramètre ReadProtection
Bool = <CRemoteData>.WriteProtection	Bool	Renvoie la valeur du paramètre WriteProtection
Bool = <CRemoteData>.LogSession	Bool	VRAI si la consignation de session est activée
String = <CRemoteData>.RecordFile	Chaîne	Renvoie le nom et le chemin d'accès intégralement qualifié du fichier d'enregistrement de la session.
Bool = <CRemoteData>.RecordSession	Bool	Renvoie la valeur du paramètre d'enregistrement de session.

Propriétés du type de connexion

Le [Tableau 2-21](#) définit les propriétés et les paramètres du type de connexion. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-21 Type de connexion - Propriétés et paramètres

Propriété	Paramètre	Description
<CRemoteData>.ConnectionType(chaine)	Chaîne	<p>Définit le type de connexion de l'ordinateur Maître. La valeur transmise doit être un type de connexion valide, défini par les fonctions FirstConnectionType() et NextConnectionType().</p> <p>Les types suivants sont des exemples de connexions valides :</p> <ul style="list-style-type: none"> ■ COM1 ■ COM2 ■ COM3 ■ COM4 ■ LPT1 ■ LPT2 ■ LPT3 ■ LPT4 ■ TCP/IP ■ SPX ■ NetBIOS ■ Infrarouge ■ RNIS via CAPI 2.0 ■ Nom du modem (tel qu'il apparaît sur l'ordinateur)

Le [Tableau 2-22](#) définit les propriétés et les valeurs renvoyées du type de connexion. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-22 Type de connexion - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CRemoteData>.ConnectionTypes	Chaîne	Renvoie le type de connexion de l'objet Maître.
Integer = <CRemoteData>.ConnectionType	Entier	<p>Renvoie le nombre de types de connexion disponibles.</p> <p>Les types suivants sont des exemples de connexions valides :</p> <ul style="list-style-type: none"> ■ COM1 ■ COM2 ■ COM3 ■ COM4 ■ LPT1 ■ LPT2 ■ LPT3 ■ LPT4 ■ TCP/IP ■ SPX ■ NetBIOS ■ Infrarouge ■ RNIS via CAPI 2.0 ■ Nom du modem (tel qu'il apparaît sur l'ordinateur)
String = <CRemoteData>.FirstConnectionType	Chaîne	Renvoie le premier type de connexion disponible
String = <CRemoteData>.NextConnectionType	Chaîne	Renvoie le type de connexion disponible suivant. L'appel s'effectue de façon séquentielle pour le nombre de types de connexion défini dans <CRemoteData>.ConnectionType, afin d'énumérer tous les types de connexion.
Bool = <CRemoteData>.FindConnectionType(type de connexion)	Bool	VRAI si le type de connexion indiqué a été trouvé dans la liste des types de connexion disponibles.

Propriétés de numérotation

Le [Tableau 2-23](#) définit les propriétés et les paramètres pour configurer les propriétés de numérotation des connexions par modem. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-23 Propriétés de numérotation - Propriétés et paramètres

Propriété	Paramètre	Description
<CRemoteData>.AreaCode(chaine)	Chaîne	Définit les propriétés de numérotation de code de zone pour les connexions par modem
<CRemoteData>.CountryCode(chaine)	Chaîne	Définit les propriétés de numérotation d'indicatif international pour les connexions par modem

Le [Tableau 2-24](#) définit les propriétés et les valeurs renvoyées pour les propriétés de numérotation par modem. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-24 Propriétés de numérotation - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CRemoteData>.AreaCode	Chaîne	Renvoie les propriétés de numérotation de code de zone
String= <CRemoteData>.CountryCode	Chaîne	Renvoie les propriétés de numérotation d'indicatif international
Integer = <CRemoteData>.CountryCodes	Entier	Renvoie le nombre d'indicatifs internationaux disponibles
String= <CRemoteData>.FirstCountryCode	Chaîne	Renvoie le premier indicatif de pays apparaissant dans la liste du système d'exploitation
String= <CRemoteData>.NextCountryCode	Chaîne	Renvoie l'indicatif international disponible suivant. L'appel s'effectue de façon séquentielle pour le nombre d'indicatifs internationaux défini dans <CRemoteData.CountryCodes>, afin d'énumérer tous les indicatifs internationaux.

Propriétés de périphérique COM

Le [Tableau 2-25](#) décrit les propriétés et les paramètres qui vous permettent de personnaliser les paramètres de port du modem et des autres connexions COM. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-25 Périphérique COM - Propriétés et paramètre

Propriété	Paramètre	Description
<CRemoteData>.ComParity(chaine)	Chaîne	Définit la parité des communications Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ <Aucune>■ Impaire■ Paire■ Marque■ Espace
<CRemoteData>.ComFlowControl(chaine)	Chaîne	Définit le contrôle de flux des types de connexion COM Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ <Aucun>■ XONXOFF■ RTS/CTS■ Les deux
<CRemoteData>.ComStartedBy(chaine)	Chaîne	Définit le paramètre de début des types de connexion COM. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ Toujours connecté■ Détection de porteuse (DCD)■ Prêt à émettre (CTS)■ Ensemble de données prêt (DSR)■ Signal de sonnerie (RI)■ Réception 2 <RC>■ Réponse du modem

Tableau 2-25 Périphérique COM - Propriétés et paramètre

Propriété	Paramètre	Description
<CRemoteData>.ComEndedBy(chaine)	Chaîne	<p>Définit le paramètre de fin des types de connexion COM.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ Toujours connecté ■ Détection de porteuse (DCD) ■ Prêt à émettre (CTS) ■ Ensemble de données prêt (DSR) ■ Signal de sonnerie (RI)
<CRemoteData>.ComSpeed(Long)	Long	<p>Contient le paramètre de vitesse COM maximale</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 110 ■ 300 ■ 600 ■ 1200 ■ 2400 ■ 4800 ■ 9600 ■ 38400 ■ 57600 ■ 115200

Le [Tableau 2-26](#) définit les propriétés et les valeurs renvoyées par le périphérique COM. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-26 Périphériques COM - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CRemoteData>.ComParity	Chaîne	Renvoie l'une des valeurs suivantes comme parité de communications : <ul style="list-style-type: none"> ■ <Aucune> ■ Impaire ■ Paire ■ Marque ■ Espace
String = <CRemoteData>.ComFlowControl	Chaîne	Renvoie le paramètre de contrôle de flux de l'objet Maître. Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ <Aucun> ■ XONXOFF ■ RTS/CTS ■ Les deux
String = <CRemoteData>.ComStartedBy	Chaîne	Renvoie le contrôle de début des types de connexion COM Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ Toujours connecté ■ Détection de porteuse (DCD) ■ Prêt à émettre (CTS) ■ Ensemble de données prêt (DSR) ■ Signal de sonnerie (RI) ■ Réception 2 <RC> ■ Réponse du modem
String = <CRemoteData>.ComEndedBy	Chaîne	Renvoie le contrôle de fin des types de connexion COM Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ Toujours connecté ■ Détection de porteuse (DCD) ■ Prêt à émettre (CTS) ■ Ensemble de données prêt (DSR) ■ Signal de sonnerie (RI)

Tableau 2-26 Périphériques COM - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Long = <CRemoteData>.ComSpeed	Long	Renvoie le paramètre de vitesse de communication de l'objet Maître Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ 110 ■ 300 ■ 600 ■ 1200 ■ 2400 ■ 4800 ■ 9600 ■ 38400 ■ 57600 ■ 115200

Propriétés de périphérique NetBIOS

Le [Tableau 2-27](#) définit les propriétés et les paramètres d'un périphérique réseau NetBIOS. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-27 NetBIOS - Propriétés et paramètres

Propriété	Paramètre	Description
<CRemoteData>.LanaNumber(entier)	Entier	Définit le numéro d'adaptateur réseau (LANA) à utiliser pour les connexions NetBIOS

Le [Tableau 2-28](#) définit les propriétés et les valeurs renvoyées pour NetBIOS. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-28 Propriété NetBIOS et valeur renvoyée

Propriété	Valeur renvoyée	Description
Integer = <CRemoteData>.LanaNumber	Entier	Renvoie le paramètre en cours du numéro LANA pour les connexions NetBIOS.

Propriétés de périphérique RNIS via CAPI 2.0

Le [Tableau 2-29](#) définit les propriétés et les paramètres pour les connexions RNIS européennes. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-29 RNIS via CAPI 2.0 - Propriétés et paramètres

Propriété	Paramètre	Description
<CRemoteData>.CapiChannelBonding(Bool)	Bool	Active ou désactive la liaison de canaux pour les périphériques RNIS CAPI.
<CRemoteData>.CapiExtensions(chaine)	Chaîne	Définit toute extension CAPI supplémentaire nécessaire pour les communications.

Le [Tableau 2-30](#) définit les propriétés et les valeurs renvoyées pour les connexions RNIS européennes. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-30 RNIS via CAPI 2.0 - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Bool= <CRemoteData>.CapiChannelBonding	Bool	Renvoie le paramètre de liaison de canaux RNIS CAPI en cours dans l'objet Maître.
String = <CRemoteData>.CapiExtensions	Chaîne	Renvoie la liste actuelle des extensions CAPI à partir de l'objet Maître.

Détails de périphérique réseau (TCP/IP, SPX) pour les passerelles

Vous pouvez utiliser les propriétés suivantes avec pcAnywhere 9.2x uniquement :

- GatewayUse comme valeur booléenne
- GatewayName comme chaîne
- GatewayClass comme chaîne
- GatewayParity comme chaîne

Remarque : Les versions ultérieures de pcAnywhere ne prennent pas en charge la fonction de passerelle.

Objet CRemoteDataEx

L'objet CRemoteDataEx contient la même fonctionnalité que l'objet CRemoteData avec quelques fonctions supplémentaires. Le [Tableau 2-31](#) définit les propriétés et les paramètres. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-31 CRemoteDataEx - Paramètres

Propriété	Paramètre	Description
<CRemoteData>.PrivateKey (chaîne)	Chaîne	Définit le nom du conteneur de clé privée à utiliser
<CRemoteData>.CertificateName(chaîne)	Chaîne	Définit le nom commun de la clé privée à utiliser
<CRemoteData>.AutoLoginPassword(octet)	Octet	Définit le niveau de chiffrement Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ -1 : Aucun ■ 0 : pcAnywhere ■ 1 : Symétrique ■ 2 : Clé publique
<CRemoteData>.DenyLowerEncrypt(Bool)	Bool	Définit si l'ordinateur Maître autorise une connexion à un ordinateur Elève utilisant un niveau de chiffrement inférieur.
<CRemoteData>.AutoDomain(chaîne)	Chaîne	Définit le nom de domaine pour les connexions automatiques. Cette option s'utilise avec les types d'authentification NT et Windows.

Le [Tableau 2-32](#) définit les propriétés et les valeurs renvoyées pour l'objet CRemoteDataEx. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-32 CRemoteDataEx - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CRemoteData>.PrivateKey	Chaîne	Renvoie le nom du conteneur de clé privée actuellement actif
String = <CRemoteData>.CertificateName	Chaîne	Renvoie le nom commun du conteneur de clé privée actif
Byte = <CRemoteData>.EncryptionLevel	Octet	Renvoie l'un des paramètres de chiffrement suivants : <ul style="list-style-type: none">■ -1 : Aucun■ 0 : pcAnywhere■ 1 : Symétrique■ 2 : Clé publique
Bool = <CRemoteData>.DenyLowerEncrypt	Bool	Renvoie la valeur du paramètre de rejet de niveau de chiffrement inférieur.
String = <CRemoteData>.AutoDomain	Chaîne	Renvoie le paramètre de nom de domaine pour les connexions automatiques. Cette option s'utilise avec les types d'authentification NT et Windows.

Exemple de code Visual Basic pour la fonction Maître

L'exemple de code Visual Basic suivant récupère un objet de données Maître et modifie ses propriétés.

```
Private Sub Command1_Click()  
    Dim RemoteDataManager as Object  
    Dim RemoteData as Object  
    Dim s as string  
  
    'créer un objet CRemoteDataManager  
    Set RemoteDataManager = CreateObject(WINAWSVR.REMOTEDATAMANAGER)
```

```

'afficher et modifier le répertoire en cours
s = RemoteDataManager.CurrentDirectory()
MsgBox ( s )
RemoteDataManager.ChangeDirectory ("C:\dev\bin.w32\data")
s = RemoteDataManager.CurrentDirectory()
MsgBox ( s )

'récupérer un objet de données Maître
Set RemoteData = RemoteDataManager.RetrieveObjectEx("pod.CHF",
2, 0)

'afficher certaines propriétés
s = RemoteData.AreaCode()
MsgBox (s)
s = RemoteData.PhoneNumber()
MsgBox (s)

'définir certaines propriétés
RemoteData.AreaCode = "212"
RemoteData.PhoneNumber = "555-5555"

'écrire l'objet sur disque
RemoteData.WriteObject (0)
End Sub

```

Utilisez les méthodes FindFirst et FindNext pour afficher le fichier Maître dans un répertoire, comme suit :

```

Private Sub Command5_Click()

Dim RemoteDataManager as Object
Dim RemoteData as Object
Dim s as string

Set RemoteDataManager =
CreateObject("WINAWSVR.REMOTEDATAMANAGER")
RemoteDataManager.ChangeDirectory
("C:\dev\bin.w32\data")
RemoteDataManager.FindFirst "*", s
MsgBox (s)
RemoteDataManager.FindNext s
MsgBox (s)

End Sub

```

Créez un objet Maître. Définissez le type de connexion sur TCP/IP et le nom de l'ordinateur sur "Host1". Lancez l'objet comme suit :

```
Private Sub Command6_Click()  
    Dim RemoteDataManager as Object  
    Dim RemoteData as Object  
    Dim s as string  
  
    Set RemoteDataManager =  
        CreateObject ("WINAWSVR.REMOTEDATAMANAGER")  
    MsgBox (RemoteDataManager.CurrentDirectory())  
  
    RemoteDataManager.ChangeDirectory ("C:\dev\bin.w32\data")  
    MsgBox (RemoteDataManager.CurrentDirectory())  
    Set RemoteData = RemoteDataManager.CreateObject("test")  
    RemoteData.ConnectionType = "TCP/IP"  
    RemoteData.ComputerName = "Host1"  
    s = RemoteData.ConnectionType  
    MsgBox (s)  
    s = RemoteData.ComputerName  
    MsgBox (s)  
    RemoteData.WriteObject (0)  
End Sub
```

Méthodes CHostDataManager

Les méthodes CHostDataManager fournissent les paramètres et les valeurs renvoyées pour accéder aux objets CHostData et les contrôler.

CurrentDirectory()

Renvoie le chemin d'accès complet du répertoire en cours dans lequel les objets Elève de pcAnywhere sont stockés.

Le [Tableau 2-33](#) définit la valeur renvoyée par CurrentDirectory.

Tableau 2-33 CurrentDirectory() - Valeur renvoyée

Valeur renvoyée	Description
Chaîne	Chemin d'accès complet du répertoire de données actuel de pcAnywhere

FindFirst(schéma, nom)

Trouve le premier fichier d'objet Elève pcAnywhere (*.bhf) dans le dossier en cours, sur la base du schéma du nom de fichier spécifié.

Le [Tableau 2-34](#) définit les paramètres FindFirst.

Tableau 2-34 FindFirst - Paramètres

Paramètre	Description
Schéma comme chaîne	Schéma de nom de fichier pour filtrer les fichiers d'objet (un astérisque [*] trouve tous les fichiers dans le répertoire en cours).
Nom comme chaîne	Tampon de retour pour le chemin d'accès complet du fichier d'objet Elève correspondant au schéma spécifié.

Le [Tableau 2-35](#) définit la valeur renvoyée.

Tableau 2-35 FindFirst - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si un fichier d'objet Elève correspondant au schéma spécifié est trouvé. Le chemin d'accès complet du fichier est stocké dans Nom.

FindNext(nom)

Après l'appel réussi de FindFirst() pour obtenir le nom d'un fichier d'objet Elève dans le dossier en cours, FindNext() peut être appelé pour trouver le fichier suivant correspondant au schéma.

Le [Tableau 2-36](#) définit le paramètre FindNext.

Tableau 2-36 FindNext - Paramètre

Paramètre	Description
Nom comme chaîne	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié dans l'appel initial de FindFirst().

Le [Tableau 2-37](#) définit la valeur renvoyée.

Tableau 2-37 FindNext - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si un autre fichier d'objet Elève correspondant au schéma spécifié dans l'appel de FindFirst() est trouvé. Le chemin d'accès complet du fichier est stocké dans Nom.

RetrieveObject(nom, mode d'accès, mot de passe)

Récupère un objet CHostData par nom de fichier.

Le [Tableau 2-38](#) définit les paramètres RetrieveObject.

Tableau 2-38 RetrieveObject - Paramètres

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Elève à charger.
Mode d'accès comme valeur entière	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none"> ■ 0 = Non spécifié ■ 1 = Afficher uniquement ■ 2 = Afficher et modifier ■ 3 = Exécuter
Mot de passe comme chaîne	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 2-39](#) définit la valeur renvoyée.

Tableau 2-39 RetrieveObject - Valeur renvoyée

Valeur renvoyée	Description
Objet	Objet CHostData du fichier spécifié

RetrieveObjectEx(nom, mode d'accès, mot de passe)

Récupère un objet CHostDataEx par nom de fichier.

Le [Tableau 2-40](#) définit les paramètres RetrieveObjectEx.

Tableau 2-40 RetrieveObjectEx - Paramètres

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Elève à charger.
Mode d'accès comme valeur entière	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none">■ 0 = Non spécifié■ 1 = Afficher uniquement■ 2 = Afficher et modifier■ 3 = Exécuter
Mot de passe comme chaîne	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 2-41](#) définit la valeur renvoyée.

Tableau 2-41 RetrieveObjectEx - Valeur renvoyée

Valeur renvoyée	Description
Objet	Objet CHostDataEx du fichier spécifié

CreateObject(nom)

Crée un objet CHostData et renvoie un pointeur LPDISPATCH qui le désigne.

Le [Tableau 2-42](#) définit le paramètre CreateObject.

Tableau 2-42 CreateObject - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Elève pour le nouvel objet.

Le [Tableau 2-43](#) définit la valeur renvoyée.

Tableau 2-43 CreateObject - Valeur renvoyée

Valeur renvoyée	Description
Objet	CHostData

CreateObjectEx(nom)

Crée un objet CHostDataEx et renvoie un pointeur LPDISPATCH qui le désigne.
Le [Tableau 2-44](#) définit le paramètre CreateObjectEx.

Tableau 2-44 CreateObjectEx - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Elève pour le nouvel objet.

Le [Tableau 2-45](#) définit la valeur renvoyée.

Tableau 2-45 CreateObjectEx - Valeur renvoyée

Valeur renvoyée	Description
Objet	CHostDataEx

DeleteObject(nom, mot de passe)

Supprime un fichier d'objet Elève.
Le [Tableau 2-46](#) définit les paramètres DeleteObject.

Tableau 2-46 DeleteObject - Paramètres

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Elève de l'objet à supprimer.
Mot de passe comme chaîne	Mot de passe de l'objet

Le [Tableau 2-47](#) définit la valeur renvoyée.

Tableau 2-47 DeleteObject - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si l'objet est supprimé

Launch(nom)

Lance un fichier d'objet Elève, ce qui ouvre la fenêtre de terminal de l'Elève pcAnywhere.

Le [Tableau 2-48](#) définit le paramètre Launch.

Tableau 2-48 Launch - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier d'objet Elève de l'objet à lancer.

Le [Tableau 2-49](#) définit la valeur renvoyée.

Tableau 2-49 Launch - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si l'objet est lancé avec succès.

Propriétés CHostData

Le [Tableau 2-50](#) décrit les propriétés et paramètres disponibles pour l'objet CHostData. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-50 CHostData - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.PhoneNumber(chaîne)	Chaîne	Définit le numéro de téléphone de l'ordinateur Elève.
<CHostData>.UseDialingProperties(Bool)	Bool	Définit les propriétés de numérotation du système.
<CHostData>.RedialCount(entier)	Entier	Définit le nombre de tentatives de rappel avant l'annulation de l'appel.
<CHostData>.RedialDelay(entier)	Entier	Définit le nombre de secondes d'attente entre les tentatives de rappel.

Tableau 2-50 CHostData - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.Password(chaine)	Chaîne	Définit le mot de passe sur l'objet Elève, à utiliser avec les méthodes ExecuteProtection, ReadProtection et WriteProtection. Pour des raisons de sécurité, le serveur d'automatisation pcAnywhere ne permet pas de lire la valeur du mot de passe. Aucune valeur de mot de passe n'est renvoyée.
<CHostData>.ExecuteProtection (Bool)	Bool	Définit l'obligation d'un mot de passe pour exécuter l'objet. Définir par mot de passe.
<CHostData>.ReadProtection(Bool)	Bool	Définit l'obligation d'un mot de passe pour afficher les propriétés de l'objet Elève. Définir par mot de passe.
<CHostData>.WriteProtection(Bool)	Bool	Définit l'obligation d'un mot de passe pour enregistrer les modification de l'objet Elève. Définir par mot de passe.
<CHostData>.LogSession (Bool)	Bool	Active et désactive la consignation des sessions.
<CHostData>.RecordFile(chaine)	Chaîne	Définit le nom et le chemin d'accès intégralement qualifié du fichier d'enregistrement de la session active.
<CHostData>.RecordSession(Bool)	Bool	Active et désactive la consignation automatique de session.
<CHostData>.ReadObject(chaine)	Chaîne	Définit le mot de passe de l'objet. Vous pouvez utiliser cette propriété pour actualiser la copie locale des données de l'objet Elève.
<CHostData>.WriteObject(chaine)	Chaîne	Définit le mot de passe de l'objet. Utilisez cette propriété pour créer l'objet Elève ou pour écrire les changements apportés à celui-ci.

Le [Tableau 2-51](#) définit les propriétés et les valeurs renvoyées. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-51 CHostData - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CHostData>.PhoneNumber	Chaîne	Renvoie le numéro de téléphone de l'ordinateur Elève
Bool = <CHostData>.UseDialingProperties	Bool	Renvoie l'utilisation des propriétés de numérotation du système définies dans l'objet Elève.
Integer = <CHostData>.RedialCount	Entier	Renvoie le nombre de tentatives d'appel défini dans l'objet Elève.
Integer = <CHostData>.RedialDelay	Entier	Renvoie le nombre de secondes d'attente entre les tentatives de rappel.
Bool = <CHostData>.ExecuteProtection	Bool	Renvoie la valeur du paramètre ExecuteProtection.
Bool = <CHostData>.ReadProtection	Bool	Renvoie la valeur du paramètre ReadProtection.
Bool = <CHostData>.WriteProtection	Bool	Renvoie la valeur du paramètre WriteProtection.
Bool = <CHostData>.LogSession	Bool	VRAI si la consignation de session est activée.
String = <CHostData>.RecordFile	Chaîne	Renvoie le nom et le chemin d'accès intégralement qualifié du fichier d'enregistrement de la session.
Bool = <CHostData>.RecordSession	Bool	Renvoie la valeur du paramètre d'enregistrement de session.

Propriétés du type de connexion

Le [Tableau 2-52](#) définit les propriétés et les paramètres. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-52 Type de connexion - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.ConnectionType(chaine)	Chaîne	<p>Définit le type de connexion de l'Elève. La valeur transmise doit être un type de connexion valide, défini par les fonctions FirstConnectionType() et NextConnectionType().</p> <p>Les types suivants sont des exemples de connexions valides :</p> <ul style="list-style-type: none">■ COM1■ COM2■ COM3■ COM4■ LPT1■ LPT2■ LPT3■ LPT4■ TCP/IP■ SPX■ NetBIOS■ Infrarouge■ RNIS via CAPI 2.0■ Nom du modem (tel qu'il apparaît sur l'ordinateur)
<CHostData>.AssignConnection(chaine)	Chaîne	Définit le type de connexion présent dans la chaîne sur l'état actif.
<CHostData>.UnassignConnection(chaine)	Chaîne	Définit le type de connexion présent dans la chaîne sur l'état inactif.

Le [Tableau 2-53](#) définit les propriétés et les valeurs renvoyées. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-53 Type de connexion - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CHostData>.ConnectionTypes	Chaîne	Renvoie le type de connexion de l'objet Elève.
Integer = <CHostData>.ConnectionType	Entier	<p>Renvoie le nombre de types de connexion disponibles.</p> <p>Les types suivants sont des exemples de connexions valides :</p> <ul style="list-style-type: none"> ■ COM1 ■ COM2 ■ COM3 ■ COM4 ■ LPT1 ■ LPT2 ■ LPT3 ■ LPT4 ■ TCP/IP ■ SPX ■ NetBIOS ■ Infrarouge ■ RNIS via CAPI 2.0 ■ Nom du modem (tel qu'il apparaît sur l'ordinateur)
String = <CHostData>.FirstConnectionType	Chaîne	Renvoie le premier type de connexion disponible.
String = <CHostData>.NextConnectionType	Chaîne	Renvoie le type de connexion disponible suivant. L'appel s'effectue de façon séquentielle pour le nombre de types de connexion défini dans <CHostData>.ConnectionType, afin d'énumérer tous les types de connexion.
Bool = <CHostData>.FindConnectionType (ConnectionType)	Bool	VRAI si le type de connexion indiqué a été trouvé dans la liste des types de connexion disponibles.
Integer = <CHostData>.MaxAssignedConnections	Entier	Renvoie le nombre maximum de types de connexion qui peuvent être actifs sur cet Elève.
String = <CHostData>.FirstAssignedConnection	Chaîne	Renvoie le premier type de connexion actif affecté.
String = <CHostData>.NextAssignedConnection	Chaîne	Renvoie le type de connexion actif affecté suivant.
Bool = <CHostData>.FindAssignedConnection (ConnectionType)	Bool	VRAI si le type de connexion transmis correspond à l'un des types de connexion actifs.

Méthode AssignConnection(type de connexion)

Cette méthode place le type de connexion requis dans la liste des types de connexion affectés à l'objet Elève et en fait le type de connexion en cours lors du traitement ultérieur des appels de méthodes spécifiques au périphérique.

Si le type de connexion requis se trouve déjà dans la liste des connexions affectées, celle-ci ne change pas. Seul le type de connexion en cours change pour le type requis. Il est normal d'appeler la méthode AssignConnection sur le même objet plusieurs fois lors de l'obtention et de la définition de valeurs propres à la connexion.

AssignConnection renvoie VRAI si le type de connexion transmis existe sur l'ordinateur et est affecté avec succès ou déjà affecté. Il renvoie FAUX si le type de connexion requis n'existe pas sur l'ordinateur ou si le nombre de connexions affectées a déjà atteint le maximum.

Un objet Elève pcAnywhere peut prendre en charge jusqu'à deux types de connexion affectés. La méthode AssignConnection renvoie FAUX si elle détecte une tentative de dépassement de cette limite.

Le [Tableau 2-54](#) définit le paramètre.

Tableau 2-54 AssignConnection - Paramètre

Paramètre	Description
Type de connexion comme chaîne	Nom d'un type de périphérique de connexion à affecter.

Le [Tableau 2-55](#) définit la valeur renvoyée.

Tableau 2-55 AssignConnection - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si ce type de périphérique est disponible et si le nombre maximum de connexions affectées n'a pas encore été atteint.

Méthode UnassignConnection(type de connexion)

Cette méthode renvoie VRAI si le type de connexion transmis a pu être retiré de la liste des types de connexion affectés.

Le [Tableau 2-56](#) définit le paramètre.

Tableau 2-56 UnassignConnection - Paramètre

Paramètre	Description
Type de connexion comme chaîne	Nom d'un type de périphérique de connexion à désaffecter.

Le [Tableau 2-57](#) définit la valeur renvoyée.

Tableau 2-57 UnassignConnection - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si ce type de périphérique est désaffecté avec succès.

Propriétés de numérotation

Le [Tableau 2-58](#) décrit les propriétés et les valeurs renvoyées pour définir les propriétés de numérotation pour les connexions par modem. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-58 Propriétés de numérotation - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.AreaCode(chaine)	Chaîne	Définit le code de zone pour les connexions par modem
<CHostData>.CountryCode(chaine)	Chaîne	Définit l'indicatif international pour les connexions par modem

Le [Tableau 2-59](#) définit les propriétés et les valeurs renvoyées pour les propriétés de numérotation par modem. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-59 Propriétés de numérotation par modem - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CHostData>.AreaCode	Chaîne	Renvoie le code de zone
String = <CHostData>.CountryCode	Chaîne	Renvoie l'indicatif international.
Integer = <CHostData>.CountryCodes	Entier	Renvoie le nombre d'indicatifs internationaux disponibles.
String= <CHostData>.FirstCountryCode	Chaîne	Renvoie le premier indicatif de pays apparaissant dans la liste du système d'exploitation.
String = <CHostData>.NextCountryCode	Chaîne	Renvoie l'indicatif international disponible suivant. L'appel s'effectue de façon séquentielle pour le nombre d'indicatifs internationaux défini dans <CHostData>.CountryCodes, afin d'énumérer tous les indicatifs internationaux.

Propriétés de périphérique COM

Le [Tableau 2-60](#) définit les propriétés et les paramètres qui vous permettent de personnaliser les paramètres de port du modem et des autres connexions COM. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-60 Périphérique COM - Propriétés et paramètre

Propriété	Paramètre	Description
<CHostData>.ComParity(chaine)	Chaîne	Définit la parité des communications. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ <Aucune>■ Impaire■ Paire■ Marque■ Espace

Tableau 2-60 Périphérique COM - Propriétés et paramètre

Propriété	Paramètre	Description
<CHostData>.ComFlowControl(chaine)	Chaîne	<p>Définit le contrôle de flux des types de connexion COM.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ <Aucun> ■ XONXOFF ■ RTS/CTS ■ Les deux
<CHostData>.ComStartedBy(chaine)	Chaîne	<p>Définit le paramètre de début des types de connexion COM.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ Toujours connecté ■ Détection de porteuse (DCD) ■ Prêt à émettre (CTS) ■ Ensemble de données prêt (DSR) ■ Signal de sonnerie (RI) ■ Réception 2 <RC> ■ Réponse du modem
<CHostData>.ComEndedBy(chaine)	Chaîne	<p>Définit le paramètre de fin des types de connexion COM.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ Toujours connecté ■ Détection de porteuse (DCD) ■ Prêt à émettre (CTS) ■ Ensemble de données prêt (DSR) ■ Signal de sonnerie (RI)
<CHostData>.ComSpeed(Long)	Long	<p>Définit la vitesse COM maximale</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 110 ■ 300 ■ 600 ■ 1200 ■ 2400 ■ 4800 ■ 9600 ■ 38400 ■ 57600 ■ 115200

Le [Tableau 2-61](#) définit les propriétés et les valeurs renvoyées par le périphérique COM. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-61 Périphériques COM - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CHostData>.ComParity	Chaîne	Renvoie l'une des valeurs suivantes comme parité de communications : <ul style="list-style-type: none">■ <Aucune>■ Impaire■ Paire■ Marque■ Espace
String = <CHostData>.ComFlowControl	Chaîne	Renvoie le paramètre de contrôle de flux de l'objet Elève. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ <Aucun>■ XONXOFF■ RTS/CTS■ Les deux
String = <CHostData>.ComStartedBy	Chaîne	Renvoie le contrôle de début des types de connexion COM. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ Toujours connecté■ Détection de porteuse (DCD)■ Prêt à émettre (CTS)■ Ensemble de données prêt (DSR)■ Signal de sonnerie (RI)■ Réception 2 <RC>■ Réponse du modem
String = <CHostData>.ComEndedBy	Chaîne	Renvoie le contrôle de fin des types de connexion COM Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ Toujours connecté■ Détection de porteuse (DCD)■ Prêt à émettre (CTS)■ Ensemble de données prêt (DSR)■ Signal de sonnerie (RI)

Tableau 2-61 Périphériques COM - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Long = <CHostData>.ComSpeed	Long	Renvoie le paramètre de vitesse de communication de l'objet Elève. Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ 110 ■ 300 ■ 600 ■ 1200 ■ 2400 ■ 4800 ■ 9600 ■ 38400 ■ 57600 ■ 115200

Propriétés de périphérique NetBIOS

Le [Tableau 2-62](#) définit les propriétés et les paramètres d'un périphérique réseau NetBIOS. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-62 NetBIOS - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.LanaNumber(entier)	Entier	Définit le numéro d'adaptateur réseau (LANA) à utiliser pour les connexions NetBIOS.

Le [Tableau 2-63](#) définit les propriétés et les valeurs renvoyées d'un périphérique réseau NetBIOS. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-63 Valeurs NetBIOS renvoyées

Propriété	Valeur renvoyée	Description
Integer = <CHostData>.LanaNumber	Entier	Renvoie le paramètre en cours du numéro LANA pour les connexions NetBIOS.

Propriétés de périphérique RNIS via CAPI 2.0

Le [Tableau 2-64](#) définit les propriétés et les paramètres pour les connexions RNIS européennes. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-64 RNIS via CAPI 2.0 - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.CapiChannelBonding(Bool)	Bool	Active ou désactive la liaison de canaux pour les périphériques RNIS CAPI.
<CHostData>.CapiExtensions(chaine)	Chaîne	Définit toute extension CAPI supplémentaire nécessaire pour les communications.

Le [Tableau 2-65](#) définit les propriétés et les valeurs renvoyées pour les connexions RNIS européennes.

Tableau 2-65 RNIS via CAPI 2.0 - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Bool= <CHostData>.CapiChannelBonding	Bool	Renvoie le paramètre de liaison de canaux RNIS CAPI en cours dans l'objet Elève.
String = <CHostData>.CapiExtensions	Chaîne	Renvoie la liste actuelle des extensions CAPI à partir de l'objet Elève.

Détails de périphérique réseau (TCP/IP, SPX) pour les passerelles

Vous pouvez utiliser les propriétés suivantes avec pcAnywhere 9.2x uniquement :

- GatewayUse comme valeur booléenne
- GatewayName comme chaîne
- GatewayClass comme chaîne
- GatewayParity comme chaîne

Remarque : Les versions ultérieures de pcAnywhere ne prennent pas en charge la fonction de passerelle.

Objet CHostDataEx

L'objet CHostDataEx contient la même fonctionnalité que l'objet CHostData avec quelques fonctions supplémentaires. Le [Tableau 2-66](#) définit les propriétés et les paramètres. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-66 CHostDataEx - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.CryptPrivateKey (chaîne)	Chaîne	Définit le nom du conteneur de clé privée à utiliser
<CHostData>.CryptCommonName (chaîne)	Chaîne	Définit le nom commun de la clé privée à utiliser.
<CHostData>.CryptReqLevel(Octet)	Octet	Définit le niveau de chiffrement. Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ -1 : Aucun ■ 0 : pcAnywhere ■ 1 : Symétrique ■ 2 : Clé publique
<CHostData>.CryptRefuseLower(Bool)	Bool	Définit si l'ordinateur Elève accepte une connexion d'un ordinateur Maître utilisant un niveau de chiffrement inférieur.
<CHostData>.CallersPath(chaîne)	Chaîne	Définit le chemin d'accès intégralement qualifié des fichiers d'appelant.
<CHostData>.ConfirmConnect(Bool)	Bool	Définit si un message demandera à l'utilisateur Elève de confirmer les connexions.
<CHostData>.ConfirmTimeout(Octet)	Octet	Définit le nombre de secondes de délai pour confirmer la connexion.
<CHostData>.ConfirmDeny(Bool)	Bool	Définit si la connexion doit se terminer lorsque le délai de confirmation de la connexion expire.
<CHostData>.PwCaseSensitive(Bool)	Bool	Impose l'utilisation de mots de passe sensibles à la casse.
<CHostData>.PwAttempts(Octet)	Octet	Définit le nombre d'échecs d'ouverture de session consécutifs autorisés avant l'abandon de la connexion.
<CHostData>.PwTimeout(Octet)	Octet	Définit le nombre de minutes dont dispose l'utilisateur pour ouvrir une session avant l'abandon de la connexion.

Tableau 2-66 CHostDataEx - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.ActiveKbds(Octet)	Octet	Définit quelle souris et quel clavier seront actifs pendant la connexion. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ 0 : Elève et Maître■ 1 : Elève■ 2 : Maître
<CHostData>.InactiveTimeout(Octet)	Octet	Définit le nombre de minutes pendant lequel le clavier et la souris peuvent rester inactifs avant l'abandon de la connexion.
<CHostData>.LockSystemWhileWait(Bool)	Bool	Définit le verrouillage de l'ordinateur Elève lors du démarrage.
<CHostData>.MinimizeOnLaunch(Bool)	Bool	Définit l'option de démarrage de l'Elève en réduction.
<CHostData>.RunAsService(Bool)	Bool	Active le fonctionnement de l'Elève comme service.
<CHostData>.ConnLostWait(Octet)	Octet	Définit le nombre de minutes d'attente avant d'autoriser un autre appelant à se connecter.
<CHostData>.ConnLostHostOpts(Bool)	Bool	Définit s'il faut attendre une autre connexion ou annuler l'Elève si la session se termine de façon anormale. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ FAUX : Attente de la connexion suivante■ VRAI : Annulation de l'Elève
<CHostData>.EnableConnLostSecurity(Bool)	Bool	Active ou désactive les options de sécurité de fin de session pour les sessions qui se terminent de façon anormale.

Tableau 2-66 CHostDataEx - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.AuthenticationType(octet)	Octet	<p>Définit le type d'authentification.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 0 : pcAnywhere ■ 1 : pcAnywhere ■ 2 : Windows ■ 3 : NT ■ 4 : pcAnywhere ■ 5 : pcAnywhere ■ 6 : ADS Active Directory Services ■ 7 : Microsoft LDAP ■ 8 : FTP ■ 9 : HTTP ■ 10 : HTTPS ■ 11 : Netscape LDAP ■ 12 : Novell LDAP ■ 13 : SecurID RSA <p>L'authentification pcAnywhere est utilisée par défaut si aucune valeur d'authentification n'est définie ou si la valeur définie n'est pas valide (par exemple, si le type d'authentification n'est pas disponible).</p>
<CHostData>.ConnLostSecurity(octet)	Octet	<p>Définit les options de sécurité pour traiter une fin de session anormale.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 1 : Déconnecter l'utilisateur ■ 2 : Redémarrer l'ordinateur Elève ■ 3 : Verrouiller l'ordinateur
<CHostData>.CallbkDelay(octet)	Octet	<p>Définit le nombre de secondes d'attente avant que le modem Elève ne rappelle l'ordinateur Maître.</p>
<CHostData>.EndSessHostOpts(Bool)	Bool	<p>Définit si l'Elève attend une autre connexion ou est annulé après la fin normale de la session.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ FAUX : Attente de la connexion suivante ■ VRAI : Annulation de l'Elève
<CHostData>.EnableEndSessSecurity(Bool)	Bool	<p>Active ou désactive les options de sécurité pour une fin de session normale.</p>

Tableau 2-66 CHostDataEx - Propriétés et paramètres

Propriété	Paramètre	Description
<CHostData>.EndSessSecurity(octet)	Octet	Définit les options de sécurité pour une fin de session normale. Les valeurs suivantes sont valides : <ul style="list-style-type: none">■ 1: Déconnecter l'utilisateur■ 2: Redémarrer l'ordinateur Elève■ 3: Verrouiller l'ordinateur
<CHostData>.BlankHost(Bool)	Bool	Définit l'effacement de l'écran de l'ordinateur Elève. Certaines cartes d'affichage ne prennent pas en charge cette fonction.
<CHostData>.AllowRemoteMouse(Bool)	Bool	Désactive l'utilisation de la souris sur l'ordinateur Maître pendant une session.
<CHostData>.RebootOnDisconnect(Bool)	Bool	Si VRAI, force l'ordinateur à redémarrer après toute fin de session.
<CHostData>.PasswordAfterDisc(Bool)	Bool	Déconnecte l'utilisateur lorsque la session se termine.
<CHostData>.LogFailures(Bool)	Bool	Consigne les tentatives de mot de passe qui ont échoué.
<CHostData>.AllowDriveSecurity(Bool)	Bool	Active les options de sécurité de lecteur. Ce paramètre n'est valide que sur les systèmes de fichiers NTFS.
<CHostData>.UseDirectoryServices(Bool)	Bool	Active l'utilisation des services d'annuaire pour l'authentification.
<CHostData>.DirectoryServiceEntry(chaine)	Chaîne	Définit les paramètres des services d'annuaire.

Le [Tableau 2-67](#) définit les propriétés et les valeurs renvoyées pour l'objet CHostDataEx. Remplacez les informations entre les chevrons par les valeurs réelles.

Tableau 2-67 CHostDataEx - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
String = <CHostDataEx>.CryptPrivateKey	Chaîne	Renvoie le nom du conteneur de clé privée actuellement actif.
String = <CHostDataEx>.CryptCommonName	Chaîne	Renvoie le nom commun du conteneur de clé privée actif.

Tableau 2-67 CHostDataEx - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Byte = <CHostDataEx>.CryptReqLevel	Octet	Renvoie l'un des paramètres de chiffrement suivants : <ul style="list-style-type: none"> ■ -1 : Aucun ■ 0 : pcAnywhere ■ 1 : Symétrique ■ 2 : Clé publique
Bool = <CHostDataEx>.CryptRefuseLower	Bool	Renvoie la valeur du paramètre de rejet de niveau de chiffrement inférieur.
String = <CHostDataEx>.CallersPath	Chaîne	Renvoie le chemin d'accès intégralement qualifié des fichiers d'appelant.
Bool = <CHostDataEx>.ConfirmConnect	Bool	Renvoie la valeur du paramètre de confirmation de connexion.
Byte = <CHostDataEx>.ConfirmTimeout	Octet	Renvoie la valeur du paramètre de rejet de niveau de chiffrement inférieur.
Bool = <CHostDataEx>.ConfirmDeny	Bool	Renvoie la valeur du paramètre de déconnexion sur délai dépassé.
Bool = <CHostDataEx>.PwCaseSensitive	Bool	Renvoie la valeur du paramètre de mot de passe sensible à la casse.
Byte = <CHostDataEx>.PwAttempts	Octet	Renvoie la valeur du paramètre de nombre limite de tentatives d'ouverture de session par appel.
Byte = <CHostDataEx>.PwTimeout	Octet	Renvoie la valeur du paramètre de délai limite d'ouverture de session (en minutes).
Byte = <CHostDataEx>.ActiveKbds	Octet	Renvoie les paramètres de clavier et de souris actifs. Les valeurs suivantes sont valides : <ul style="list-style-type: none"> ■ 0 : Elève et Maître ■ 1 : Elève ■ 2 : Maître
Byte = <CHostDataEx>.InactiveTimeout	Octet	Renvoie le nombre de minutes d'attente avant déconnexion si la limite d'inactivité est atteinte.
Bool = <CHostDataEx>.LockSystemWhileWait	Bool	Renvoie le paramètre de verrouillage de l'ordinateur Elève au démarrage.

Tableau 2-67 CHostDataEx - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Bool = <CHostDataEx>.MinimizeOnLaunch	Bool	Renvoie le paramètre de démarrage de l'Elève en réduction.
Bool = <CHostDataEx>.RunAsService	Bool	Renvoie le paramètre de fonctionnement de l'Elève comme service.
Byte = <CHostDataEx>.ConnLostWait	Octet	Renvoie le nombre de minutes d'attente avant d'autoriser un autre appelant à se connecter.
Bool = <CHostDataEx>.ConnLostHostOpts	Bool	<p>Renvoie le paramètre déterminant si l'Elève attend une autre connexion ou s'arrête si la session se termine de façon anormale.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ FAUX : Attente de la connexion suivante ■ VRAI : Annulation de l'Elève
Bool = <CHostDataEx>.ConnLostWait	Bool	Renvoie la valeur de l'option de sécurité définie pour traiter une fin de session anormale.
Byte = <CHostDataEx>.AuthenticationType	Octet	<p>Renvoie la référence de nombre du type d'authentification.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 0 : pcAnywhere ■ 1 : pcAnywhere ■ 2 : Windows ■ 3 : NT ■ 4 : pcAnywhere ■ 5 : pcAnywhere ■ 6 : ADS Active Directory Services ■ 7 : Microsoft LDAP ■ 8 : FTP ■ 9 : HTTP ■ 10 : HTTPS ■ 11 : Netscape LDAP ■ 12 : Novell LDAP ■ 13 : SecurID RSA <p>L'authentification pcAnywhere est utilisée par défaut si aucune valeur d'authentification n'est définie ou si la valeur définie n'est pas valide (par exemple, si le type d'authentification n'est pas disponible).</p>

Tableau 2-67 CHostDataEx - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Byte = <CHostDataEx>.ConnLostSecurity	Octet	<p>Renvoie la représentation numérique du niveau de sécurité défini pour traiter une fin de session anormale.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 1: Déconnecter l'utilisateur ■ 2: Redémarrer l'ordinateur Elève ■ 3: Verrouiller l'ordinateur
Byte = <CHostDataEx>.CallbkDelay	Octet	Renvoie le nombre de secondes d'attente avant que le modem Elève ne rappelle l'ordinateur Maître.
Bool = <CHostDataEx>.EndSessHostOpts	Bool	<p>Renvoie le paramètre déterminant si l'Elève attend une autre connexion ou est annulé après une fin de session normale.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ FAUX : Attente de la connexion suivante ■ VRAI : Annulation de l'Elève
Bool = <CHostDataEx>.EnableEndSessSecurity	Bool	Renvoie le paramètre déterminant si les options de sécurité de fin de session sont activées pour une fin de session normale.
Byte = <CHostDataEx>.EndSessSecurity	Octet	<p>Renvoie l'option de sécurité définie pour une fin de session normale.</p> <p>Les valeurs suivantes sont valides :</p> <ul style="list-style-type: none"> ■ 1: Déconnecter l'utilisateur ■ 2: Redémarrer l'ordinateur Elève ■ 3: Verrouiller l'ordinateur
Bool = <CHostDataEx>.BlankHost	Bool	Renvoie l'option définie pour l'effacement de l'écran.
Bool = <CHostDataEx>.AllowRemoteMouse	Bool	Renvoie l'option déterminant si l'utilisateur Maître dispose du contrôle du clavier et de la souris au cours d'une session.
Bool = <CHostDataEx>.RebootOnDisconnect	Bool	Renvoie l'option déterminant si l'ordinateur Elève redémarre après la fin de la session.
Bool = <CHostDataEx>.PasswordAfterDisc	Bool	Renvoie l'option déterminant si l'Elève est déconnecté après la fin de la session.

Tableau 2-67 CHostDataEx - Propriétés et valeurs renvoyées

Propriété	Valeur renvoyée	Description
Bool = <CHostDataEx>.LogFailures	Bool	Renvoie l'option déterminant si les échecs de mots de passe sont consignés.
Bool = <CHostDataEx>.AllowDriveSecurity	Bool	Renvoie l'option déterminant si la sécurité du lecteur est activée.
Bool = <CHostDataEx>.UseDirectoryServices	Bool	Renvoie l'option déterminant si les services d'annuaire sont activés.
String = <CHostDataEx>.DirectoryServiceEntry	Chaîne	Renvoie les paramètres de services d'annuaire.

Exemple de code Visual Basic pour la fonction Elève.

L'exemple de code Visual Basic suivant récupère un objet de données Elève et modifie ses propriétés.

```
Private Sub Command1_Click()  
    Dim HostDataManager as Object  
    Dim HostData as Object  
    Dim s as string  
  
    'créer un objet CHostDataManager  
    Set HostDataManager = CreateObject(WINAWSVR.BEHOSTDATAMANAGER)  
  
    'afficher et modifier le répertoire en cours  
    s = HostDataManager.CurrentDirectory()  
    MsgBox ( s )  
    HostDataManager.ChangeDirectory ("C:\dev\bin.w32\data")  
    s = HostDataManager.CurrentDirectory()  
    MsgBox ( s )  
  
    'récupérer un objet de données Maître  
    Set HostData = HostDataManager.RetrieveObject("pod.BHF", 2, 0)  
  
    'afficher certaines propriétés  
    s = HostData.AreaCode()  
    MsgBox (s)  
    s = HostData.PhoneNumber()  
    MsgBox (s)
```

```
'définir certaines propriétés
RemoteData.HostData = "212"
RemoteData.HostData = "555-5555"

'écrire l'objet sur disque
HostData.WriteObject (0)
End Sub
```

Utilisez les méthodes FindFirst et FindNext pour afficher le fichier d'Elève dans un répertoire, comme suit :

```
Private Sub Command5_Click()

Dim HostDataManager as Object
Dim HostData as Object
Dim s as string

Set HostDataManager = CreateObject("WINAWSVR.BEHOSTDATAMANAGER")
HostDataManager.ChangeDirectory ("C:\dev\bin.w32\data")
HostDataManager.FindFirst "*", s
MsgBox (s)
HostDataManager.FindNext s
MsgBox (s)

End Sub
```

Fonctions awrem32

Les fonctions Awrem32 fournissent des paramètres et renvoient des valeurs pour la gestion des connexions entre l'ordinateur Elève et l'ordinateur Maître.

awConnect(nom de fichier)

Crée la connexion à l'ordinateur Elève.

Le [Tableau 2-68](#) définit le paramètre.

Tableau 2-68 awConnect - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier .chf contenant les informations sur l'ordinateur Elève.

Le [Tableau 2-69](#) définit la valeur renvoyée.

Tableau 2-69 awConnect - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

awDisconnect()

Déconnecte l'ordinateur Elève.

Le [Tableau 2-70](#) définit la valeur renvoyée.

Tableau 2-70 awDisconnect - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	Après avoir appelé cette fonction, le programme appelant doit supprimer l'objet (C++ - delete IAwrem32X*, VB - set ObjectName = Nothing;)

FileXferFromHost(fichier Elève, fichier Maître)

Copie un fichier de l'ordinateur Elève vers l'ordinateur Maître. Les paramètres peuvent contenir des caractères génériques.

Le [Tableau 2-71](#) définit les paramètres.

Tableau 2-71 FileXferFromHost - Paramètres

Paramètre	Description
Fichier Elève comme chaîne	Contient le chemin d'accès intégralement qualifié du fichier à copier à partir de l'ordinateur Elève.
Fichier Maître comme chaîne	Contient le chemin d'accès de destination, intégralement qualifié. Les chaînes de fichier Elève et de fichier Maître ne sont pas obligatoirement identiques.

Le [Tableau 2-72](#) définit la valeur renvoyée.

Tableau 2-72 FileXferFromHost - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

FileXferToHost(fichier Elève, fichier Maître)

Copie un fichier de l'ordinateur Maître vers l'ordinateur Elève. Les paramètres peuvent contenir des caractères génériques.

Le [Tableau 2-73](#) définit les paramètres.

Tableau 2-73 FileXferToHost - Paramètres

Paramètre	Description
Fichier Elève comme chaîne	Contient le chemin d'accès de destination, intégralement qualifié.
Fichier Maître comme chaîne	Contient le chemin d'accès intégralement qualifié du fichier à copier à partir de l'ordinateur Maître. Les chaînes de fichier Elève et de fichier Maître ne sont pas obligatoirement identiques.

Le [Tableau 2-74](#) définit la valeur renvoyée.

Tableau 2-74 FileXferToHost - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

CreateFolderOnHost(nom de répertoire)

Crée un nouveau dossier sur l'ordinateur Elève. Cette fonction crée un dossier temporaire sur l'ordinateur Maître, puis copie ce dossier sur l'ordinateur Elève.

Le [Tableau 2-75](#) définit le paramètre.

Tableau 2-75 CreateFolderOnHost - Paramètre

Paramètre	Description
Nom de dossier comme chaîne	Contient le lecteur et le chemin d'accès du dossier sur l'ordinateur Elève.

Le [Tableau 2-76](#) définit les valeurs renvoyées.

Tableau 2-76 CreateFolderOnHost - Valeurs renvoyées

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

ExecuteHostFile(nom de fichier)

Exécute un fichier existant sur l'ordinateur Elève. Cette fonction ne peut exécuter que les fichiers batch, de commandes et exécutables. Elle n'exécute pas les fichiers associés aux exécutables. Par exemple, cette fonction n'ouvre pas Microsoft Word si vous exécutez un fichier .doc.

Le [Tableau 2-77](#) définit le paramètre.

Tableau 2-77 ExecuteHostFile - Paramètre

Paramètre	Description
Nom de fichier comme chaîne	Contient le chemin d'accès intégralement qualifié du fichier sur l'ordinateur Elève.

Le [Tableau 2-78](#) définit la valeur renvoyée.

Tableau 2-78 ExecuteHostFile - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

GetError()

Renvoie la dernière erreur sous forme de chaîne.

Le [Tableau 2-79](#) définit la valeur renvoyée.

Tableau 2-79 GetError() - Valeur renvoyée

Valeur renvoyée	Description
Chaîne	Renvoie la dernière erreur générée dans Awrem32.

ConnectionStatus()

Renvoie l'état actuel de la connexion vers l'ordinateur Elève.

Le [Tableau 2-80](#) définit la valeur renvoyée.

Tableau 2-80 ConnectionStatus() - Valeur renvoyée

Valeur renvoyée	Description
Courte	Les valeurs possibles sont les suivantes : <ul style="list-style-type: none">■ -1 = Connexion perdue■ 0 = Aucune connexion■ 1 = Session connectée

Définitions d'objets Visual C++

Ce chapitre traite des sujets suivants :

- [A propos des objets Visual C++](#)
- [Méthodes CRemoteDataManager](#)
- [Objet CRemoteData](#)
- [Objet CRemoteDataEx](#)
- [Méthodes CHostDataManager](#)
- [Objet CHostData](#)
- [Objet CHostDataEx](#)
- [Fonctions awrem32](#)

A propos des objets Visual C++

Le serveur d'automatisation pcAnywhere offre les composants suivants pour la prise en charge de l'automatisation OLE :

- Winawsvr : contient les informations nécessaires pour se connecter au serveur d'automatisation pcAnywhere et accéder à ses interfaces.
- Awrem32 : contient les informations nécessaires au contrôle des connexions pcAnywhere.

Les objets décrits dans le [Tableau 3-1](#) composent Winawsvr.

Tableau 3-1 Objets Winawsvr

Objet	Description	Référence
CRemoteDataManager	Fournit les méthodes pour créer, ouvrir, modifier, enregistrer et supprimer les objets CRemoteData.	Se reporter à « Méthodes CRemoteDataManager » à la page 69.
CRemoteData	Définit les paramètres permettant d'accéder à la fonction Maître de pcAnywhere et de la contrôler.	Se reporter à « Objet CRemoteData » à la page 75. Se reporter à « Objet CRemoteDataEx » à la page 84.
CHostDataManager	Fournit les méthodes pour créer, ouvrir, modifier, enregistrer et supprimer les objets ChostData.	Se reporter à « Méthodes CHostDataManager » à la page 85.
CHostData	Définit les paramètres permettant d'accéder à la fonction Elève de pcAnywhere et de la contrôler.	Se reporter à « Objet CRemoteData » à la page 75. Se reporter à « Objet CRemoteDataEx » à la page 84.

Awrem32 possède un objet, constitué de huit interfaces, pour prendre en charge les sessions de téléintervention et de transfert de fichiers.

Se reporter à « [Fonctions awrem32](#) » à la page 105.

Certaines fonctions, notamment les passerelles, ne sont plus prises en charge. Cependant, les définitions d'objets sont utilisables avec les versions précédentes.

Se reporter à « [Détails de périphérique réseau \(TCP/IP, SPX\) pour les passerelles](#) » à la page 95.

Pour les fonctions nécessitant des mots de passe, les valeurs des mots de passe peuvent être définies mais non récupérées, pour des raisons de sécurité.

Méthodes CRemoteDataManager

Les méthodes CRemoteDataManager fournissent les paramètres et renvoient les valeurs permettant d'accéder aux objets CRemoteData et de les contrôler.

BSTR CurrentDirectory();

Obtient le chemin d'accès complet du dossier actuel dans lequel les objets Maître pcAnywhere sont stockés.

Le [Tableau 3-2](#) définit le paramètre.

Tableau 3-2 BSTR CurrentDirectory(); - Paramètre

Paramètre	Description
BSTR	Chemin d'accès complet du répertoire de données actuel de pcAnywhere.

BOOL ChangeDirectory(LPCTSTR lpszNewDirectory);

Change le dossier actuel dans lequel les objets Maître pcAnywhere sont stockés.

Le [Tableau 3-3](#) définit le paramètre.

Tableau 3-3 BOOL ChangeDirectory - Paramètre

Paramètre	Description
LPCTSTR lpszNewDirectory	Nom d'un répertoire existant.

Le [Tableau 3-4](#) définit la valeur renvoyée.

Tableau 3-4 BOOL ChangeDirectory - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si réussi.

BOOL FindFirst(LPCTSTR lpszPattern, BSTR FAR* pbstrFullQualName);

Recherche le premier fichier d'objet Maître pcAnywhere (*.chf) dans le dossier en cours, sur la base du schéma de nom de fichier spécifié.

Le [Tableau 3-5](#) définit les paramètres.

Tableau 3-5 BOOL FindFirst - Paramètre

Paramètre	Description
LPCTSTR lpszPattern	Schéma de nom de fichier pour filtrer les fichiers d'objet (un astérisque [*] trouve tous les fichiers dans le répertoire en cours).
BSTR FAR * pbstrFullQualName	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié.

Le [Tableau 3-6](#) définit la valeur renvoyée.

Tableau 3-6 BOOL FindFirst - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si un fichier d'objet Maître correspondant au schéma spécifié est trouvé. Le chemin d'accès complet du fichier est stocké dans pbstrFullQualName.

BOOL FindNext(BSTR FAR* pbstrFullQualName);

Après l'appel réussi de FindFirst() pour obtenir le nom d'un fichier d'objet Maître dans le dossier en cours, FindNext() peut être appelé pour trouver le fichier suivant correspondant au schéma.

Le [Tableau 3-7](#) définit le paramètre.

Tableau 3-7 BOOL FindNext - Paramètre

Paramètre	Description
BSTR FAR * pbstrFullQualName	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié dans l'appel initial de FindFirst().

Le [Tableau 3-8](#) définit la valeur renvoyée.

Tableau 3-8 BOOL FindNext renvoyées - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si un autre fichier d'objet Maître correspondant au schéma spécifié dans l'appel de FindFirst() est trouvé. Le chemin d'accès complet du fichier est stocké dans pbstrFullQualName.

LPDISPATCH RetrieveObject(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);

Récupère un objet CRemoteData par nom de fichier.

Le [Tableau 3-9](#) définit les paramètres.

Tableau 3-9 LPDISPATCH RetrieveObject - Paramètres

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Maître à charger.
short wAccessMode	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none">■ 0 = Non spécifié■ 1 = Afficher uniquement■ 2 = Afficher et modifier■ 3 = Exécuter
LPCTSTR lpszPassword	Mot de passe de l'objet. Peut être NULL.

LPDISPATCH RetrieveObjectEx(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);

Récupère un objet CRemoteDataEx par nom de fichier.

Le [Tableau 3-10](#) définit les paramètres.

Tableau 3-10 LPDISPATCH RetrieveObjectEx - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Maître à charger.
short wAccessMode	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none">■ 0 = Non spécifié■ 1 = Afficher uniquement■ 2 = Afficher et modifier■ 3 = Exécuter
LPCTSTR lpszPassword	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 3-11](#) définit la valeur renvoyée.

Tableau 3-11 LPDISPATCH RetrieveObjectEx - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CRemoteDataEx. Pour consulter un exemple d'attachement de ce pointeur à un objet CRemoteDataEx, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Maître » à la page 84.

LPDISPATCH CreateObject(LPCTSTR lpszFQName);

Crée un objet CRemoteData et renvoie un pointeur LPDISPATCH qui le désigne.
Le [Tableau 3-12](#) définit le paramètre.

Tableau 3-12 LPDISPATCH CreateObject - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Maître pour le nouvel objet.

Le [Tableau 3-13](#) définit la valeur renvoyée.

Tableau 3-13 LPDISPATCH CreateObject - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CRemoteData. Pour consulter un exemple d'attachement de ce pointeur à un objet CRemoteData, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Maître » à la page 84.

LPDISPATCH CreateObjectEx(LPCTSTR lpszFQName);

Crée un objet CRemoteDataEx et renvoie un pointeur LPDISPATCH qui le désigne.
Le [Tableau 3-14](#) définit le paramètre.

Tableau 3-14 LPDISPATCH CreateObjectEx - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Maître pour le nouvel objet.

Le [Tableau 3-15](#) définit la valeur renvoyée.

Tableau 3-15 LPDISPATCH CreateObjectEx - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CRemoteDataEx. Pour consulter un exemple d'attachement de ce pointeur à un objet CRemoteDataEx, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Maître » à la page 84.

BOOL DeleteObject(LPCTSTR IpszFQName, LPCTSTR IpszPassword);

Supprime un fichier d'objet Maître.

Le [Tableau 3-16](#) définit les paramètres.

Tableau 3-16 BOOL DeleteObject - Paramètre

Paramètre	Description
LPCTSTR IpszFQName	Nom intégralement qualifié du fichier d'objet Maître de l'objet à supprimer.
LPCTSTR IpszPassword	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 3-17](#) définit la valeur renvoyée.

Tableau 3-17 BOOL DeleteObject - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est supprimé.

BOOL Launch(LPCTSTR IpszFQName);

Lance un fichier d'objet Maître, ce qui ouvre la fenêtre de terminal Maître pcAnywhere.

Le [Tableau 3-18](#) définit le paramètre.

Tableau 3-18 BOOL Launch - Paramètre

Paramètre	Description
LPCTSTR IpszFQName	Nom intégralement qualifié du fichier d'objet Maître de l'objet à lancer.

Le [Tableau 3-19](#) définit la valeur renvoyée.

Tableau 3-19 BOOL Launch - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est lancé avec succès.

Objet CRemoteData

Utilisez cet objet pour modifier les données d'objet Maître.

Méthodes Get et Set (obtenir et définir)

Les méthodes suivantes servent à obtenir (Get) et définir (Set) les propriétés de l'objet CRemoteData.

Le nom de l'ordinateur est le nom de système de l'Elève pcAnywhere à appeler lorsque l'objet Maître est lancé.

```
BSTR GetComputerName();  
void SetComputerName(LPCTSTR lpszNewValue);
```

Le numéro de téléphone est le numéro à composer pour établir une connexion par modem avec un ordinateur Elève pcAnywhere.

```
BSTR GetPhoneNumber();  
void SetPhoneNumber(LPCTSTR lpszNewValue);
```

Indique si des propriétés de numérotation TAPI doivent être utilisées (informations d'emplacement) (VRAI) ou si la chaîne du numéro de téléphone doit être utilisée exactement telle qu'elle apparaît (FAUX).

```
BOOL GetUseDialingProperties();  
void SetUseDialingProperties(BOOL bNewValue);
```

Si les propriétés de numérotation sont utilisées, il s'agit du code de zone du numéro à appeler.

```
BSTR GetAreaCode();  
void SetAreaCode(LPCTSTR lpszNewValue);
```

Si les propriétés de numérotation sont utilisées, il s'agit de l'indicatif international du numéro à appeler.

```
BSTR GetCountryCode();  
void SetCountryCode(LPCTSTR lpszNewValue);
```

Nombre de tentatives d'appel de ce numéro si l'appel échoue.

```
short GetRedialCount();  
void SetRedialCount(short nNewValue);
```

Délai en secondes entre les tentatives d'appel.

```
short GetRedialDelay();  
void SetRedialDelay(short nNewValue);
```

Nom d'utilisateur à envoyer à l'Elève lorsqu'une connexion est établie.
 Si ce champ reste vide, l'utilisateur doit fournir un nom à la connexion.

```
BSTR GetAutoLoginName();  
void SetAutoLoginName(LPCTSTR lpszNewValue);
```

Mot de passe de session envoyé à l'Elève lorsqu'une connexion est établie. Si ce champ reste vide, l'utilisateur doit fournir un mot de passe lors de la connexion.

```
BSTR GetAutoLoginPassword();  
void SetAutoLoginPassword(LPCTSTR lpszNewValue);
```

Mot de passe de cet objet.

```
BSTR GetPassword();  
void SetPassword(LPCTSTR lpszNewValue);
```

L'objet ne peut être lancé que si le mot de passe est utilisé (VRAI).

```
BOOL GetExecuteProtection();  
void SetExecuteProtection(BOOL bNewValue);
```

L'objet ne peut être affiché que si le mot de passe correct est fourni (VRAI).

```
BOOL GetReadProtection();  
void SetReadProtection(BOOL bNewValue);
```

L'objet ne peut être écrit que si le mot de passe correct est fourni (VRAI).

```
BOOL GetWriteProtection();  
void SetWriteProtection(BOOL bNewValue);
```

Contrôle si les sessions utilisant cet objet sont consignées.

```
BOOL GetLogSession();  
void SetLogSession(BOOL bNewValue);
```

Contrôle si les sessions utilisant cet objet sont enregistrées depuis le début.

```
BOOL GetRecordSession();  
void SetRecordSession(BOOL bNewValue);
```

Nom du fichier d'enregistrement pour les sessions utilisant cet objet.

```
BSTR GetRecordFile();  
void SetRecordFile(LPCTSTR lpszNewValue);
```

Définit les types de connexion.

```
BOOL GetRunOnConnect();  
void SetRunOnConnect(BOOL bNewValue);
```

Les types de connexion suivants sont disponibles :

- COM1
- COM2
- COM3
- COM4
- Infrarouge
- RNIS via CAPI 2.0
- LPT1
- LPT2
- LPT3
- LPT4
- NetBIOS
- SPX
- DEFAULT TAPI
- TCP/IP

Le nom d'un périphérique TAPI peut également servir de type de connexion. DEFAULT TAPI utilise le premier périphérique TAPI trouvé dans le système. Pour utiliser un périphérique TAPI spécifique, utilisez FirstConnectionType() / NextConnectionType() pour rechercher les périphériques disponibles.

Méthodes de détail d'objet Maître

Lorsqu'un type de connexion est affecté à un objet Maître, les détails du périphérique sont définis sur les valeurs par défaut valides. Les types de connexion suivants possèdent des options de configuration avancées qui peuvent être définies dans l'application :

- Périphériques COM
- Périphériques de passerelle réseau (TCP/IP, SPX)
Les périphériques de passerelle ne sont pris en charge que dans pcAnywhere 9.2x.
- Périphériques NetBIOS
- Périphériques RNIS via CAPI 2.0 (RNIS européen seulement)

Détails de périphérique COM

Définit la parité des communications.

```
BSTR GetComParity();  
void SetComParity(LPCTSTR lpszNewValue);
```

Les valeurs de parité de communication sont les suivantes :

- Aucune
- Impaire
- Paire
- Marque
- Espace

Définit le niveau de contrôle de flux.

```
BSTR GetComFlowControl();  
void SetComFlowControl(LPCTSTR lpszNewValue);
```

Les valeurs de contrôle de flux sont les suivantes :

- <Aucun>
- XONXOFF
- RTS/CTS
- LES DEUX

Définit le paramètre de début de connexion.

```
BSTR GetComStartedBy();  
void SetComStartedBy(LPCTSTR lpszNewValue);
```

Les valeurs de début de connexion sont les suivantes :

- Toujours connecté
- Détection de porteuse (DCD)
- Prêt à émettre (CTS)
- Ensemble de données prêt (DSR)
- Signal de sonnerie (RI)
- Réception 2 <RC>
- Réponse du modem

Définit les valeurs de fin de connexion.

```
BSTR GetComEndedBy();  
void SetComEndedBy(LPCTSTR lpszNewValue);
```

Les valeurs de fin de connexion sont les suivantes :

- Toujours connecté
- Détection de porteuse (DCD)
- Prêt à émettre (CTS)
- Ensemble de données prêt (DSR)
- Signal de sonnerie (RI)

Définit la vitesse de connexion.

```
long GetComSpeed();  
void SetComSpeed(long nNewValue);
```

Les valeurs de vitesse de connexion sont les suivantes :

- 110
- 300
- 600
- 1200
- 2400
- 4800
- 9600
- 19200
- 38400
- 57600
- 115200

Détails de périphérique réseau (TCP/IP, SPX) pour les passerelles

Vous pouvez utiliser les propriétés suivantes avec les connexions pcAnywhere 9.2x uniquement :

Se connecter par une passerelle pcAnywhere (VRAI).

```
BOOL GetGatewayUse();  
void SetGatewayUse(BOOL bNewValue);
```

Nom de la passerelle pcAnywhere à utiliser.

```
BSTR GetGatewayName();  
void SetGatewayName(LPCTSTR lpszNewValue);
```

Classe de passerelle pcAnywhere à utiliser.

```
BSTR GetGatewayClass();  
void SetGatewayClass(LPCTSTR lpszNewValue);
```

Valeur de parité de passerelle pcAnywhere à utiliser.

```
BSTR GetGatewayParity();  
void SetGatewayParity(LPCTSTR lpszNewValue);
```

Les valeurs de parité sont les suivantes :

- <Aucune>
- Impaire
- Paire
- Marque
- Espace

Détails de périphérique NetBios

Définit le numéro de LANA (adaptateur réseau) à utiliser pour cette connexion.

```
short GetLanaNumber();  
void SetLanaNumber(short nNewValue);
```

Détails de périphérique RNIS via CAPI 2.0

Active la liaison de canaux (utilise deux canaux RNIS pour une connexion) si VRAI.

```
BOOL GetCapiChannelBonding();  
void SetCapiChannelBonding(BOOL bNewValue);
```

Définit toute extension CAPI supplémentaire nécessaire pour les communications.

```
BSTR GetCapiExtensions();  
void SetCapiExtensions(LPCTSTR lpszNewValue);
```


Méthodes d'objet Maître

Les méthodes suivantes sont les méthodes normales de l'objet Maître (elles ne servent pas à obtenir et définir les propriétés de l'objet) :

- `short ConnectionTypes();`
- `BSTR FirstConnectionType();` et `BSTR NextConnectionType();`
- `BOOL FindConnectionType(LPCTSTR lpszConnectionType);`
- `short CountryCodes();`
- `BSTR FirstCountryCode();` et `BSTR NextCountryCode();`
- `BOOL ReadObject(LPCTSTR lpszPassword);`
- `BOOL WriteObject(LPCTSTR lpszPassword);`

short ConnectionTypes();

Renvoie le nombre de types de connexion disponibles.

Le [Tableau 3-20](#) définit la valeur renvoyée.

Tableau 3-20 `short ConnectionTypes();` - Valeur renvoyée

Valeur renvoyée	Description
Courte	Nombre de types de connexion disponibles sur cet ordinateur

BSTR FirstConnectionType(); et BSTR NextConnectionType();

`FirstConnectionType()` et `NextConnectionType()` servent à itérer parmi les types de connexion disponibles. Ces fonctions renvoient un BSTR qui est le nom d'un type de connexion disponible. Les types de connexion renvoyés peuvent être utilisés avec la fonction `SetConnectionType()`.

Le [Tableau 3-21](#) définit la valeur renvoyée.

Tableau 3-21 `BSTR FirstConnectionType();` et `BSTR NextConnectionType();` - Valeur renvoyée

Valeur renvoyée	Description
BSTR	Nom d'un type de périphérique de connexion pris en charge.

BOOL FindConnectionType(LPCTSTR lpszConnectionType);

Renvoie VRAI si le type de connexion transmis existe sur l'ordinateur.

Le [Tableau 3-22](#) définit le paramètre.

Tableau 3-22 BOOL FindConnectionType - Paramètre

Paramètre	Description
LPCTSTR lpszConnectionType	Nom d'un type de périphérique de connexion

Le [Tableau 3-23](#) définit la valeur renvoyée.

Tableau 3-23 BOOL FindConnectionType - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si ce type de périphérique est disponible

short CountryCodes();

Renvoie le nombre d'indicatifs internationaux disponibles.

Le [Tableau 3-24](#) définit la valeur renvoyée.

Tableau 3-24 short CountryCodes(); - Valeur renvoyée

Valeur renvoyée	Description
Courte	Nombre d'indicatifs internationaux disponibles

BSTR FirstCountryCode(); et BSTR NextCountryCode();

FirstCountryCode() et NextCountryCode() servent à circuler parmi les indicatifs internationaux disponibles. Ces fonctions renvoient un BSTR qui est le nom d'un indicatif international disponible. Les valeurs renvoyées peuvent être utilisées avec la fonction SetCountryCode().

Le [Tableau 3-25](#) définit la valeur renvoyée.

Tableau 3-25 BSTR FirstCountryCode(); et BSTR NextCountryCode(); -Valeur renvoyée

Valeur renvoyée	Description
BSTR	Première chaîne d'indicateur international ou chaîne suivante.

BOOL ReadObject(LPCTSTR lpszPassword);

Lit les données d'objet du fichier d'objet Maître.

Le [Tableau 3-26](#) définit le paramètre.

Tableau 3-26 BOOL ReadObject - Paramètre

Paramètre	Description
LPCTSTR lpszPassword	Mot de passe de l'objet

Le [Tableau 3-27](#) définit la valeur renvoyée.

Tableau 3-27 BOOL ReadObject - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est lu avec succès.

BOOL WriteObject(LPCTSTR lpszPassword);

Ecrit les données d'objet dans le fichier d'objet Maître.

Le [Tableau 3-28](#) définit le paramètre.

Tableau 3-28 BOOL WriteObject - Paramètre

Paramètre	Description
LPCTSTR lpszPassword	Mot de passe de l'objet

Le [Tableau 3-29](#) définit la valeur renvoyée.

Tableau 3-29 BOOL WriteObject - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est écrit avec succès.

Objet CRemoteDataEx

L'objet CRemoteDataEx contient la même fonctionnalité que l'objet CRemoteData, avec les méthodes supplémentaires Get et Set (obtenir et définir) suivantes :

```
BSTR GetPrivateKey(); //Renvoie les informations de PrivateKey
void SetPrivateKey(LPCTSTR lpszNewValue);
BSTR GetCertificationName(); //Renvoie le nom de certificat
void SetCertificationName(LPCTSTR lpszNewValue);
short GetEncryptionLevel(); //Renvoie la valeur du niveau
de chiffrement
void SetEncryptionLevel(short nNewValue);
BOOL GetDenyLowerEncrypt(); //Renvoie la valeur DenyLowerEncrypt
void SetDenyLowerEncrypt(BOOL bNewValue);
BSTR GetAutoDomain(); //Renvoie la valeur AutoDomain
void SetAutoDomain(LPCTSTR lpszNewValue);
```

Exemple de code Visual C++ pour la fonction Maître

L'exemple suivant de fonction C++ crée un objet Maître, définit son type de connexion sur TCP/IP, définit le nom de l'ordinateur sur l'adresse TCP/IP transmise dans la fonction, puis lance l'objet Maître.

```
BOOL LaunchTCPRemote(LPCTSTR lpszAddress)
{
    BOOL bReturn = FALSE;

    CRemoteDataManager remoteDM;

    CRemoteData remoteData;

    // Créer d'abord le CRemoteDataManager
    remoteDM.CreateDispatch( _T( "WINAHSV.R.RemoteDataManager" ) );
    // Créer ensuite CRemoteData et le rattacher
    remoteData.AttachDispatch( remoteDM.CreateObject("Test", 0) );

    // Définir à présent les propriétés requises
    remoteData.SetConnectionType("TCP/IP");

    remoteData.SetComputerName(lpszAddress);
```

```
// Enregistrer les données de l'objet
if (remoteData.WriteObject(0))
{
    // Et le lancer
    if (remoteData.Launch())
        bReturn = TRUE;
}
// Libérer l'objet Maître.
remoteData.ReleaseDispatch();

remoteDM.ReleaseDispatch( _T( "WINAWSVR.RemoteDataManager" ) );

return bReturn;
}
```

Méthodes CHostDataManager

Les méthodes CHostDataManager fournissent les paramètres et les valeurs renvoyées pour accéder aux objets CHostData et les contrôler.

BSTR CurrentDirectory();

Renvoie le chemin d'accès complet du répertoire en cours dans lequel les objets Elève de pcAnywhere sont stockés.

Le [Tableau 3-30](#) définit la valeur renvoyée.

Tableau 3-30 BSTR CurrentDirectory(); - Valeur renvoyée

Valeur renvoyée	Description
BSTR	Chemin d'accès complet du répertoire de données actuel de pcAnywhere.

BOOL ChangeDirectory(LPCTSTR lpszNewDirectory);

Change le dossier en cours dans lequel les objets Elève de pcAnywhere sont stockés.

Le [Tableau 3-31](#) définit le paramètre.

Tableau 3-31 BOOL ChangeDirectory - Paramètre

Paramètre	Description
LPCTSTR lpszNewDirectory	Nom d'un répertoire existant

Le [Tableau 3-32](#) définit la valeur renvoyée.

Tableau 3-32 BOOL ChangeDirectory - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si réussi.

BOOL FindFirst(LPCTSTR lpszPattern, BSTR FAR* pbstrFullQualName);

Trouve le premier fichier d'objet Elève pcAnywhere (*.BHF) dans le dossier en cours, sur la base du schéma du nom de fichier spécifié.

Le [Tableau 3-33](#) définit les paramètres.

Tableau 3-33 BOOL FindFirst - Paramètres

Paramètre	Description
LPCTSTR lpszPattern	Schéma de nom de fichier pour filtrer les fichiers d'objet (un astérisque [*] trouve tous les fichiers Elève dans le répertoire en cours).
BSTR FAR * pbstrFullQualName	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié.

Le [Tableau 3-34](#) définit la valeur renvoyée.

Tableau 3-34 BOOL FindFirst - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si un fichier d'objet Elève correspondant au schéma spécifié est trouvé. Le chemin d'accès complet du fichier est stocké dans pbstrFullQualName.

BOOL FindNext(BSTR FAR* pbstrFullQualName);

Après l'appel réussi de FindFirst() pour obtenir le nom d'un fichier d'objet Elève dans le dossier en cours, FindNext() peut être appelé pour trouver le fichier suivant correspondant au schéma.

Le [Tableau 3-35](#) définit le paramètre.

Tableau 3-35 BOOL FindNext - Paramètre

Paramètre	Description
BSTR FAR * pbstrFullQualName	Tampon de retour pour le chemin d'accès complet du fichier d'objet Maître correspondant au schéma spécifié dans l'appel initial de FindFirst().

Le [Tableau 3-36](#) définit la valeur renvoyée.

Tableau 3-36 BOOL FindNext renvoyées - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si un autre fichier d'objet Elève correspondant au schéma spécifié dans l'appel de FindFirst() est trouvé. Le chemin d'accès complet du fichier est stocké dans pbstrFullQualName.

LPDISPATCH RetrieveObject(LPCTSTR lpszFQName, short wAccessMode, LPCTSTR lpszPassword);

Récupère un objet CHostData par nom de fichier.

Le [Tableau 3-37](#) définit les paramètres.

Tableau 3-37 LPDISPATCH RetrieveObject - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Elève à charger.
short wAccessMode	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none"> ■ 0 = Non spécifié ■ 1 = Afficher uniquement ■ 2 = Afficher et modifier ■ 3 = Exécuter
LPCTSTR lpszPassword	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 3-38](#) définit la valeur renvoyée.

Tableau 3-38 LPDISPATCH RetrieveObject - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CHostData. Pour consulter un exemple d'attachement de ce pointeur à un objet CHostData, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Elève » à la page 104.

LPDISPATCH RetrieveObjectEx(LPCTSTR IpszFQName,
short wAccessMode, LPCTSTR IpszPassword);

Récupère un objet CHostDataEx par nom de fichier.

Le [Tableau 3-39](#) définit les paramètres.

Tableau 3-39 LPDISPATCH RetrieveObjectEx - Paramètre

Paramètre	Description
LPCTSTR IpszFQName	Nom intégralement qualifié du fichier d'objet Elève à charger.
short wAccessMode	Spécifie comment cet objet doit être utilisé. Ce paramètre est lié à la protection par mot de passe. Les options sont les suivantes : <ul style="list-style-type: none">■ 0 = Non spécifié■ 1 = Afficher uniquement■ 2 = Afficher et modifier■ 3 = Exécuter
LPCTSTR IpszPassword	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 3-40](#) définit la valeur renvoyée.

Tableau 3-40 LPDISPATCH RetrieveObjectEx - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CHostDataEx. Pour consulter un exemple d'attachement de ce pointeur à un objet CHostDataEx, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Elève » à la page 104.

LPDISPATCH CreateObject(LPCTSTR lpszName);

Crée un objet CHostData et renvoie un pointeur LPDISPATCH qui le désigne.

Le [Tableau 3-41](#) définit le paramètre.

Tableau 3-41 LPDISPATCH CreateObject - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Elève pour le nouvel objet.

Le [Tableau 3-42](#) définit la valeur renvoyée.

Tableau 3-42 LPDISPATCH CreateObject - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CHostData. Pour consulter un exemple d'attachement de ce pointeur à un objet CHostData, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Elève » à la page 104.

LPDISPATCH CreateObjectEx(LPCTSTR lpszName);

Crée un objet CHostDataEx et renvoie un pointeur LPDISPATCH qui le désigne.

Le [Tableau 3-43](#) définit le paramètre.

Tableau 3-43 LPDISPATCH CreateObjectEx - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Elève pour le nouvel objet.

Le [Tableau 3-44](#) définit la valeur renvoyée.

Tableau 3-44 LPDISPATCH CreateObjectEx - Valeur renvoyée

Valeur renvoyée	Description
LPDISPATCH	Pointeur sur un objet envoi OLE. L'objet est un objet CHostDataEx. Pour consulter un exemple d'attachement de ce pointeur à un objet CHostDataEx, reportez-vous à la section « Exemple de code Visual C++ pour la fonction Elève » à la page 104.

BOOL DeleteObject(LPCTSTR lpszFQName, LPCTSTR lpszPassword);

Supprime un fichier d'objet Elève.

Le [Tableau 3-45](#) définit les paramètres.

Tableau 3-45 BOOL DeleteObject - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Elève de l'objet à supprimer.
LPCTSTR lpszPassword	Mot de passe de l'objet. Peut être NULL.

Le [Tableau 3-46](#) définit la valeur renvoyée.

Tableau 3-46 BOOL DeleteObject - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est supprimé.

BOOL Launch(LPCTSTR lpszFQName);

Lance un fichier d'objet Elève, ce qui ouvre la fenêtre de terminal de l'Elève pcAnywhere.

Le [Tableau 3-47](#) définit le paramètre.

Tableau 3-47 BOOL Launch - Paramètre

Paramètre	Description
LPCTSTR lpszFQName	Nom intégralement qualifié du fichier d'objet Elève de l'objet à lancer.

Le [Tableau 3-48](#) définit la valeur renvoyée.

Tableau 3-48 BOOL Launch - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est lancé avec succès.

Objet CHostData

Utilisez cet objet pour modifier les données de l'objet Elève.

Méthodes Get et Set (obtenir et définir)

Les méthodes suivantes servent à obtenir (Get) et définir (Set) les propriétés de l'objet CHostData.

Le numéro de téléphone est le numéro à composer pour établir une connexion par modem avec un ordinateur Maître pcAnywhere.

```
BSTR GetPhoneNumber();  
void SetPhoneNumber(LPCTSTR lpszNewValue);
```

Indique si des propriétés de numérotation TAPI doivent être utilisées (informations d'emplacement) (VRAI) ou si la chaîne du numéro de téléphone doit être utilisée exactement telle qu'elle apparaît (FAUX).

```
BOOL GetUseDialingProperties();  
void SetUseDialingProperties(BOOL bNewValue);
```

Si les propriétés de numérotation sont utilisées, il s'agit du code de zone du numéro à appeler.

```
BSTR GetAreaCode();  
void SetAreaCode(LPCTSTR lpszNewValue);
```

Si les propriétés de numérotation sont utilisées, il s'agit de l'indicatif international du numéro à appeler.

```
BSTR GetCountryCode();  
void SetCountryCode(LPCTSTR lpszNewValue);
```

Nombre de tentatives d'appel de ce numéro si l'appel échoue.

```
short GetRedialCount();  
void SetRedialCount(short nNewValue);
```

Délai en secondes entre les tentatives d'appel.

```
short GetRedialDelay();  
void SetRedialDelay(short nNewValue);
```

Contrôle si les sessions utilisant cet objet sont consignées.

```
BOOL GetLogSession();  
void SetLogSession(BOOL bNewValue);
```

Contrôle si les sessions utilisant cet objet sont enregistrées depuis le début.

```
BOOL GetRecordSession();  
void SetRecordSession(BOOL bNewValue);
```

Nom du fichier d'enregistrement pour les sessions utilisant cet objet.

```
BSTR GetRecordFile();  
void SetRecordFile(LPCTSTR lpszNewValue);
```

Méthodes de détail d'objet Elève

Lorsqu'un type de connexion est affecté à un objet Elève, les détails du périphérique sont définis sur des valeurs par défaut valides. Les types de connexion suivants possèdent des options de configuration avancées qui peuvent être définies dans l'application :

- Périphériques COM
- Périphériques de passerelle réseau (TCP/IP, SPX)
Les périphériques de passerelle ne sont pris en charge que dans pcAnywhere 9.2x.
- Périphériques NetBIOS
- Périphériques NASI/NCSI
- Périphériques RNIS via CAPI 2.0 (RNIS européen seulement)

Détails de périphérique COM

Le code suivant place le type de connexion requis dans la liste des types de connexion affectés à l'objet Elève et en fait le type de connexion en cours lors du traitement ultérieur des appels de méthodes propres au périphérique.

```
BOOL AssignConnection(LPCTSTR lpszNewValue);
```

Si le type de connexion requis se trouve déjà dans la liste des connexions affectées, celle-ci ne change pas. Seul le type de connexion actuel est remplacé par le type demandé. Il est normal d'appeler la méthode AssignConnection sur le même objet plusieurs fois lors de l'obtention et de la définition de valeurs propres à la connexion.

AssignConnection renvoie VRAI si le type de connexion transmis existe sur l'ordinateur et est affecté avec succès ou déjà affecté. Il renvoie FAUX si le type de connexion requis n'existe pas sur l'ordinateur ou si le nombre de connexions affectées a déjà atteint le maximum.

Un objet Elève pcAnywhere peut prendre en charge jusqu'à deux types de connexion affectés. La méthode AssignConnection renvoie FAUX si elle détecte une tentative de dépassement de cette limite.

Les types de connexion suivants sont disponibles :

- COM1
- COM2
- COM3
- COM4
- SPX
- NetBIOS
- TCP/IP
- LPT1
- LPT2
- LPT3
- LPT4
- RNIS via CAPI 2.0
- Infrarouge
- DEFAULT TAPI

Le nom d'un périphérique TAPI peut également servir de type de connexion. DEFAULT TAPI utilise le premier périphérique TAPI trouvé dans le système.

Pour utiliser un périphérique TAPI spécifique, utilisez FirstConnectionType() et NextConnectionType() pour rechercher les périphériques disponibles.

Le code suivant désaffecte un type de connexion. Après avoir désaffecté un type de connexion, la connexion affectée restante, s'il y en a une, devient le type de connexion actuel pour les appels ultérieurs de méthode propres au périphérique.

```
BOOL UnassignConnection(LPCTSTR lpszNewValue);
```

Le code suivant définit la parité des communications :

```
BSTR GetComParity();  
void SetComParity(LPCTSTR lpszNewValue);
```

Les valeurs de parité de communication sont les suivantes :

- Aucune
- Impaire
- Paire
- Marque
- Espace

Le code suivant définit les niveaux de contrôle de flux :

```
BSTR GetComFlowControl();void SetComFlowControl(LPCTSTR  
lpszNewValue);
```

Les valeurs de contrôle de flux sont les suivantes :

- <Aucun>
- XONXOFF
- RTS/CTS
- LES DEUX

Le code suivant définit les valeurs de début de connexion :

```
BSTR GetComStartedBy();  
void SetComStartedBy(LPCTSTR lpszNewValue);
```

Les valeurs de début de connexion sont les suivantes :

- Toujours connecté
- Détection de porteuse (DCD)
- Prêt à émettre (CTS)
- Ensemble de données prêt (DSR)
- Signal de sonnerie (RI)
- Réception 2 <RC>
- Réponse du modem

Le code suivant définit les valeurs de fin de connexion :

```
BSTR GetComEndedBy();  
void SetComEndedBy(LPCTSTR lpszNewValue);
```

Les valeurs de fin de connexion sont les suivantes :

- Toujours connecté
- Détection de porteuse (DCD)
- Prêt à émettre (CTS)
- Ensemble de données prêt (DSR)
- Signal de sonnerie (RI)

Le code suivant définit la vitesse de connexion :

```
long GetComSpeed();  
void SetComSpeed(long nNewValue);
```

Les valeurs de vitesse de connexion sont les suivantes :

- 110
- 300
- 600
- 1200
- 2400
- 4800
- 9600
- 19200
- 38400
- 57600
- 115200

Détails de périphérique réseau (TCP/IP, SPX) pour les passerelles

Vous pouvez utiliser les propriétés suivantes avec les connexions pcAnywhere 9.2x uniquement :

Se connecter par une passerelle pcAnywhere (VRAI).

```
BOOL GetGatewayUse();  
void SetGatewayUse(BOOL bNewValue);
```

Nom de la passerelle pcAnywhere à utiliser.

```
BSTR GetGatewayName();  
void SetGatewayName(LPCTSTR lpszNewValue);
```

Classe de passerelle pcAnywhere à utiliser.

```
BSTR GetGatewayClass();  
void SetGatewayClass(LPCTSTR lpszNewValue);
```

Parité de passerelle pcAnywhere à utiliser.

```
BSTR GetGatewayParity();  
void SetGatewayParity(LPCTSTR lpszNewValue);
```

Les valeurs de parité sont les suivantes :

- <Aucune>
- Impaire
- Paire
- Marque
- Espace

Détails de périphérique NetBios

Définit le numéro de LANA (adaptateur réseau) à utiliser pour cette connexion.

```
short GetLanaNumber(); void SetLanaNumber(short nNewValue);
```

Détails de périphérique NASI/NCSI

Définit le nom de l'utilisateur pour le serveur NASI.

```
BSTR GetNasiUserName();  
void SetNasiUserName(LPCTSTR lpszNewValue);
```

Définit le mot de passe de l'utilisateur pour le serveur NASI.

```
BSTR GetNasiPassword();  
void SetNasiPassword(LPCTSTR lpszNewValue);
```

Définit le nom de session NASI.

```
BSTR GetNasiSessionName();  
void SetNasiSessionName(LPCTSTR lpszNewValue);  
BOOL GetNasiSessionNameAvailable();  
void SetNasiSessionNameAvailable(BOOL bNewValue);
```


Spécifie le serveur NASI à utiliser.

```
BOOL NasiServer();  
BSTR GetNasiServerName();  
void SetNasiServerName(LPCTSTR lpszNewValue);
```

Spécifie le service NASI à utiliser.

```
BOOL NasiService();  
BSTR GetNasiServiceName();  
void SetNasiServiceName(LPCTSTR lpszNewValue);
```

Spécifie le port NASI à utiliser.

```
BOOL NasiPort();  
BSTR GetNasiPortName();  
void SetNasiPortName(LPCTSTR lpszNewValue);  
BOOL GetNasiSelectOnConnect();  
void SetNasiSelectOnConnect(BOOL bNewValue);
```

Détails de périphérique RNIS via CAPI 2.0

Active la liaison de canaux (utilise deux canaux RNIS pour une connexion) si VRAI.

```
BOOL GetCapiChannelBonding();  
void SetCapiChannelBonding(BOOL bNewValue);
```

Définit toute extension CAPI supplémentaire nécessaire pour les communications.

```
BSTR GetCapiExtensions();  
void SetCapiExtensions(LPCTSTR lpszNewValue);
```

Méthodes d'objet Elève

Les méthodes suivantes sont les méthodes normales de l'objet (elles ne servent pas à obtenir et à définir les propriétés de l'objet).

- `short ConnectionTypes();`
- `BSTR FirstConnectionType();` et `BSTR NextConnectionType();`
- `BOOL FindConnectionType(LPCTSTR lpszConnectionType);`
- `short MaxAssignedConnections()`
- `short AssignedConnections()`
- `BSTR FirstAssignedConnection();` et `BSTR NextAssignedConnection ();`
- `BOOL FindAssignedConnection (LPCTSTR lpszConnectionType);`

- `short CountryCodes();`
- `BSTR FirstCountryCode();` et `BSTR NextCountryCode();`
- `BOOL ReadObject(LPCTSTR lpszPassword);`
- `BOOL WriteObject(LPCTSTR lpszPassword);`

short ConnectionTypes();

Renvoie le nombre de types de connexion disponibles.

Le [Tableau 3-49](#) définit la valeur renvoyée.

Tableau 3-49 `short ConnectionTypes();` - Valeur renvoyée

Valeur renvoyée	Description
Courte	Nombre de types de connexion disponibles sur cet ordinateur

BSTR FirstConnectionType(); et BSTR NextConnectionType();

`FirstConnectionType()` et `NextConnectionType()` servent à itérer parmi les types de connexion disponibles. Ces fonctions renvoient un BSTR qui est le nom d'un type de connexion disponible. Les types de connexion renvoyés peuvent être utilisés avec la fonction `SetConnectionType()`.

Le [Tableau 3-50](#) définit la valeur renvoyée.

Tableau 3-50 `BSTR FirstConnectionType();` et `BSTR NextConnectionType();` - Valeur renvoyée

Valeur renvoyée	Description
BSTR	Nom d'un type de périphérique de connexion pris en charge

BOOL FindConnectionType(LPCTSTR lpszConnectionType);

Renvoie VRAI si le type de connexion transmis existe sur l'ordinateur.

Le [Tableau 3-51](#) définit le paramètre.

Tableau 3-51 `BOOL FindConnectionType` - Paramètre

Paramètre	Description
LPCTSTR lpszConnectionType	Nom d'un type de périphérique de connexion

Le [Tableau 3-52](#) définit la valeur renvoyée.

Tableau 3-52 BOOL FindConnectionType - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si ce type de périphérique est disponible.

short MaxAssignedConnections()

Renvoie le nombre maximum de types de connexion qui peuvent être affectés simultanément (actuellement deux).

Le [Tableau 3-53](#) définit la valeur renvoyée.

Tableau 3-53 short MaxAssignedConnections() - Valeur renvoyée

Valeur renvoyée	Description
Courte	Nombre maximum d'affectations de type de connexion.

short AssignedConnections()

Renvoie le nombre de types de connexion affectés.

Le [Tableau 3-54](#) définit la valeur renvoyée.

Tableau 3-54 short AssignedConnections() - Valeur renvoyée

Valeur renvoyée	Description
Courte	Nombre de types de connexion affectés sur cet ordinateur.

BSTR FirstAssignedConnection(); et BSTR NextAssignedConnection ();

FirstAssignedConnection() et NextAssignedConnection() servent à itérer parmi les connexions affectées. Ces fonctions renvoient un BSTR, qui est le nom d'un type de connexion affecté. Les types de connexion renvoyés peuvent être utilisés avec la fonction AssignConnection().

Le [Tableau 3-55](#) définit la valeur renvoyée.

Tableau 3-55 BSTR FirstAssignedConnection(); et BSTR
NextAssignedConnection (); - Valeur renvoyée

Valeur renvoyée	Description
BSTR	Nom d'un type de périphérique de connexion pris en charge.

**BOOL FindAssignedConnection
(LPCTSTR lpszConnectionType);**

Renvoie VRAI si le type de connexion transmis est actuellement affecté sur l'ordinateur.

Le [Tableau 3-56](#) définit le paramètre.

Tableau 3-56 BOOL FindAssignedConnection - Paramètre

Paramètre	Description
LPCTSTR lpszConnectionType	Nom d'un type de périphérique de connexion.

Le [Tableau 3-57](#) définit la valeur renvoyée.

Tableau 3-57 BOOL FindAssignedConnection - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si ce type de périphérique est actuellement affecté.

short CountryCodes();

Renvoie le nombre d'indicatifs internationaux disponibles.

Le [Tableau 3-58](#) définit la valeur renvoyée.

Tableau 3-58 short CountryCodes(); - Valeur renvoyée

Valeur renvoyée	Description
Courte	Nombre d'indicatifs internationaux disponibles.

BSTR FirstCountryCode(); et BSTR NextCountryCode();

FirstCountryCode() et NextCountryCode() servent à itérer parmi la liste des indicatifs internationaux disponibles. Les fonctions renvoient un BSTR qui est le nom d'un indicatif international disponible. Les valeurs renvoyées peuvent être utilisées avec la fonction SetCountryCode().

Le [Tableau 3-59](#) définit la valeur renvoyée.

Tableau 3-59 BSTR FirstCountryCode(); et BSTR NextCountryCode(); - Valeur renvoyée

Valeur renvoyée	Description
BSTR	Première chaîne d'indicatif international ou chaîne suivante.

BOOL ReadObject(LPCTSTR lpszPassword);

Lit les données d'objet à partir du fichier d'objet Elève.

Le [Tableau 3-60](#) définit le paramètre.

Tableau 3-60 BOOL ReadObject - Paramètre

Paramètre	Description
LPCTSTR lpszPassword	Mot de passe de l'objet

Le [Tableau 3-61](#) définit la valeur renvoyée.

Tableau 3-61 BOOL ReadObject - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est lu avec succès.

BOOL WriteObject(LPCTSTR lpszPassword);

Ecrit les données d'objet dans le fichier d'objet Elève.

Le [Tableau 3-62](#) définit le paramètre.

Tableau 3-62 BOOL WriteObject - Paramètre

Paramètre	Description
LPCTSTR lpszPassword	Mot de passe de l'objet

Le [Tableau 3-63](#) définit la valeur renvoyée.

Tableau 3-63 BOOL WriteObject - Valeur renvoyée

Valeur renvoyée	Description
BOOL	VRAI si l'objet est écrit avec succès.

Objet CHostDataEx

L'objet CHostDataEx contient la même fonctionnalité que l'objet CHostData, avec les méthodes supplémentaires Get et Set (obtenir et définir) suivantes :

```
BOOL GetReadProtection();  
void SetReadProtection(BOOL bNewValue);  
BOOL GetWriteProtection();  
void SetWriteProtection(BOOL bNewValue);  
BSTR GetPassword(); //Renvoie "NON IMPLEMENTE"  
void SetPassword(LPCTSTR lpszNewValue);  
BSTR GetCallersPath();  
void SetCallersPath(LPCTSTR lpszNewValue);  
BOOL GetConfirmConnect();  
void SetConfirmConnect(BOOL bNewValue);  
short GetConfirmTimeout();  
void SetConfirmTimeout(short nNewValue);  
BOOL GetConfirmDeny();  
void SetConfirmDeny(BOOL bNewValue);  
BOOL GetPwCaseSensitive();  
void SetPwCaseSensitive(BOOL bNewValue);  
short GetPwAttempts();  
void SetPwAttempts(short nNewValue);  
short GetPwTimeout();  
void SetPwTimeout(short nNewValue);  
short GetActiveKbds();  
void SetActiveKbds(short nNewValue); //Sets ActiveKbds  
short GetInactiveTimeout();  
void SetInactiveTimeout(short nNewValue);  
short GetCryptReqLevel();  
void SetCryptReqLevel(short nNewValue);  
BOOL GetCryptRefuseLower();  
void SetCryptRefuseLower(BOOL bNewValue);  
short GetAuthenticationType();  
void SetAuthenticationType(short nNewValue);  
BOOL GetLockSystemWhileWait();  
void SetLockSystemWhileWait(BOOL bNewValue);  
BOOL GetMinimizeOnLaunch();
```

```
void SetMinimizeOnLaunch(BOOL bNewValue);
BOOL GetRunAsService();
void SetRunAsService(BOOL bNewValue);
short GetConnLostWait();
void SetConnLostWait(short nNewValue);
BOOL GetConnLostHostOpts();
void SetConnLostHostOpts(BOOL bNewValue);
BOOL GetEnableConnLostSecurity();
void SetEnableConnLostSecurity(BOOL bNewValue);
short GetConnLostSecurity();
void SetConnLostSecurity(short nNewValue);
short GetCallbkDelay();
void SetCallbkDelay(short nNewValue);
BOOL GetEndSessHostOpts();
void SetEndSessHostOpts(BOOL bNewValue);
BOOL GetEnableEndSessSecurity();
void SetEnableEndSessSecurity(BOOL bNewValue);
short GetEndSessSecurity();
void SetEndSessSecurity(short nNewValue);
BSTR GetCryptPrivateKey();
void SetCryptPrivateKey(LPCTSTR lpszNewValue);
BSTR GetCryptCommonName();
void SetCryptCommonName(LPCTSTR lpszNewValue);
BOOL GetBlankHost();
void SetBlankHost(BOOL bNewValue);
BOOL GetAllowRemoteMouse();
void SetAllowRemoteMouse(BOOL bNewValue);
short GetRebootOnDisconnect();
void SetRebootOnDisconnect(short nNewValue);
BOOL GetPasswordAfterDisc();
void SetPasswordAfterDisc(BOOL bNewValue);
BOOL GetLogFailures();
void SetLogFailures(BOOL bNewValue);
BOOL GetAllowDriveSecurity();
void SetAllowDriveSecurity(BOOL bNewValue);
BOOL GetExecuteProtection();
void SetExecuteProtection(BOOL bNewValue);
```

Exemple de code Visual C++ pour la fonction Elève

L'exemple suivant de fonction Visual C++ crée un objet Elève, définit son type de connexion sur TCP/IP, définit le nom de l'ordinateur sur l'adresse TCP/IP transmise à la fonction, puis lance l'objet Elève.

```
BOOL LaunchTCPHost(LPCTSTR lpszAddress)
{
    BOOL bReturn = FALSE;

    CHostDataManager hostDM;
    CHostData hostData;

    // Créer d'abord le CHostDataManager
    hostDM.CreateDispatch( _T( "WINAWSVR.BeHostDataManager" ) );

    // Créer ensuite CRemoteData et le rattacher
    hostData.AttachDispatch(hostDM.CreateObject("Test", 0) );

    // Définir à présent les propriétés requises
    hostData.SetConnectionType("TCP/IP");

    // Enregistrer les données de l'objet
    if (hostData.WriteObject(0))
    {
        // Et le lancer
        if (hostData.Launch())
            bReturn = TRUE;
    }

    // Libérer l'objet Elève.
    hostData.ReleaseDispatch();

    return (bReturn);
}
```


Fonctions awrem32

Les fonctions Awrem32 fournissent des paramètres et renvoient des valeurs pour la gestion des connexions entre un ordinateur Elève et un ordinateur Maître.

boolean awConnect(BSTR FileName);

Crée la connexion à l'ordinateur Elève.

Le [Tableau 3-64](#) définit le paramètre.

Tableau 3-64 boolean awConnect - Paramètre

Paramètre	Description
Nom comme chaîne	Nom intégralement qualifié du fichier .chf contenant les informations sur l'ordinateur Elève.

Le [Tableau 3-65](#) définit la valeur renvoyée.

Tableau 3-65 boolean awConnect - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	Exécute la commande

boolean awDisconnect();

Déconnecte l'ordinateur Elève.

Le [Tableau 3-66](#) définit la valeur renvoyée.

Tableau 3-66 boolean awDisconnect(); - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	Après avoir appelé cette fonction, le programme appelant doit supprimer l'objet (C++ - delete IAwrem32X*, VB – set ObjectName = Nothing;)

boolean FileXferFromHost(BSTR HostFile, BSTR RemoteFile);

Copie un fichier de l'ordinateur Elève vers l'ordinateur Maître. Les paramètres peuvent contenir des caractères génériques.

Le [Tableau 3-67](#) définit les paramètres.

Tableau 3-67 boolean FileXferFromHost - Paramètres

Paramètre	Description
Fichier Elève comme chaîne	Contient le chemin d'accès intégralement qualifié du fichier à copier à partir de l'ordinateur Elève.
Fichier Maître comme chaîne	Contient le chemin d'accès de destination, intégralement qualifié. Les chaînes de fichier Elève et de fichier Maître ne sont pas obligatoirement identiques.

Le [Tableau 3-68](#) définit la valeur renvoyée.

Tableau 3-68 boolean FileXferFromHost - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

boolean FileXferToHost(BSTR HostFile, BSTR RemoteFile);

Copie un fichier de l'ordinateur Maître vers l'ordinateur Elève. Les paramètres peuvent contenir des caractères génériques.

Le [Tableau 3-69](#) définit les paramètres.

Tableau 3-69 boolean FileXferToHost - Paramètres

Paramètre	Description
Fichier Elève comme chaîne	Contient le chemin d'accès de destination, intégralement qualifié.
Fichier Maître comme chaîne	Contient le chemin d'accès intégralement qualifié du fichier à copier à partir de l'ordinateur Maître. Les chaînes de fichier Elève et de fichier Maître ne sont pas obligatoirement identiques.

Le [Tableau 3-70](#) définit la valeur renvoyée.

Tableau 3-70 boolean FileXferToHost - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

boolean CreateFolderOnHost(Nom de dossier BSTR);

Crée un nouveau dossier sur l'ordinateur Elève. Cette fonction crée un dossier temporaire sur l'ordinateur Maître, puis copie ce dossier sur l'ordinateur Elève.

Le [Tableau 3-71](#) définit le paramètre.

Tableau 3-71 boolean CreateFolderOnHost - Paramètre

Paramètre	Description
Nom de dossier comme chaîne	Contient le lecteur et le chemin d'accès du dossier sur l'ordinateur Elève.

Le [Tableau 3-72](#) définit la valeur renvoyée.

Tableau 3-72 boolean CreateFolderOnHost - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

boolean ExecuteHostFile(BSTR FileName);

Exécute un fichier existant sur l'ordinateur Elève. Cette fonction ne peut exécuter que les fichiers batch, de commandes et exécutables. Elle n'exécute pas les fichiers associés aux exécutables. Par exemple, elle n'ouvre pas Microsoft Word si vous exécutez un fichier .doc.

Le [Tableau 3-73](#) définit le paramètre.

Tableau 3-73 boolean ExecuteHostFile - Paramètre

Paramètre	Description
Nom de fichier comme chaîne	Contient le chemin d'accès intégralement qualifié du fichier sur l'ordinateur Elève.

Le [Tableau 3-74](#) définit la valeur renvoyée.

Tableau 3-74 boolean ExecuteHostFile - Valeur renvoyée

Valeur renvoyée	Description
Booléenne	VRAI si la commande est exécutée.

BSTR GetError();

Renvoie la dernière erreur sous forme de chaîne.

Le [Tableau 3-75](#) définit la valeur renvoyée.

Tableau 3-75 BSTR GetError(); - Valeur renvoyée

Valeur renvoyée	Description
Chaîne	Renvoie la dernière erreur générée dans Awrem32.

short ConnectionStatus();

Renvoie l'état en cours de la connexion à l'ordinateur Elève.

Le [Tableau 3-76](#) définit la valeur renvoyée.

Tableau 3-76 short ConnectionStatus(); - Valeur renvoyée

Valeur renvoyée	Description
Courte	<p>Les valeurs possibles sont les suivantes :</p> <ul style="list-style-type: none"> ■ -1 = Connexion perdue ■ 0 = Aucune connexion ■ 1 = Session connectée

Index

A

automatisation OLE *Voir* serveur
d'automatisation pcAnywhere
Awrem32
 bibliothèque de type 12
 fonctions 62, 105

B

bibliothèques API 12
bibliothèques de type 12

C

C++. *Voir* Visual C++
CHostData
 utilisation de Visual Basic 41
 utilisation de Visual C++ 91
CHostDataEx
 utilisation de Visual Basic 53
 utilisation de Visual C++ 102
CHostDataManager
 utilisation de Visual Basic 36
 utilisation de Visual C++ 85
connexions
 arrêt 62, 105
 erreurs de retour 108
 lancement 41, 62, 74, 105
 retour de l'état 65, 108
connexions CAPI
 propriétés Elève 52, 97
 propriétés Maître 32, 80
connexions NetBIOS
 propriétés Elève 51, 96
 propriétés Maître 31, 80

connexions par modem
 sur l'Elève
 indicatifs internationaux 100
 périphériques TAPI 93
 propriétés COM 48, 93
 propriétés de numérotation 47
 sur le Maître
 indicatifs internationaux 82
 périphériques TAPI 77
 propriétés COM 28, 78
 propriétés de numérotation 27
connexions réseau
 sur l'Elève
 propriétés de la passerelle 52, 95
 propriétés NetBIOS 51, 96
 sur le Maître
 propriétés de la passerelle 32, 79
 propriétés NetBIOS 31, 80
connexions RNIS CAPI
 propriétés Elève 52, 97
 propriétés Maître 32, 80
contrôle de flux 28, 49, 78, 94
contrôleurs d'automatisation
 à propos de Visual Basic 10
 à propos de Visual C++ 11
CRemoteData
 utilisation de Visual Basic 22
 utilisation de Visual C++ 75
CRemoteDataEx
 utilisation de Visual Basic 33
 utilisation de Visual C++ 84
CRemoteDataManager
 utilisation de Visual Basic 17
 utilisation de Visual C++ 69

D

définition de classes
 affichage 13
 importation 12
 dossiers, création sur l'Elève 64, 107

E

erreurs, retour 65, 108
 état, retour 65, 108
 exemples de code
 Visual Basic 34, 60
 Visual C++ 84, 104

F

fichiers, exécution sur l'Elève 64, 107

G

GUID 9

I

identificateurs. *Voir* GUID
 indicatifs internationaux
 sur l'Elève 100
 sur le Maître 82

M

mot de passe
 sur les objets Maître 83
 moteur Maître
 inscription automatique 9
 inscription manuelle 10
 mots de passe
 sur les objets Elève 101

O

objets Elève
 création 39, 89
 détails du périphérique 92
 lancement 41, 91
 méthodes 97
 mots de passe 101
 propriétés de numérotation 47
 recherche 37–39, 86–88
 répertoires 36, 85
 suppression 40, 90

objets Maître

 création 73
 détails du périphérique 77
 fichiers 18–20, 37, 70–72
 méthodes 81
 mots de passe 83
 propriétés de numérotation 27
 répertoires 17, 69
 suppression 21, 74

P

paramètres de chiffrement
 sur l'Elève 53, 102
 sur le Maître 33, 34, 84
 parité 28, 48, 78, 94
 périphériques COM
 détails de périphérique
 Elève 48, 93
 détails de périphérique
 Maître 28, 78
 périphériques NASI 96
 périphériques NCSI 96
 périphériques TAPI
 sur l'Elève 93
 sur le Maître 77
 périphériques. *Voir* types de connexion
 propriétés de la passerelle 32, 52,
 79, 95

S

serveur d'automatisation pcAnywhere
 à propos 8
 accès avec Visual Basic 10
 accès avec Visual C++ 11
 bibliothèques de type 12
 exemples d'utilisation 8
 inscription des GUID 9
 serveur d'automatisation. *Voir* serveur
 d'automatisation pcAnywhere

T

transfert de fichiers
 à partir de l'Elève 63, 106
 vers l'Elève 63, 106
 types de connexion
 attribution 46
 désaffectation 47
 propriétés Elève 44, 92, 97
 propriétés Maître 25, 77, 81

V

Visual Basic

- accès au serveur d'automatisation
 - pcAnywhere 10
- exemples de code 34, 60
- fonctions awrem32 62
- objet CHostData 41
- objet CHostDataEx 53
- objet CHostDataManager 36
- objet CRemoteData 22
- objet CRemoteDataEx 33
- objet CRemoteDataManager 17

Visual C++

- accès au serveur d'automatisation
 - pcAnywhere 11
- ajout de Winawsvr.h 13
- exemples de code 84, 104
- fonctions awrem32 105
- importation de classes 12
- objet CHostData 91
- objet CHostDataEx 102
- objet CHostDataManager 85
- objet CRemoteData 75
- objet CRemoteDataEx 84
- objet CRemoteDataManager 69

W

Winawsvr

- bibliothèque de type 12
- exécutable 10
- fichier d'en-tête 13
- objets
 - CHostData 41, 91
 - CHostDataEx 53, 102
 - CHostDataManager 36, 85
 - CRemoteData 22, 75
 - CRemoteDataEx 33, 84
 - CRemoteDataManager 17, 69

Solutions de service et de support

Symantec se consacre à fournir un excellent service dans le monde entier. Notre objectif est de vous apporter une assistance professionnelle pour utiliser nos logiciels et nos services, où que vous vous trouviez.

Les solutions de support technique et de service clientèle varient selon les pays.

Si vous avez des questions sur les services décrits ci-dessous, consultez la section « Informations de service et de support dans le monde ».

Enregistrement et licences

Si vous déployez un produit qui nécessite un enregistrement et/ou une clé de licence, le système le plus rapide et le plus simple consiste à accéder à notre site de licence et d'enregistrement (en anglais) à l'adresse www.symantec.com/certificate.

Si vous avez acheté un abonnement de support, vous êtes habilité à bénéficier d'un support technique par téléphone et sur Internet. Lorsque vous contactez les services de support pour la première fois, vous devez disposer du numéro de votre certificat de licence ou de l'identification de contact fournie lors de l'enregistrement, pour permettre la vérification de vos droits au support. Si vous n'avez pas acheté d'abonnement de support, contactez votre revendeur ou le service clientèle de Symantec pour savoir comment obtenir un support technique auprès de Symantec.

Mises à jour de la sécurité

Pour obtenir les informations les plus récentes sur les virus et les menaces de sécurité, visitez le site de Symantec Security Response (anciennement SARC - Centre de Recherche AntiVirus de Symantec), à l'adresse

<http://www.symantec.fr/region/fr/avcenter/index.html>

Ce site contient des informations exhaustives sur la sécurité et les virus, ainsi que les dernières définitions de virus. Vous pouvez également télécharger les définitions de virus en utilisant la fonction LiveUpdate de votre produit.

Renouvellement d'abonnement aux définitions de virus

Votre achat d'un service de support avec un produit vous permet de télécharger gratuitement des définitions de virus pendant la durée de l'abonnement. Si votre abonnement au support a expiré, contactez votre revendeur ou le Service clientèle de Symantec pour savoir comment le renouveler.

Sites Web Symantec :

Page d'accueil Symantec (par langue) :

Allemand :	http://www.symantec.de
Anglais :	http://www.symantec.com
Espagnol :	http://www.symantec.com/region/es
	http://www.symantec.com/mx
Français :	http://www.symantec.fr
Italien :	http://www.symantec.it
Néerlandais :	http://www.symantec.nl
Portugais :	http://www.symantec.com/br

Symantec Security Response :

<http://www.symantec.fr/region/fr/avcenter/index.html>

Page de service et assistance Symantec :

<http://www.symantec.com/region/fr/techsupp/enterprise/index.html>

Bulletin d'informations spécifique produit :

Etats-Unis, Asie-Pacifique :

<http://www.symantec.com/techsupp/bulletin/index.html>

Europe, Moyen-Orient, Afrique/Anglais :

http://www.symantec.com/region/reg_eu/techsupp/bulletin/index.html

Allemand :

<http://www.symantec.com/region/de/techsupp/bulletin/index.html>

Français :

<http://www.symantec.com/region/fr/techsupp/bulletin/index.html>

Italien :

<http://www.symantec.com/region/it/techsupp/bulletin/index.html>

Néerlandais :

<http://www.symantec.com/region/nl/techsupp/bulletin/index.html>

Amérique latine

Espagnol :

<http://www.symantec.com/region/mx/techsupp/bulletin/index.html>

Portugais :

<http://www.symantec.com/region/br/techsupp/bulletin/index.html>

Support technique

Au sein de Symantec Security Response, l'équipe de support technique internationale gère les centres d'assistance dans le monde entier. Son objectif premier est de répondre aux questions spécifiques sur les fonctionnalités/fonctions, l'installation et la configuration des produits Symantec ainsi que sur le contenu de la Base de connaissances accessible via le Web. Symantec Security Response est en collaboration étroite avec les autres départements de Symantec pour répondre rapidement à vos questions. Nous travaillons par exemple avec notre service d'ingénierie produit et nos centres de recherche en sécurité pour fournir des services d'alertes et des mises à jour des définitions de virus, face aux attaques virales et aux alertes de sécurité. Caractéristiques de nos offres :

- Une panoplie d'options de support vous permet de choisir le service approprié quel que soit le type d'entreprise.
- Le support Web et téléphonique fournit des réponses rapides et des informations de dernière minute.
- Les mises à jour des produits fournissent une protection de mise à niveau automatique
- Les mises à jour de contenu des définitions de virus et les signatures de sécurité assurent la meilleure protection.
- Le support mondial des experts Symantec Security Response est disponible 24h/24, 7j/7 dans le monde entier et dans différentes langues.
- Les fonctionnalités avancées telles que le Service d'alertes Symantec (Symantec Alerting Service) et le Responsable de compte technique (Technical Account Manager) offrent un support d'intervention et de sécurité proactive.

Rendez-vous sur notre site Web pour obtenir les dernières informations sur les programmes de support.

Coordonnées du support

Les clients disposant d'un contrat de support peuvent contacter l'équipe de support technique par téléphone, sur le site Web suivant ou sur les sites régionaux de Service et Support internationaux.

<http://www.symantec.com/region/fr/techsupp/enterprise/index.html>

Lorsque vous contactez le support, vérifiez que vous disposez des informations suivantes :

- Version du produit
- Informations sur le matériel
- Mémoire disponible, espace disque et informations sur la carte d'interface réseau
- Système d'exploitation
- Niveau de version et correctif
- Topologie du réseau
- Informations sur le routeur, la passerelle et l'adresse IP
- Description du problème
- Messages d'erreur/fichiers journaux
- Intervention effectuée avant de contacter Symantec
- Modifications récentes de la configuration du logiciel ou du réseau

Service clientèle

Le Centre de service clientèle de Symantec peut vous seconder pour vos questions non techniques :

- Informations générales sur les produits (caractéristiques, langues disponibles, adresse des distributeurs, etc)
- Dépannage de base, par exemple vérification de la version du produit
- Dernières informations sur les mises à jour produit
- Comment mettre votre produit à jour/à niveau
- Comment enregistrer votre produit et/ou votre licence
- Informations sur les programmes de licences de Symantec
- Informations sur les contrats de mise à niveau et de maintenance
- Remplacement des CD et des manuels
- Mise à jour des données d'enregistrement produit en cas de changement de nom ou d'adresse
- Conseil sur les options de support technique de Symantec

Des informations détaillées sur le Service clientèle sont disponibles sur le site Web de l'assistance Symantec. Vous pouvez également contacter le Centre de service clientèle par téléphone. Pour des informations sur les numéros de support clientèle et les sites Web, consultez la section « Informations de service et de contact en bref ».

Service et support internationaux

Europe, Moyen-Orient, Afrique et Amérique latine

Sites Web de service et assistance Symantec

Allemand :	www.symantec.de/desupport/
Anglais :	www.symantec.com/eusupport/
Espagnol :	www.symantec.com/region/mx/techsupp/
Français :	www.symantec.fr/frsupport
Italien :	www.symantec.it/itsupport/
Néerlandais :	www.symantec.nl/nlsupport/
Portugais :	www.symantec.com/region/br/techsupp/
FTP Symantec :	ftp.symantec.com
(téléchargement des notes techniques et des derniers correctifs)	

Visitez le site Service et assistance de Symantec pour trouver des informations techniques et non techniques sur votre produit.

Symantec Security Response :

<http://securityresponse.symantec.com>

Bulletin d'informations spécifique produit :

Anglais :

<http://www.symantec.com/techsupp/bulletin/index.html>

Europe, Moyen-Orient, Afrique/Anglais :

http://www.symantec.com/region/reg_eu/techsupp/bulletin/index.html

Allemand :

<http://www.symantec.com/region/de/techsupp/bulletin/index.html>

Français :

<http://www.symantec.com/region/fr/techsupp/bulletin/index.html>

Italien :

<http://www.symantec.com/region/it/techsupp/bulletin/index.html>

Espagnol :

<http://www.symantec.com/region/mx/techsupp/bulletin/index.html>

Néerlandais :

<http://www.symantec.com/region/nl/techsupp/bulletin/index.html>

Portugais :

<http://www.symantec.com/region/br/techsupp/bulletin/index.html>

Service Clientèle de Symantec

Fournit des informations non techniques et des conseils par téléphone dans les langues suivantes : anglais, allemand, français et italien.

Autriche	+ (43) 1 50 137 5030
Belgique	+ (32) 2 2750173
Danemark	+ (45) 35 44 57 04
Espagnol	+ (34) 91 7456467
Finlande	+ (358) 9 22 906003
France	+ (33) 1 70 20 00 00
Allemagne	+ (49) 69 6641 0315
Irlande	+ (353) 1 811 8093
Italie	+ (39) 02 48270040
Luxembourg	+ (352) 29 84 79 50 30
Pays-Bas	+ (31) 20 5040698
Norvège	+ (47) 23 05 33 05
Afrique du Sud	+ (27) 11 797 6639
Suède	+ (46) 8 579 29007
Suisse	+ (41) 2 23110001
Royaume Uni	+ (44) 20 7744 0367
Autres pays	+ (353) 1 811 8093
(service en anglais uniquement)	

Service Clientèle Symantec – Adresse postale

Symantec Ltd
 Customer Service Centre
 Europe, Moyen-Orient et Afrique (EMEA)
 PO Box 5689
 Dublin 15
 Irlande

En Amérique latine

Symantec dispose d'un support technique et d'un service clientèle internationaux. Les services varient selon les pays et incluent des partenaires internationaux qui représentent Symantec dans les régions où il n'y a pas de bureau Symantec. Pour des informations générales, contactez le service de support de Symantec pour votre région.

Argentine

Pte. Roque Saenz Peña 832 - Piso 6
C1035AAQ, Ciudad de Buenos Aires
Argentina

Numéro principal	+54 (11) 5811-3225
Site Web	http://www.service.symantec.com/mx
Support Gold	0800-333-0306

Venezuela

Avenida Francisco de Miranda. Centro Lido
Torre D. Piso 4, Oficina 40
Urbanización el Rosal
1050, Caracas D.F.
Venezuela

Numéro principal	+58 (212) 905-6327
Site Web	http://www.service.symantec.com/mx
Support Gold	0800-1-00-2543

Colombia

Carrera 18# 86A-14
Oficina 407, Bogota D.C.
Colombia

Numéro principal	+57 (1) 638-6192
Site Web	http://www.service.symantec.com/mx
Support Gold	980-915-5241

Brésil

Symantec Brésil
Market Place Tower
Av. Dr. Chucri Zaidan, 920
12° andar
São Paulo - SP
CEP: 04583-904
Brésil, SA

Numéro principal	+55 (11) 5189-6300
Télécopie	+55 (11) 5189-6210
Site Web	http://www.service.symantec.com/br
Support Gold	000814-550-4172

Chile

Alfredo Barros Errazuriz 1954
Oficina 1403
Providencia,
Santiago de Chile
Chile

Numéro principal	+56 (2) 378-7480
Site Web	http://www.service.symantec.com/mx
Support Gold	0800-333-0306

Mexique

Boulevard Adolfo Ruiz Cortines 3642 Piso 8,
Colonia Jardines del Pedregal,
01900, Mexico D.F.
Mexico

Numéro principal	+52 (55) 5481-2600
Site Web	http://www.service.symantec.com/mx
Support Gold	001880-232-4615

Reste de l'Amérique latine

9155 South Dadeland Blvd.
Suite 1100,
Miami, FL 33156
U.S.A

Site Web	http://www.service.symantec.com/mx	
Support Gold	Costa Rica	800-242-9445
	Panama	800-234-4856
	Puerto Rico	800-232-4615

Asie-Pacifique

Symantec dispose d'un support technique et d'un service clientèle internationaux. Les services varient selon les pays et incluent des partenaires internationaux qui représentent Symantec dans les régions où il n'y a pas de bureau Symantec. Pour des informations générales, contactez le service de support de Symantec pour votre région.

Service et support

AUSTRALIE

Symantec Australia
Level 2, 1 Julius Avenue
North Ryde, NSW 2113
Australie

Numéro principal	+61 2 8879 1000	
Télécopie	+61 2 8879 1001	
Site Web	http://service.symantec.com	
Support Gold	1800 805 834	gold.au@symantec.com
Admin. contrats de support	1800 808 089	contractsadmin@symantec.com

CHINE

Symantec China
Unit 1-4, Level 11,
Tower E3, The Towers, Oriental Plaza
No.1 East Chang An Ave.,
Dong Cheng District
Beijing 100738
China P.R.C.

Numéro principal	+86 10 8518 3338
Support technique	+86 10 8518 6923
Télécopie	+86 10 8518 6928
Site Web	http://www.symantec.com.cn

HONG KONG

Symantec Hong Kong
Central Plaza
Suite #3006
30th Floor, 18 Harbour Road
Wanchai
Hong Kong

Numéro principal	+852 2528 6206
Support technique	+852 2528 6206
Télécopie	+852 2526 2646
Site Web	http://www.symantec.com.hk

INDE

Symantec India
Suite #801
Senteck Centrako
MMTC Building
Bandra Kurla Complex
Bandra (East)
Mumbai 400051, Inde

Numéro principal	+91 22 652 0658
Télécopie	+91 22 652 0617
Site Web	http://www.symantec.com./india
Support technique	+91 22 657 0669

COREE

Symantec Korea
15,16th Floor
Dukmyung B/D
170-9 Samsung-Dong
KangNam-Gu
Séoul 135-741
Corée du Sud

Numéro principal	+822 3420 8600
Télécopie	+822 3452 1610
Support technique	+822 3420 8650
Site Web	http://www.symantec.com.kr

MALAISIE

Symantec Corporation (Malaysia) Sdn Bhd
31-3A Jalan SS23/15
Taman S.E.A.
47400 Petaling Jaya
Selangor Darul Ehsan
Malaisie

Numéro principal	+603 7805 4910
Télécopie	+603 7804 9280
E-mail société	gold.apac@symantec.com
N° vert société	+1800 805 104
Site Web	http://www.symantec.com.my

NOUVELLE-ZELANDE

Symantec New Zealand
Level 5, University of Otago Building
385 Queen Street
Auckland Central 1001
Nouvelle-Zélande

Numéro principal	+64 9 375 4100	
Télécopie	+64 9 375 4101	
Site Web de support	http://service.symantec.co.nz	
Support Gold	0800 174 045	gold.nz@symantec.com
Admin. contrats de support	0800 445 450	contractsadmin@symantec.com

SINGAPOUR

Symantec Singapore
3 Phillip Street
#17-00 & #19-00 Commerce Point
Singapour 048693

Numéro principal	+65 6239 2000
Télécopie	+65 6239 2001
Support technique	+65 6239 2099
Site Web	http://www.symantec.com.sg

TAIWAN

Symantec Taiwan
2F-7, No.188 Sec.5
Nanjing E. Rd.,
105 Taïpei
Taïwan

Numéro principal	+886 2 8761 5800
Support entreprise	+ 886 2 8761 5800
Télécopie	+ 886 2 2742 2838
Site Web	http://www.symantec.com.tw

L'exactitude des informations contenues dans ce document a fait l'objet de toutes les attentions. Toutefois, les informations fournies ici sont susceptibles d'être modifiées sans préavis. Symantec Corporation se réserve le droit d'apporter ces modifications sans avertissement préalable.

